APRENDIZADO SUPERVISIONADO COM CONJUNTOS DE DADOS DESBALANCEADOS

Cristiano Leite de Castro*
crislcastro@ufmg.br

Antônio Pádua Braga*
apbraga@ufmg.br

*Universidade Federal de Minas Gerais
Departamento de Engenharia Eletrônica
Belo Horizonte, MG, Brasil

ABSTRACT

Supervised Learning with Imbalanced Data Sets: An Overview
Traditional learning algorithms induced by complex and highly imbalanced training sets may have difficulty in distinguishing between examples of the groups. The tendency is to create classification models that are biased toward the overrepresented (majority) class, resulting in a low rate of recognition for the minority group. This paper provides a survey of this problem which has attracted the interest of many researchers in recent years. In the scope of two-class classification tasks, concepts related to the nature of the imbalanced class problem and evaluation metrics are presented, including the foundations of the ROC (Receiver Operating Characteristic) analysis; plus a state of the art of the proposed solutions. At the end of the paper a brief discussion on how the subject can be extended to multiclass learning is provided.

KEYWORDS: imbalanced data sets, supervised learning, evaluation metrics, ROC analysis, resampling methods, cost-sensitive approach.

RESUMO

Algoritmos de aprendizado tradicionais induzidos por conjuntos de treinamento complexos e altamente desbalanceados têm apresentado dificuldade em diferenciar entre os grupos. A tendência é produzir modelos (ou regras) de classificação que favorecem a classe com maior probabilidade de ocorrência (majoritária), resultando em uma baixa taxa de reconhecimento para o grupo minoritário. O objetivo desse artigo é fornecer uma investigação sobre esse problema, que tem atraído o interesse de muitos pesquisadores nos últimos anos. No escopo de tarefas de classificação binária, são apresentados conceitos associados à natureza do problema de classes desbalanceadas e métricas de avaliação, incluindo os fundamentos da análise ROC (*Receiver Operating Characteristic*); além do estado da arte das soluções propostas na literatura. Uma breve discussão a respeito de como os tópicos abordados no artigo podem ser estendidos para o aprendizado multiclasse é também fornecida.

PALAVRAS-CHAVE: classes desbalanceadas, aprendizado supervisionado, métricas de avaliação, análise ROC, métodos de reamostragem, abordagem sensível ao custo.

1 INTRODUÇÃO

Um aspecto fundamental em problemas de classificação é a desigualdade na distribuição dos padrões entre os grupos, que surge principalmente em situações onde informações associadas a determinadas classes são mais difíceis de se obter. Pode-se observar esse comportamento, por exemplo, em um estudo sobre uma doença rara em uma dada população. A proporção de pessoas doentes encontradas é muito menor que a proporção de pessoas sadias. Em problemas dessa natureza, em que os números de exemplos entre as classes no conjunto de treinamento variam significativamente, algorit-

Artigo submetido em 07/10/2010 (Id.: 01203)
Revisado em 07/01/2011, 23/03/2011
Aceito sob recomendação do Editor Associado Prof. Ivan Nunes Da Silva

mos de aprendizado tradicionais têm apresentado dificuldade em distinguir entre os vários grupos. Em geral, a tendência é produzir modelos de classificação que favorecem as classes com maior probabilidade de ocorrência, resultando em baixas taxas de reconhecimento para os grupos minoritários.

O problema de classes desbalanceadas, como é conhecido em aprendizado de máquina e mineração de dados, surge principalmente porque os algoritmos tradicionais assumem diferentes erros como igualmente importantes, supondo que as distribuições são relativamente equilibradas (Monard and Batista, 2002; He and Garcia, 2009). Embora essa estratégia possa produzir modelos com elevadas taxas de acurácia global, ela frequentemente tende a prejudicar a identificação de exemplos pertencentes a grupos raros que, na maioria dos casos, representam os grupos de interesse.

De fato, na maioria das aplicações reais, detectar eventos anormais (ou interessantes) em uma população contendo grande número de eventos comuns é o principal objetivo. Tais aplicações, que comumente apresentam conjuntos de dados altamente complexos, têm sido reportadas em um grande número de domínios, tais como diagnóstico médico (Sun et al., 2007; Braga et al., 2008; Natowicz et al., 2008; Silva et al., 2009; Moturu et al., 2010), suporte à decisão em unidades de tratamento intensivo (Morik et al., 1999), detecção de fraudes/falhas (Fawcett and Provost, 1997; Carvalho et al., 2008; Gao et al., 2009), categorização de texto (Li and Shawe-Taylor, 2003; Manevitz and Yousef, 2007), reconhecimento de assinaturas (Souza et al., 2010), monitoramento de quebras de eixos automotivos (Hong et al., 2007), identificação de alertas de colisão entre aeronaves (Everson and Fieldsend, 2006b), entre outros.

Aprendizado com dados desbalanceados tem atraído o interesse de muitos pesquisadores nos últimos anos. Esse interesse aparece refletido, por exemplo, no grande número de estudos publicados sobre o assunto, na realização de *workshops* nas conferências AAAI (*Association for the Advancement of Artificial Intelligence*) (Japkowicz, 2000a) e ICML (*International Conference on Machine Learning*) (Chawla et al., 2003) e, em uma edição especial da revista ACM SIGKDD *Explorations* (Chawla et al., 2004).

O objetivo desse artigo é prover uma investigação sobre o problema de classes desbalanceadas com foco na abordagem discriminativa do aprendizado supervisionado, onde regras de decisão (classificadores) são induzidas diretamente do conjunto de dados a partir da minimização de um funcional risco (função custo). No âmbito dessa investigação, os conceitos relacionados ao problema assim como o estado da arte das soluções propostas são descritos no contexto de tarefas de classificação binária, ou seja, contendo somente duas classes. Na parte final do artigo, uma breve discussão a respeito de como os tópicos abordados na investigação podem ser estendidos para domínios multiclasse é fornecida.

O restante do artigo encontra-se organizado da seguinte forma. Na Seção 2, uma análise de cunho formal sobre a natureza do problema de classes desbalanceadas é apresentada com base nas Teorias de Decisão Bayesiana (Berger, 1985; Bather, 2000) e Aprendizado Estatístico (Vapnik, 1995; Vapnik, 1998). A discussão conduzida nessa seção fornece, através de fundamentos teóricos, melhor compreensão dos aspectos associados à origem do problema. Até o momento, esses aspectos não foram devidamente formalizados e, frequentemente, têm sido discutidos em caráter experimental. Na Seção 3, são descritas as medidas de desempenho comumente usadas para avaliar classificadores no contexto de aprendizado com grupos desbalanceados. Além disso, são apresentados os principais fundamentos da análise ROC (*Receiver Operating Characteristic*). A Seção 4 traz uma revisão crítica das abordagens propostas para solucionar o problema. Seguindo padrão adotado na literatura, essas abordagens foram divididas em duas grandes categorias: pré-processamento de dados e adaptações em algoritmos de aprendizado. Dentro da segunda categoria, uma maior atenção é dedicada às soluções baseadas em propostas e/ou modificações de funcionais risco otimizados por algoritmos de aprendizado. Por último, as discussões e conclusões são apresentadas na Seção 5.

2 PROBLEMA DE CLASSES DESBALANCEADAS

A maioria dos estudos sobre o problema de classes desbalanceadas foca no desenvolvimento de soluções. Uma quantidade menor tem investigado as suas causas e/ou tentado propor algum tipo de formalismo (Lawrence et al., 1998; Japkowicz and Stephen, 2002; Wu and Chang, 2003; Weiss and Provost, 2003; Prati et al., 2004b; Batista et al., 2004; Weiss, 2004; Khoshgoftaar et al., 2010). Nesses trabalhos, a metodologia comumente adotada é a caracterização do problema a partir de observações obtidas com resultados experimentais através de algoritmos de aprendizado específicos.

Nessa seção, uma interpretação para a natureza do problema de classes desbalanceadas é fornecida com base nos fundamentos das Teorias de Decisão Bayesiana e Aprendizado Estatístico. A argumentação é desenvolvida explorando as propriedades da solução (ou regra de decisão) ótima que minimiza a taxa de erro esperado, também conhecida como erro de generalização. Tal solução pode ser estimada e analisada analiticamente em um cenário controlado, onde todas as distribuições de probabilidade são conhecidas. A caracterização da natureza do problema é então conduzida contrastando as características da solução ótima com regras de decisão estimadas por modelos discriminativos, baseados na minimiza-

ção da taxa de erro global sobre um conjunto de treinamento desbalanceado. No decorrer da discussão, as principais conclusões e observações publicadas em Lawrence et al. (1998), Japkowicz and Stephen (2002), Wu and Chang (2003), Weiss and Provost (2003), Prati et al. (2004b), Batista et al. (2004), Weiss (2004) e, Khoshgoftaar et al. (2010) são contextualizadas e comentadas.

Como resultado da análise realizada, é demonstrado que o viés causado pelo grupo dominante é uma consequência direta da formulação padrão comumente adotada na abordagem discriminativa e também, do nível de incerteza (ruído) associado aos dados. Além disso, é apontada a falta de representatividade do grupo minoritário como fator importante a ser considerado no aprendizado com classes desbalanceadas.

Para a apresentação dos conceitos nas seções a seguir, considere as seguintes definições/notações fornecidas no escopo de classificação binária: um exemplo de entrada, representado por um vetor de características $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ deve ser atribuído a uma (e somente uma) das 2 classes (ou grupos) denotadas por C_0 e C_1. A existência das classes é conhecida a priori. Seja $y = \{0, 1\}$ a variável simbólica que denota a classe (rótulo) para um dado exemplo $\mathbf{x}$, tal que $y = k$ indica que $\mathbf{x}$ pertence à classe C_k. Sem perda de generalidade, assume-se que C_0 e C_1 e seus rótulos associados, correspondem, respectivamente, às classes majoritária (ou negativa) e minoritária (ou positiva). O objetivo da tarefa de classificação é portanto, construir um mapeamento (ou regra de decisão) que descreve o relacionamento entre as variáveis de entrada $\mathbf{x}$ e de saída y. Uma vez definida, tal regra pode ser usada para decidir a classe para um dado exemplo de entrada, i.e., estimar y a partir de $\mathbf{x}$.

2.1 Teoria de Decisão Bayesiana

A Teoria de Decisão Bayesiana fornece o modelo probabilístico fundamental para os bem conhecidos procedimentos de classificação de padrões (Berger, 1985; Bather, 2000). Com base nesse modelo, regras de decisão ótimas podem ser obtidas quando as distribuições de probabilidade são conhecidas.

Em um problema de classificação, sejam $p(\mathbf{x}|y = k)$ e $P(y = k)$ respectivamente, a densidade condicional e a probabilidade de ocorrência (a priori) para a classe C_k. A partir dessas quantidades, a probabilidade (a posteriori) de um exemplo $\mathbf{x}$, previamente observado, pertencer à classe C_k pode ser calculada, usando o teorema de *Bayes* a seguir (Duda et al., 2000),

$$P(y = k|\mathbf{x}) = \frac{p(\mathbf{x}|y = k)P(y = k)}{p(\mathbf{x})} \ . \qquad (1)$$

onde $p(\mathbf{x}) = \sum_k p(\mathbf{x}|y = k)P(y = k)$ é a densidade (incondicional) da entrada $\mathbf{x}$.

Uma regra de decisão binária divide o espaço de entrada em duas regiões disjuntas, denotadas por $\mathbf{R}_0$ e $\mathbf{R}_1$, uma para cada classe, tal que todos os pontos em $\mathbf{R}_k$ serão assinalados à classe C_k. Os limites entre as regiões de decisão são conhecidos como superfícies de decisão (ou separação).

Seja λ uma *função de perda* que associa custos às possíveis decisões tomadas por uma regra de decisão. λ é comumente descrita através de uma matriz de custo, onde o elemento λ_{kj} fornece o custo associado ao se classificar um exemplo $\mathbf{x}$ à classe C_j, sendo que $\mathbf{x}$ pertence à classe C_k[1].

A melhor regra de decisão (ótima) que pode ser obtida é aquela que minimiza o *risco* global, que corresponde ao valor esperado (médio) da perda em relação às densidades de probabilidade conjuntas $p(\mathbf{x}, y = k)$,

$$\begin{aligned} \mathbb{E}[\lambda] \ &= \ \int_{\mathbf{R}_1} \lambda_{01}\, p(\mathbf{x}, y = 0)\, d\mathbf{x} \\ &\quad + \int_{\mathbf{R}_0} \lambda_{10}\, p(\mathbf{x}, y = 1)\, d\mathbf{x} \ . \end{aligned} \qquad (2)$$

onde $p(\mathbf{x}, y = k) = p(\mathbf{x}|y = k)P(y = k)$. Seja $L_j(\mathbf{x})$ o *risco* condicional (ou custo esperado) de atribuir um exemplo arbitrário $\mathbf{x}$ à classe C_j (Duda et al., 2000),

$$L_j(\mathbf{x}) = \lambda_{kj} P(y = k|\mathbf{x}) \ . \qquad (3)$$

onde $P(y = k|\mathbf{x})$ é a probabilidade a posteriori de $\mathbf{x}$ pertencer à classe k definida em (1). Baseado na definição do *risco* condicional, a regra de decisão ótima que minimiza o *risco* global (2) é aquela que atribui cada vetor de entrada $\mathbf{x}$ à classe C_j para a qual $L_j(\mathbf{x})$ é mínimo, i.e.,

$$r\,(\mathbf{x}) = \begin{cases} C_1 & \text{se } \frac{P(y=1|\mathbf{x})}{P(y=0|\mathbf{x})} > \frac{\lambda_{01}}{\lambda_{10}}, \\ C_0 & \text{caso contrário.} \end{cases} \qquad (4)$$

Usando o teorema de *Bayes* (1), a regra (4) pode ser reescrita em termos da razão entre as densidades condicionais $p(\mathbf{x}|y = k)$ para cada classe (razão de verossimilhança),

$$r\,(\mathbf{x}) = \begin{cases} C_1 & \text{se } \frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} > \frac{\lambda_{01}}{\lambda_{10}} \frac{P(y=0)}{P(y=1)}, \\ C_0 & \text{caso contrário.} \end{cases} \qquad (5)$$

[1] Particularmente nesse artigo, os custos das classificações corretas são considerados como zero, i.e., $\lambda_{kk} = 0$ para toda classe C_k. No entanto, para um tratamento mais geral, veja Elkan (2001).

De acordo com (5), $\mathbf{x}$ é atribuído à C_1 (classe positiva) se a razão de verossimilhança excede um limiar (*threshold*) independente de $\mathbf{x}$, que é baseado na interação das razões entre custos e probabilidades a priori dos grupos. Essa interação possui importante papel na determinação das probabilidades de erro para cada classe e tem sido muito explorada na obtenção de soluções para o problema de classes desbalanceadas. Esses aspectos são discutidos mais adiante nas Seções 3 e 4, respectivamente.

Note que quando as densidades conjuntas $p(\mathbf{x}, y = k)$ e os custos λ_{kj} são conhecidos, regras de decisão ótimas que minimizam o funcional *risco* (2) podem ser diretamente obtidas. Funções discriminantes lineares e quadráticas, por exemplo, podem ser derivadas analiticamente, considerando que as verossimilhanças $p(\mathbf{x}|y = k)$ são distribuições normais multivariadas (Duda et al., 2000). Essa propriedade é explorada na Seção 2.3, durante a caracterização da natureza do problema. Particularmente, funções discriminantes lineares serão usadas para a obtenção de superfícies de decisão ideais em cenários desbalanceados.

Na Seção 2.2 a seguir, a formulação padrão para a tarefa de classificação de padrões é apresentada com base na Teoria do Aprendizado Estatístico (SLT) (Vapnik, 1995; Vapnik, 1998). Nessa abordagem é assumido por definição, que as densidades $p(\mathbf{x}, y = k)$ são desconhecidas. Assim, regras de decisão devem ser aprendidas usando somente um conjunto de dados observados.

2.2 Problema do Aprendizado

Considerando o caso particular, em que $\lambda_{01} = \lambda_{10} = 1$ (*função de perda* $0/1$), o funcional *risco* global (2) se reduz à probabilidade do erro global de classificação (ou taxa de erro esperado) dado pela seguinte expressão (Duda et al., 2000),

$$\begin{aligned} P\,(\text{Erro}) &= P(\mathbf{x} \in \mathbf{R}_1, y = 0) + P(\mathbf{x} \in \mathbf{R}_0, y = 1) \\ &= \int_{\mathbf{R}_1} p(\mathbf{x}, y = 0)\, d\mathbf{x} \\ &\quad + \int_{\mathbf{R}_0} p(\mathbf{x}, y = 1)\, d\mathbf{x}\ . \end{aligned} \tag{6}$$

onde $P(\mathbf{x} \in \mathbf{R}_j, y = k)$ é a probabilidade conjunta de $\mathbf{x}$ ser atribuído à classe C_j, sendo que sua verdadeira classe é C_k.

Sob a formulação do aprendizado estatístico, dado um conjunto finito de exemplos (conjunto de treinamento),

$$\{(\mathbf{x}_i, y_i) \in \mathbb{R}^n \times \{0, 1\} \mid i = 1 \ldots N\} \tag{7}$$

obtidos (i.i.d.) a partir das distribuições desconhecidas $p(\mathbf{x}, y = k)$, o problema de classificação é encontrar a função ótima f^* (ou regra de decisão) que minimiza a probabilidade do erro global de classificação (6) sobre a classe de funções $f : \mathbb{R}^n \rightarrow \{0, 1\}$ suportadas pela máquina (algoritmo) de aprendizado.

Note que a formulação apresentada assume consequências (custos) iguais para os diferentes erros de classificação (*função de perda* $0/1$), visando assim a minimização de um critério (funcional) que corresponde à taxa de erro esperado. Embora a premissa de custos iguais seja mais fiel ao modelo probabilístico adotado, ela tende, em um cenário desbalanceado, a produzir regras de decisão que favorecem a classe com maior probabilidade de ocorrência (majoritária). Essa característica pode não ser adequada para muitos problemas reais em que o objetivo é detectar eventos raros a partir de uma população contendo grande quantidade de eventos comuns.

A Seção 2.3 a seguir, fornece uma interpretação para o problema de classes desbalanceadas. Desde que sua natureza está diretamente associada à formulação padrão do problema de aprendizado, a discussão é conduzida com base nas propriedades da solução ótima f^* que minimiza a taxa de erro esperado (6).

2.3 Natureza do Problema

A regra de decisão ótima f^* pode ser obtida a partir das expressões (4) ou (5), ao considerarmos custos iguais para os erros de classificação ($\lambda_{01} = \lambda_{10}$). Tal regra, popularmente conhecida como *regra de Bayes* (Duda et al., 2000), pode ser descrita em função das verossimilhanças $p(\mathbf{x}|y = k)$, da seguinte forma,

$$f^*\,(\mathbf{x}) = \begin{cases} 1 & \text{se } \frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} > \frac{P(y=0)}{P(y=1)}, \\ 0 & \text{caso contrário.} \end{cases} \tag{8}$$

Observe a partir de (8), que a decisão sobre a pertinência de um exemplo arbitrário à classe positiva (minoritária) é diretamente influenciada pela razão entre as probabilidades de ocorrência das classes. Assim, para um problema com classes desbalanceadas, em que o limiar $\frac{P(y=0)}{P(y=1)}$ é muito maior que 1, a solução ótima f^*, buscada pelas máquinas de aprendizado, naturalmente deve favorecer a classe majoritária. Para fins de ilustração, considere a situação hipotética apresentada na Figura 1, onde as densidades condicionais $p(x|y = k)$ são representadas por distribuições gaussianas unidimensionais (conhecidas) possuindo sobreposição e mesma variância; x^* é a superfície de decisão estimada a partir da regra f^* que divide o espaço de entrada entre as regiões $\mathbf{R}_0$ e $\mathbf{R}_1$. Note, através dessa figura, que se uma ambiguidade surge na classificação de um exemplo de entrada

particular x_i, devido aos valores similares observados para as densidades condicionais, i.e., $p(x|y=0) \approx p(x|y=1)$, f^* irá atribuir x_i à classe majoritária, desde que a razão entre as verossimilhanças não excede o limiar imposto por $\frac{P(y=0)}{P(y=1)}$. Analisando a superfície de decisão x^* no espaço de entrada, um desvio em direção à classe minoritária pode ser verific

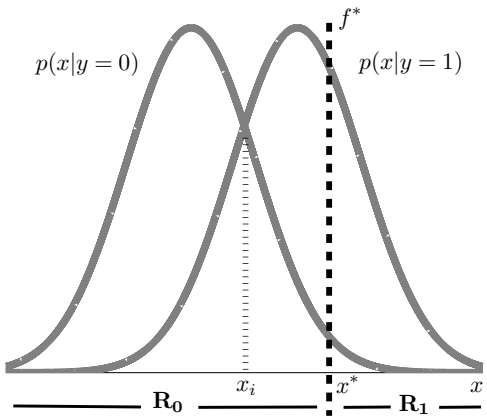

Figura 1: Ilustração do problema através de distribuições unidimensionais conhecidas. Desde que as prioris são desbalanceadas, a solução ótima f^* favorece a classe majoritária.

Em situações práticas, no entanto, não é possível encontrar exatamente f^*. Assim, define-se $\hat{f}$ como uma estimativa da solução ótima obtida a partir de um conjunto finito de exemplos usando algum método de aprendizado[2]. Considere então, um cenário desbalanceado, em que as proporções de exemplos para as classes no conjunto de treinamento refletem as probabilidades de ocorrência $P(y=k)$. Desde que a regra de decisão estimada $\hat{f}$ aproxima f^*, é esperado devido ao viés imposto pelo grupo dominante, que um número maior de erros seja obtido para a classe minoritária. Essa característica é ilustrada no exemplo a seguir, onde $\hat{f}$ é estimada e avaliada, respectivamente, a partir de conjuntos representativos de treinamento e teste gerados (i.i.d.) de acordo com $p(\mathbf{x}, y=k)$. A Figura 2 apresenta dados sintéticos (treinamento) obtidos a partir de duas distribuições gaussianas bidimensionais com vetores de média $\mu_0 = (-1,-1)$ e $\mu_1 = (1,1)$, e matrizes de covariância Σ_k (diagonais) cujos elementos na diagonal principal são iguais a 1.5. Os círculos pontilhados concêntricos marcam as curvas de nível para as distribuições. A razão entre o número de exemplos da classe majoritária (círculos) e minoritária (cruzes) é $19:1$. Duas superfícies de decisão podem ser observadas: (i) f^* (linha tracejada) estimada analiticamente a partir das expressões de $p(\mathbf{x}|y=k)$ e das probabilidades $P(y=k)$ (conhecidas) (Duda et al., 2000); (ii) $\hat{f}$ (linha contínua) estimada por uma *Support Vector Machine* (SVM) (Boser et al., 1992; Cortes and Vapnik, 1995) com *kernel* linear através do conjunto de treinamento desbalanceado. Note que $\hat{f} \approx f^*$ e, uma vez que o cenário é desbalanceado, ambas as superfícies encontram-se desviadas em direção à classe com menor número de exemplos. Na Figura 3, $\hat{f}$ foi avaliada em relação ao conjunto de teste. Foram observados 6 erros em relação à classe minoritária e apenas 1 erro em relação à classe majoritária. Exemplo similar ao apresentado, é mostrado em Wu and Chang (2003) com o objetivo de caracterizar o desvio da função de decisão $\hat{f}$ estimada por uma SVM com dados desbalanceados. Nesse trabalho, no entanto, os autores consideram distribuições uniformes para as densidades condicionais $p(\mathbf{x}|y=k)$ e, assumem a existência de uma superfície de separação "ideal" que é então usada como referência para avaliar o desvio apresentado por $\hat{f}$.

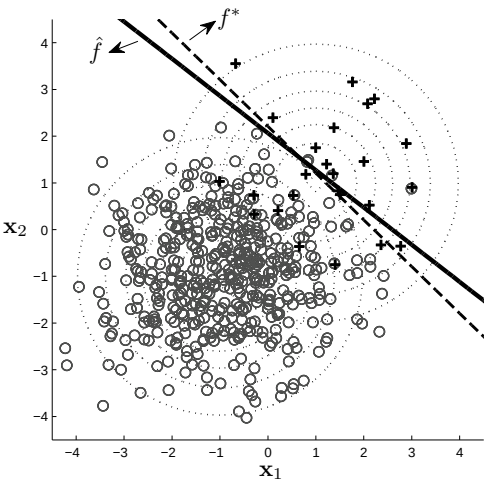

Figura 2: Superfície de decisão $\hat{f}$ estimada usando o conjunto de dados desbalanceado; $\hat{f} \approx f^*$ e portanto, também encontra-se desviada em direção ao grupo minoritário.

É importante ressaltar que o problema em questão não é somente causado pelo desequilíbrio entre as distribuições a priori dos grupos. Outro fator determinante é o nível de incerteza (ruído) associado à tarefa de classificação. Através de experimentos conduzidos com dados sintéticos e reais, os trabalhos de Japkowicz and Stephen (2002) e, Prati et al. (2004b) mostraram que para uma mesma razão de desbalanceamento, um aumento no nível de sobreposição das classes pode diminuir significativamente o número de classificações corretas para a classe minoritária. Em Japkowicz and Stephen (2002), os autores também sugeriram que domínios linearmente separáveis são praticamente insensíveis ao desbalanceamento. Em trabalho recente, Khoshgoftaar et al. (2010) realizaram uma extensa investigação empírica

[2]Um princípio indutivo fornece uma prescrição geral para a obtenção de $\hat{f}$ sobre a classe de funções $f : \mathbb{R}^n \rightarrow \{0,1\}$ suportadas pela máquina de aprendizado (Cherkassky and Mulier, 2007). Princípios indutivos comumente usados como, por exemplo, Minimização Estrutural do Risco (Vapnik, 1995) e Regularização (Girosi et al., 1995), estabelecem condições que permitem a escolha de uma função que constitui uma boa aproximação para a solução ótima. Ambos são baseados em medidas de complexidade para a classe de funções adotada.

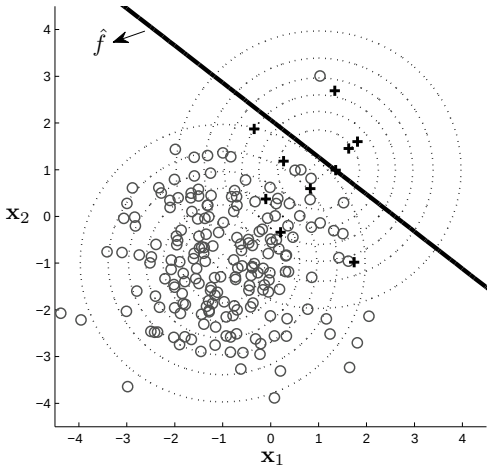

Figura 3: Avaliação de $\hat{f}$ sobre o conjunto de teste.

sobre o impacto causado pela combinação "ruído + desbalanceamento" no aprendizado de modelos baseados em redes *Multilayer Perceptron* (MLP) e *Radial Basis Function* (RBF) (Haykin, 1994). Como resultado da investigação, foi reportado que embora as redes MLP tenham se apresentado mais robustas à presença de "ruído + desbalanceamento" do que as redes RBF, a capacidade de discriminação de ambos os modelos diminui em função do aumento desses fatores. As conclusões obtidas nesses estudos se alinham com as propriedades apresentadas por funções discriminantes lineares derivadas de distribuições gaussianas multivariadas $p(\mathbf{x}|y = k)$. Para esses discriminantes, o deslocamento causado pela diferença entre as prioris é diretamente proporcional à razão entre as variâncias e o quadrado da distância entre os centróides das classes (Gallinari et al., 1991; Duda et al., 2000). Assim, se as variâncias das classes são muito pequenas em relação às distâncias entre seus centróides, as superfícies de decisão estimadas são relativamente independentes do desbalanceamento. Isso explica porque, para determinadas aplicações, pequenas razões de desbalanceamento podem comprometer mais a capacidade de reconhecimento da classe positiva do que as grandes.

Aplicações reais apresentando razões de desbalanceamento da ordem de $100 : 1$, $1000 : 1$ e até $10000 : 1$ foram reportadas, respectivamente, em He and Shen (2007), Kubat et al. (1998) e Pearson et al. (2003). Em aplicações dessa natureza, dependendo do nível de sobreposição apresentado pelas classes, regras de decisão obtidas pela simples minimização da taxa de erro global, podem vir a perder sua capacidade de discriminação, classificando todos os exemplos como pertencentes à classe dominante. Nesses casos extremos, toda a classe minoritária pode se tornar componente do erro irredutível do classificador (Erro *Bayes*). Em Lawrence et al. (1998), os autores chamam a atenção para a falta de reconhecimento de uma das classes quando os níveis de sobreposição e desbalanceamento são muito elevados. Eles demonstram essa característica através de um experimento com dados sintéticos usando um classificador baseado em rede MLP (Haykin, 1994).

Na discussão apresentada até agora, considerou-se que o grupo minoritário é representativo, i.e, que a quantidade (e a disposição espacial) de exemplos é suficiente para representar as distribuições (alvo) $p(\mathbf{x}|y = 1)$ e $P(y = 1)$ no conjunto de treinamento. Com base nessa premissa, foi mostrado que o problema de classes desbalanceadas surge como uma propriedade inerente das soluções baseadas na taxa de erro global e, que a intensidade do viés causado pelo grupo dominante está mais associada à complexidade dos dados (nível de sobreposição) do que a própria desproporção apresentada pelas classes. A partir dessas conclusões, é importante deixar claro que, para tarefas de classificação em que os grupos representam *clusters* bem definidos e separáveis no espaço de entrada, a influência do desbalanceamento deve ser mínima e, em geral, não deve prejudicar o reconhecimento da classe positiva. Para ilustrar essa idéia, considere o *toy problem* "Duas Luas" na Figura 4. Nesse exemplo, devido à separabilidade das distribuições $p(\mathbf{x}|y = k)$, a regra de decisão ótima[3] f^* (linha tracejada) praticamente não sofre influência do desequilíbrio entre as prioris $P(y = k)$ (razão $5 : 1$). As soluções $\hat{f}_1$ (linha contínua) e $\hat{f}_2$ (linha pontilhada) foram estimadas, respectivamente, por uma SVM com *kernel* RBF e uma rede MLP treinada com algoritmo MOBJ (Teixeira et al., 2000), usando o conjunto de treinamento formado pelas classes negativa (círculos) e positiva (losangos preenchidos). Note pela Figura 4, que apesar do grau de desbalanceamento (razão $5 : 1$), não houve perda na capacidade de reconhecimento da classe de interesse. Observe também, que as regras de decisão $\hat{f}_1$ (linha contínua) e $\hat{f}_2$ (linha pontilhada) possuem forma similar à f^* (linha tracejada), uma vez que a classe positiva, embora contenha poucos exemplos (losangos preenchidos), ainda é capaz de representar a distribuição alvo (losangos). Concordando com a argumentação apresentada nesse exemplo, alguns estudos experimentais têm mostrado que, para determinados problemas reais, o aprendizado da classe de interesse não sofre influência do desbalanceamento das distribuições (Weiss and Provost, 2003; Batista et al., 2004).

Para finalizar a discussão sobre o problema de classes desbalanceadas, é chamada a atenção para a possível falta de representatividade da classe minoritária no conjunto de treinamento. Esse aspecto, conhecido como "raridade absoluta" (Weiss, 2004; Weiss, 2005) surge principalmente devido à dificuldade inerente na obtenção de amostras pertencentes a

[3]Nos exemplos ilustrados pelas Figuras 4 e 5, a regra de decisão ótima f^* foi representada pela superfície de decisão de margem máxima em relação às distribuições alvo.

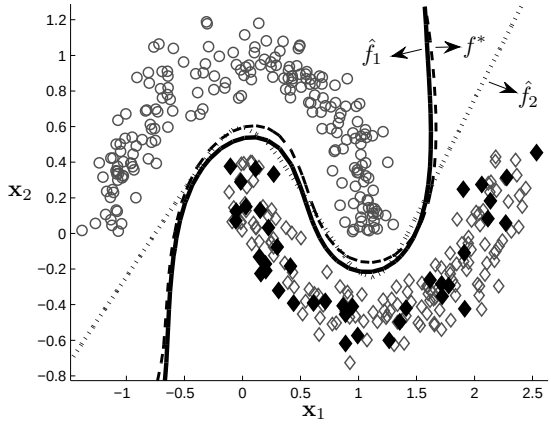

Figura 4: "Separabilidade": distribuições separáveis (sem ruído) asseguram que f^* e suas aproximações, $\hat{f}_1$ e $\hat{f}_2$, apresentem boa capacidade de reconhecimento, independente do desbalanceamento dos grupos.

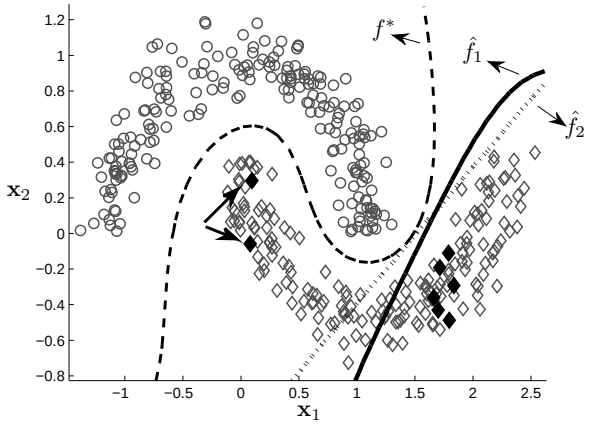

Figura 5: "Raridade absoluta:" a falta de representatividade das amostras positivas (losangos preenchidos) em relação à distribuição real (losangos) produz regras de decisão, $\hat{f}_1$ e $\hat{f}_2$, muito diferentes de f^*.

grupos raros. Nesses domínios, em que os exemplos positivos não são suficientes para representar as distribuições alvo, a qualidade da aproximação $\hat{f}$ em relação a solução ótima f^* pode ficar comprometida, independente dos fatores de desbalanceamento e sobreposição do conjunto de dados. Essa característica é ilustrada na Figura 5 através de um exemplo simples, também baseado no *toy problem* "Duas Luas". Nesse exemplo, amostras positivas (losangos preenchidos) são muito raras e portanto, não são capazes de representar de forma significativa a distribuição real (losangos). Como resultado, as regras de decisão $\hat{f}_1$, SVM com *kernel* RBF (linha contínua) e $\hat{f}_2$, rede MLP-MOBJ (linha pontilhada), estimadas a partir do conjunto de treinamento (razão $25:1$), apresentam-se muito distantes de f^* (linha tracejada). Observe ainda pela Figura 5, que devido ao conceito de "raridade absoluta", exemplos positivos isolados (indicados pelas setas) foram considerados como ruído e ignorados na estimação de $\hat{f}_1$ e $\hat{f}_2$.

Na Seção 3, a seguir, são apresentadas as métricas comumente usadas para se avaliar o desempenho de classificadores em aplicações desbalanceadas.

3 MÉTRICAS DE AVALIAÇÃO PARA PROBLEMAS DESBALANCEADOS

Tradicionalmente, a métrica usada na avaliação e seleção de modelos de classificação é a *acurácia* (ou *taxa de erro*) estimada em relação a um dado conjunto de teste. Essa metodologia é justificada pela formulação padrão do problema do aprendizado supervisionado que visa a minimização da probabilidade do erro global. Para problemas altamente desbalanceados, no entanto, a *acurácia* pode não fornecer informação adequada sobre a capacidade de discriminação de um classificador $\hat{f}$ em relação a um dado grupo específico (de interesse). Considere, por exemplo, um conjunto de dados em que a classe minoritária é representada por apenas 2% das observações. Um classificador com *acurácia* de 98% pode ser diretamente obtido, por simplesmente classificar todo exemplo como pertencente à classe majoritária. Apesar da elevada taxa de *acurácia* obtida, tal classificador torna-se inútil se o objetivo principal é a identificação de exemplos raros.

Muitos trabalhos têm chamado a atenção para os problemas causados pelo uso da *acurácia* em cenários desbalanceados (Bradley, 1997; Provost and Fawcett, 1997; Provost et al., 1998; Maloof, 2003; Cortes and Mohri, 2004; Sun et al., 2007). Nesse contexto, uma maneira mais eficaz de se avaliar um dado classificador $\hat{f}$ é através da distinção dos erros (ou acertos) cometidos para cada classe. Isso pode ser obtido descrevendo o desempenho de $\hat{f}$ a partir de uma matriz de confusão ou tabela de contingência (vide Tabela 1) (Fawcett, 2006). Cada elemento $e_{k,j}$ dessa matriz fornece o número de exemplos, cuja verdadeira classe era C_k e que foi atualmente classificado como C_j. Assim, os elementos ao longo da diagonal principal representam as decisões corretas: número de verdadeiros negativos (TN) e verdadeiros positivos (TP); enquanto os elementos fora dessa diagonal representam os erros cometidos: número de falsos positivos (FP) e falsos negativos (FN).

Tabela 1: Matriz de Confusão para um classificador binário.

	predição ($y=0$)	predição ($y=1$)
real ($y=0$)	TN	FP
real ($y=1$)	FN	TP

A partir da Tabela 1, é possível extrair 4 métricas importantes que diretamente avaliam, de forma independente, o desempenho sobre as classes positiva e negativa,

$$\textit{Taxa de Falsos Positivos: } FPr = \frac{FP}{TN + FP} \quad (9)$$

$$\textit{Taxa de Falsos Negativos: } FNr = \frac{FN}{TP + FN} \quad (10)$$

$$\textit{Taxa de Verdadeiros Positivos: } TPr = \frac{TP}{TP + FN} \quad (11)$$

$$\textit{Taxa de Verdadeiros Negativos: } TNr = \frac{TN}{TN + FP} \quad (12)$$

Além das taxas de erro/acerto para cada classe, outras métricas têm sido frequentemente adotadas com o objetivo de fornecer avaliações mais adequadas para aplicações desbalanceadas (Sun et al., 2007; He and Garcia, 2009). Em geral, esses critérios focam na detecção da classe minoritária ou consideram com mesma relevância a discriminação de ambas as classes. Entre as medidas mais usadas, encontram-se:

1. *F-measure*: a métrica *F-measure* considera somente o desempenho para a classe positiva. Ela é calculada a partir de duas importantes métricas adotadas em Recuperação de Informação: *Recall* e *Precision* (Tan et al., 2005). *Recall* (R) é equivalente à taxa de verdadeiros positivos (TPr) e denota a razão entre o número de exemplos positivos corretamente classificados e o número total de exemplos positivos originais,

$$R = TPr = \frac{TP}{TP + FN} \ . \quad (13)$$

Precision (P), por sua vez, corresponde à razão entre o número de exemplos positivos corretamente classificados e o número total de exemplos identificados como positivos pelo classificador,

$$P = \frac{TP}{TP + FP} \ . \quad (14)$$

Baseado nessas definições, *F-measure* pode ser calculada como,

$$\textit{F-measure} = \frac{(1 + \beta) \cdot R \cdot P}{\beta^2 \cdot R + P} \ . \quad (15)$$

onde β é usado para ajustar a importância relativa entre *Recall* e *Precision*. Tipicamente, $\beta = 1$.

2. *G-mean*: a métrica *G-mean* foi proposta por Kubat et al. (1998) e corresponde à média geométrica entre as taxas de verdadeiros positivos (TPr) e verdadeiros negativos (TNr),

$$\textit{G-mean} = \sqrt{TPr \cdot TNr} \ . \quad (16)$$

G-mean mede o desempenho equilibrado de um classificador em relação às taxas de acertos de ambas as classes (Sun et al., 2007).

3.1 Análise ROC

Apesar das métricas apresentadas na Seção 3 serem mais eficientes na avaliação de classificadores em cenários desbalanceados, elas não permitem comparar seus desempenhos sobre uma faixa de valores de distribuições a priori ou custos de erros de classificação. Essa limitação, no entanto, pode ser superada através dos gráficos (curvas) *Receiver Operating Characteristic* (ROC) que foram originalmente desenvolvidos na Teoria de Detecção de Sinais (Egan, 1975; Swets et al., 2000) e, nos últimos anos, têm sido usados pelas comunidades de Aprendizado de Máquina e Mineração de Dados para visualização, avaliação e seleção de modelos (Spackman, 1989; Fawcett, 2004; Fawcett, 2006; Prati et al., 2008a).

3.1.1 Curvas ROC

As curvas ROC possuem propriedades que as tornam especialmente úteis para domínios com classes desbalanceadas e custos de erros desiguais. Para compreender seu significado teórico, considere a seguinte regra de decisão expressa através da razão entre as densidades condicionais (razão de verossimilhança),

$$r(\mathbf{x}) = \begin{cases} C_1 & \text{se } \frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} > \theta, \\ C_0 & \text{caso contrário.} \end{cases} \quad (17)$$

Note que essa regra possui forma similar a (5) exceto que as razões entre os custos $\frac{\lambda_{01}}{\lambda_{10}}$ e probabilidades a priori $\frac{P(y=0)}{P(y=1)}$ estão implícitas no limiar de decisão θ (*threshold*). Assim, variar o limiar θ implica em variar as razões entre os custos $\frac{\lambda_{01}}{\lambda_{10}}$ e/ou probabilidades a priori $\frac{P(y=0)}{P(y=1)}$ (Cherkassky and Mulier, 2007).

Supondo que as distribuições $p(\mathbf{x}|y = k)$ são conhecidas (ou foram estimadas), um valor específico para θ determina as probabilidades de erro/acerto para cada classe: $P(\mathbf{x} \in \mathbf{R}_0|y = 1)$ (falsos negativos), $P(\mathbf{x} \in \mathbf{R}_1|y = 1)$ (verdadeiros positivos), $P(\mathbf{x} \in \mathbf{R}_1|y = 0)$ (falsos positivos)

e $P(\mathbf{x} \in \mathbf{R}_0|y = 0)$ (verdadeiros negativos); veja Figura 6. Tais probabilidades podem ser calculadas analiticamente por

$$P(\mathbf{x} \in \mathbf{R}_j|y = k) = \int_{\mathbf{R}_j} p(\mathbf{x}|y = k)d\mathbf{x}\ , \qquad (18)$$

com $j, k \in \{0, 1\}$.

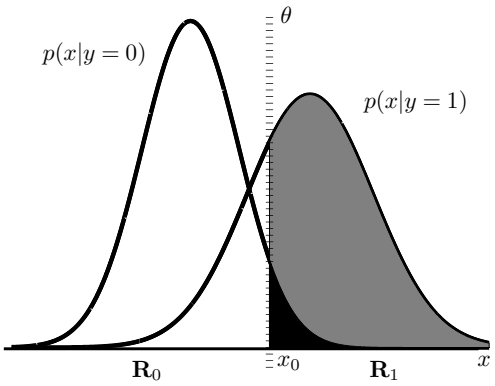

Figura 6: Limiar de decisão (θ) determinando as probabilidades de acerto para a classe positiva (área em cinza) e erro para a classe negativa (área em preto).

A capacidade de discriminação da regra (17) sobre toda a faixa de valores do limiar ($0 \leq \theta \leq \infty$) é dada pela curva *Receiver Operating Characteristic* (ROC) (Cherkassky and Mulier, 2007). Como pode ser visto na Figura 7, uma curva ROC reflete os erros de classificação em termos das probabilidades de detecção $P(\mathbf{x} \in \mathbf{R}_1|y = 1)$ (eixo vertical) e falsos alarmes $P(\mathbf{x} \in \mathbf{R}_1|y = 0)$ (eixo horizontal) quando θ é variado. Portanto, θ controla a fração de exemplos da classe C_1 corretamente classificados versus a fração de exemplos da classe C_0 incorretamente classificados. Esse relacionamento é também conhecido como *trade-off* sensibilidade-especificidade (Lasko et al., 2005).

3.1.2 Estimando Curvas ROC a partir de Conjuntos de Dados

Na prática, as distribuições das classes são desconhecidas e somente os dados de treinamento encontram-se disponíveis. Dessa forma, na abordagem generativa do aprendizado, a obtenção da curva ROC envolve a estimação das densidades $p(\mathbf{x}|y = k)$ a partir desses dados e, posterior variação do limiar na regra de decisão (17).

Nesse artigo, no entanto, a discussão está focada na abordagem discriminativa, onde regras de decisão são estimadas diretamente do conjunto de treinamento a partir da minimização de um funcional risco. Nesse caso, considere que a função estimada $\hat{f}$ produz, para cada exemplo $\mathbf{x}$, um *score*[4] que representa o grau de pertinência do exemplo à classe positiva. Uma curva ROC pode então ser obtida a partir da variação de um limiar de decisão θ sobre toda a faixa de *scores* (*ranking*) produzida. Cada valor de θ determina valores absolutos para as taxas de detecção (verdadeiros positivos) e falsos alarmes (falsos positivos). Sua variação sobre toda a faixa de saída de $\hat{f}$ gera uma curva que mostra graficamente o *trade-off* entre a taxa de verdadeiros positivos (TPr) e a taxa de falsos positivos (FPr). Para um conjunto finito de dados, essas quantidades correspondem, respectivamente, às estimativas para as probabilidades $P(\mathbf{x} \in \mathbf{R}_1|y = 1)$ e $P(\mathbf{x} \in \mathbf{R}_1|y = 0)$ (veja Figura 7). A curva ROC do classificador "ideal" possui o formato da função de *Heaviside* (*Heaviside step function*) no domínio $0 \leq FPr \leq 1$, indicando que $\hat{f}$ foi capaz de assinalar *scores* mais elevados para os exemplos positivos do que para os exemplos negativos. Isso caracteriza um *ranking* perfeito. Um algoritmo eficiente para computar a curva ROC pode ser encontrado em Fawcett (2006).

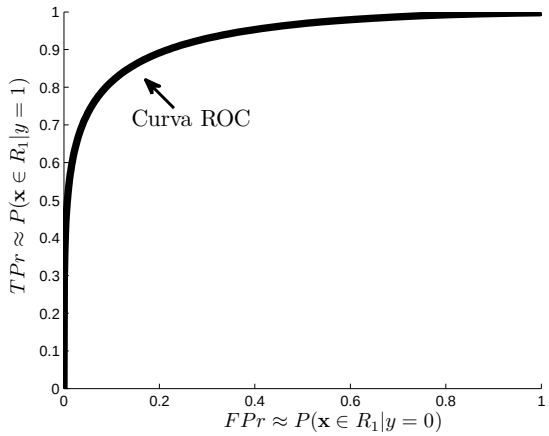

Figura 7: Curva ROC descrevendo o *trade-off* entre as probabilidades (ou taxas) de detecção e falsos alarmes.

Preferencialmente, um conjunto de teste deve ser usado para a obtenção da curva ROC que fornece uma estimativa da capacidade discriminativa do classificador em termos das probabilidades de erro (Cherkassky and Mulier, 2007). Uma vez estimada, essa curva é útil para a escolha de um ponto de operação θ segundo um critério adotado (Provost and Fawcett, 1998; Provost and Fawcett, 2001). Por exemplo, pode-se escolher um classificador (ponto de operação) que garanta uma probabilidade muito pequena de erros do tipo falso positivo (*critério de Neyman-Pearson*) (Duda et al., 2000). Cabe ressaltar entretanto, que a acurácia da curva ROC ob-

[4] Para obtenção da Curva ROC, os *scores* produzidos por um classifi cador não precisam representar estimativas exatas das probabilidades a posteriori (*scores* relativos). Em Zadrozny and Elkan (2001), no entanto, os autores mostram alguns métodos para obter probabilidades a posteriori calibradas a partir de *scores* relativos.

tida (através do conjunto de teste) é dependente da qualidade da solução estimada $\hat{f}$ usando os dados de treinamento.

Diferentes classificadores podem ser comparados através de suas curvas ROC, contrastando seus desempenhos de detecção TPr para vários valores de θ ou, equivalentemente, FPr. Em alguns casos, as curvas ROC cruzam, indicando que um classificador não fornece melhor desempenho para todos os valores de θ. A *Area Under the ROC Curve* (AUC) (Hanley and Mcneil, 1982) fornece uma medida geral da capacidade de discriminação do classificador que é independente do valor selecionado para θ. Isso resulta em uma medida de desempenho que é insensível aos custos de classificação e probabilidades a priori.

4 ESTADO DA ARTE DAS SOLUÇÕES

A discussão conduzida na Seção 2, mostrou que a formulação padrão comumente adotada na obtenção de modelos discriminativos pode prejudicar a identificação de exemplos da classe minoritária (de interesse) quando os dados disponíveis apresentam níveis elevados de desbalanceamento e sobreposição. A obtenção de soluções que melhoram o número de classificações positivas corretas, compensando (ou aliviando) o efeito causado pelo desequilíbrio entre as distribuições é portanto, um dos objetivos da pesquisa em aprendizado com classes desbalanceadas.

De forma geral, as abordagens que têm sido propostas para tratar do problema podem ser enquadradas em duas grandes categorias de acordo com a estratégia adotada. Na primeira categoria, denominada *pré-processamento de dados*, a idéia básica é modificar as distribuições das classes no conjunto de treinamento através de mecanismos de reamostragem de dados no espaço de entrada. A segunda categoria envolve *adaptações em métodos de aprendizado* existentes. Isso é feito principalmente a partir de propostas e/ou modificações de funcionais risco (função custo) otimizados. Uma prática comum é modificar a função custo para permitir a incorporação de diferentes custos de classificação.

Uma relação direta entre os princípios básicos das soluções que propõem mudanças nas distribuições das classes e incorporação de custos de classificação pode ser estabelecida. Para isso, considere novamente a regra de decisão (5) que minimiza o *risco* global (2). De acordo com essa regra, um novo exemplo $\mathbf{x}$ é atribuído à classe C_1 (positiva) se,

$$\frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} > \frac{\lambda_{01}}{\lambda_{10}} \frac{P(y=0)}{P(y=1)} \quad . \tag{19}$$

onde λ_{01} e λ_{10} denotam, respectivamente, os custos dos erros para a classe negativa e positiva. Observe a partir de (19), que o limiar de decisão é baseado na interação das razões entre custos e probabilidades a priori. Assim, uma vez que as prioris frequentemente refletem as proporções observadas no conjunto de treinamento, um aumento no número de exemplos de uma das classes deve causar um aumento no custo de seus erros da classificação. Como visto na Seção 3.1, essa mudança deve refletir nas probabilidades de erro obtidas para a regra de decisão estimada a partir do novo conjunto de treinamento. Da mesma forma, a incorporação direta de diferentes custos no funcional risco deve intensificar/aliviar o grau de influência causado pelo desnível das distribuições no conjunto treinamento. Alguns trabalhos na literatura têm explorado esse relacionamento para a obtenção de modelos sensíveis ao custo a partir de mudanças nas distribuições das classes (Pazzani et al., 1994; Elkan, 2001; Zhou and Liu, 2006).

Na Seção 4.1 a seguir, uma breve revisão das soluções propostas no âmbito da categoria *pré-processamento de dados* é apresentada. Em seguida, na Seção 4.2, é feita uma revisão das abordagens baseadas em *adaptações em algoritmos de aprendizado*.

4.1 Pré-Processamento de Dados

Na abordagem de *pré-processamento de dados*, o objetivo é balancear o conjunto de treinamento através de mecanismos de reamostragem de dados no espaço de entrada, que incluem *sobreamostragem* da classe minoritária, *subamostragem* da classe majoritária ou a combinação de ambas as técnicas (Japkowicz, 2000b; Laurikkala, 2001; Estabrooks et al., 2004; Batista et al., 2005).

A *sobreamostragem* é baseada na replicação de exemplos preexistentes (*sobreamostragem com substituição*) ou na geração de dados sintéticos. No primeiro caso, a seleção de exemplos a serem replicados pode ser aleatória (*sobreamostragem aleatória*) ou direcionada (*sobreamostragem informativa*). Com relação à geração de dados sintéticos, a técnica de interpolação é comumente usada. Por exemplo, no conhecido método SMOTE (*Synthetic Minority Oversampling Technique*), proposto em Chawla et al. (2002), para cada exemplo positivo $\mathbf{x}_i$, novos exemplos artificiais são criados entre os segmentos de reta que ligam $\mathbf{x}_i$ aos seus k vizinhos mais próximos.

A *subamostragem* envolve a eliminação de exemplos da classe majoritária. Os exemplos a serem eliminados podem ser escolhidos aleatoriamente (*subamostragem aleatória*) ou a partir de alguma informação a priori (*subamostragem informativa*). O algoritmo OSS (*One-Sided Selection*), proposto em Kubat and Matwin (1997), é considerado um exemplo de *subamostragem* informativa. Após selecionar um subconjunto representativo da classe majoritária e combiná-lo com todos os exemplos da classe minoritária, o algoritmo OSS

usa técnicas de limpeza (*data cleaning*) para obter *clusters* bem definidos para ambas as classes.

Apesar das técnicas de *subamostragem* e *sobreamostragem* possuírem o mesmo propósito, elas introduzem diferentes características ao novo conjunto de treinamento que podem algumas vezes, dificultar o aprendizado (Drummond and Holte, 2003; Mease et al., 2007; He and Garcia, 2009). Por exemplo, no caso de *subamostragem* aleatória, o principal problema é a *perda de informação* causada pela eliminação de exemplos representativos da classe majoritária. *Subamostragem* informativa tenta solucionar esse problema por eliminar uma fração menos representativa como, por exemplo, exemplos redundantes, ruidosos e/ou próximos à fronteira de separação entre as classes (*borderlines*). Cabe ressaltar, entretanto, que a escolha de critérios adequados para selecionar esses exemplos não é uma tarefa fácil. Grande parte dos métodos informativos usam o algoritmo KNN (*K-Nearest Neighbour*) para guiar o processo de subamostragem (Kubat and Matwin, 1997; Zhang and Mani, 2003; Batista et al., 2004). O algoritmo *BalanceCascade*, por sua vez, usa uma estratégia iterativa de geração de um *ensemble* de classificadores para a escolha dos exemplos a serem removidos (Liu et al., 2009).

Com relação a *sobreamostragem*, alguns problemas têm sido reportados. No contexto de árvores de decisão (Breiman et al., 1984), foi observado que o uso de *sobreamostragem com substituição* não melhora de forma significativa o reconhecimento da classe minoritária (Chawla et al., 2002; Mease et al., 2007). Isso ocorre devido à geração de inúmeras cláusulas em um regra para múltiplas cópias do mesmo padrão, tornando a regra muito específica. Outro problema relacionado à *sobreamostragem*, é o aumento da variância (sobreposição) causado por técnicas de geração de dados sintéticos que não consideram a vizinhança entre as classes, como é o caso do método SMOTE (He and Garcia, 2009). Para superar essa limitação, adaptações têm sido propostas para guiar o processo de interpolação adotado (Han et al., 2005; He et al., 2008). Além disso, técnicas de *data cleaning*, tais como *links de Tomek* (Tomek, 1976) e ENN (*Edited Nearest Neighbor rule*) (Wilson, 1972), têm sido sido aplicadas para reduzir o nível de ruído presente nos dados (Batista et al., 2004; Batista et al., 2005). No trabalho de Machado and Ladeira (2007b), por exemplo, uma estratégia de limpeza denominada *C-Clear* foi proposta com o objetivo de guiar o método SMOTE, diminuindo o grau de sobreposição entre as classes. Primeiramente, *C-clear* agrupa todos os exemplos de treinamento em *clusters*, os quais são rotulados como positivos (ou negativos) de acordo a frequência de exemplos minoritários/majoritários presentes. O SMOTE é então aplicado somente aos *clusters* positivos. Por último, caso seja necessário, os *clusters* são limpos através da eliminação de exemplos cuja classe original difere do rótulo do *cluster* ao qual pertencem.

As dificuldades observadas na aplicação dos métodos de reamostragem existentes motivam melhorias e o surgimento de novas estratégias. Entre as abordagens mais recentes, destacam-se pelos resultados obtidos: (i) o método BED (*Boundary Elimination and Domination Algorithm*) (Castro et al., 2009) proposto para melhorar a capacidade de discriminação de SVMs. BED usa informação sobre a densidade dos dados no espaço de entrada para eliminar exemplos ruidosos e intensificar o número de exemplos positivos junto à fronteira das classes; (ii) o algoritmo GSVM-RU (*Granular Support Vector Machines - Repetitive Undersampling*), proposto em Tang and Zhang (2006) e Tang et al. (2009), que usa as propriedades do aprendizado de SVMs como um mecanismo para *subamostragem*. A estratégia produz inúmeros grânulos de informação a partir de sucessivos treinamentos com uma SVM linear. A cada treinamento, um novo grânulo é formado a partir dos vetores de suporte negativos e removido do conjunto de treinamento original. Após a obtenção de múltiplos grânulos informativos, uma operação de agregação é usada para selecionar conjuntos específicos de amostras, que são posteriormente combinadas para desenvolver um classificador SVM final; (iii) um método híbrido composto por modelos baseados em regras e algoritmos evolucionários (Milaré et al., 2010). A abordagem cria diferentes conjuntos balanceados que são então usados na indução de classificadores cuja saída é um grupo de regras (tais como árvores de decisão). Cada conjunto balanceado contém todos os exemplos da classe minoritária e uma parcela de exemplos da classe dominante obtida através de *subamostragem aleatória*. Após a obtenção de todos os modelos, um algoritmo evolucionário é usado para fazer uma busca no espaço de regras, selecionando um subconjunto ótimo para construção do classificador final; (iv) um algoritmo genético (AGB) proposto para guiar o processo de *sobreamostragem* da classe minoritária (Beckmann and Lima, 2009; Beckmann, 2010). AGB evolui buscando pelo melhor posicionamento de regiões de *sobreamostragem* dentro dos limites mínimos e máximos definidos pelos exemplos positivos originais. Para que as classes tornem-se balanceadas, essas regiões são preenchidas proporcionalmente com exemplos sintéticos. Ao final do processo evolutivo, AGB fornece a solução que detém a melhor combinação de regiões e exemplos sintéticos de forma a maximizar a AUC (*Area Under the ROC Curve*) do classificador.

4.2 Adaptações em Algoritmos de Aprendizado

Soluções propostas nessa abordagem são baseadas na *adaptação de algoritmos de aprendizado* existentes visando me-

lhorar, ao mesmo tempo, o número de classificações positivas corretas e a acurácia geral do classificador. Devido à diversidade das soluções dessa categoria, o objetivo não foi de realizar uma revisão exaustiva, mas de prover uma análise de uma amostra representativa de trabalhos. Para facilitar a apresentação, essa amostra foi dividida em três grupos ou classes. Seguindo a discussão teórica apresentada na caracterização do problema de classes desbalanceadas (Seção 2), uma ênfase será dada ao grupo de métodos que modificam a formulação padrão do aprendizado que é baseada na minimização da taxa de erro global.

A primeira classe de soluções, conhecida como *abordagem baseada em reconhecimento*, considera somente exemplos positivos durante o processo de aprendizado com o objetivo de reconhecer (ou reconstruir) a classe de interesse (minoritária). As principais estratégias nessa linha incluem o *autoassociator* (Japkowicz, 2001; Manevitz and Yousef, 2007) e *one-class SVMs* (Schölkopf et al., 2001; Raskutti and Kowalczyk, 2004; Manevitz and Yousef, 2002; Bergamini et al., 2009). O *autoassociator* (Japkowicz, 2001) consiste em uma topologia de rede MLP com a camada de saída contendo o mesmo número de unidades da camada de entrada. O objetivo do aprendizado é reproduzir cada vetor de entrada na saída da rede. Assim, para cada exemplo positivo no conjunto de treinamento, o vetor que representa a saída desejada é o próprio vetor de entrada. Em Japkowicz (2001), o *autoassociator* foi comparado à redes MLP discriminadoras, i.e., treinadas a partir das 2 classes. Os resultados obtidos mostraram que sob certas condições como, por exemplo, em domínios multimodais, a abordagem proposta foi superior à tradicional. Schölkopf et al. (2001) propuseram um método para adaptar o algoritmo de SVMs para o aprendizado de uma única classe. Um estudo sistemático que explora o uso de *one-class SVMs* em dados sintéticos e reais foi realizado em Raskutti and Kowalczyk (2004). Os autores argumentaram que embora a abordagem tradicional (*two-class SVMs*) tenha sido superior para "conjuntos de dados comuns", o aprendizado a partir de uma única classe (positiva) é particularmente robusto na presença de ruído, esparsividade (alta dimensionalidade) do espaço de características e elevada desproporção entre as classes. Essas argumentações são reforçadas no trabalho de Bergamini et al. (2009), onde *one-class SVMs* foi empregado para a combinação de diferentes sistemas biométricos. Nos trabalhos de Manevitz and Yousef (2002; 2007), os autores exploram, respectivamente, o uso de *one-class SVMs* e *autoassociator* em tarefas de classificação de documentos, mostrando que ambas as técnicas foram efetivas.

Antes de abordar a próxima classe de soluções, é importante ressaltar que embora o escopo dessa revisão esteja limitado a métodos de aprendizado supervisionado, métodos não-supervisionados também podem ser usados para melhorar o reconhecimento da classe positiva, através de uma metodologia similar à adotada na primeira classe de soluções (*abordagem baseada em reconhecimento*). Nessa metodologia, algoritmos não-supervisionados como, por exemplo, o mapa auto-organizável (SOM) (Kohonen et al., 2001) são empregados para modelar, a partir dos dados de treinamento, somente a distribuição (densidade) da classe majoritária e, em seguida, verificar para cada novo exemplo observado se o mesmo é (ou não) oriundo dessa distribuição. Caso não seja, o exemplo é considerado como uma novidade (*outlier*) e assinalado ao grupo minoritário (Lee and Cho, 2006; Tamee et al., 2008). Essa técnica é conhecida na literatura como detecção de novidades e para maiores detalhes sobre sua implementação, recomenda-se os trabalhos de Markou and Singh (2003) e Hodge and Austin (2004).

A segunda classe de soluções é baseada em extensões do algoritmo de *Boosting*, cujo princípio básico é iterativamente atualizar uma função de distribuição para o conjunto de treinamento de forma que maior/menor ponderação seja dada aos exemplos incorretamente/corretamente classificados. A maior parte dessas extensões é realizada através da incorporação de diferentes fatores (ou funções) de custo diretamente na função de distribuição, com o objetivo de distinguir a importância entre grupos e aumentar de forma mais significativa os pesos associados aos exemplos (erros/acertos) da classe minoritária. Essa estratégia, conhecida na literatura como *Cost-Sensitive Boosting*, permite o uso de amostras mais relevantes no treinamento das hipóteses (*weak-learners*), visando a obtenção de uma regra de decisão final que dá mais importância à classe de interesse. Baseados nessa metodologia, métodos *Cost-Sensitive Boosting* têm sido propostos, tais como AdaCost (Fan et al., 1999), CSB1 e CSB2 (Ting, 2000) e, AdaC1, AdaC2 e AdaC3 (Sun et al., 2007). Um estudo empírico envolvendo a aplicação desses métodos a vários problemas reais de diagnóstico médico foi conduzido por Sun et al. (2007). Usando árvores de decisão como classificadores base, os autores investigaram os algoritmos com relação a suas diferentes estratégias de ponderação e mostraram que os mesmos foram efetivos em melhorar a identificação da classe positiva. Outra extensão do método de *Boosting* foi apresentada em Rodrigues et al. (2009). No algoritmo B-Boost, proposto nesse trabalho, a seleção de exemplos mais relevantes (a cada iteração) é realizada separadamente por classe. Essa pequena modificação permite a obtenção de conjuntos de treinamento balanceados contendo os exemplos mais difíceis de cada classe. Tais conjuntos são então usados na indução das hipóteses (*weak-learners*). A eficiência do B-Boost em problemas desbalanceados foi comprovada através da execução de experimentos com 20 bases de dados do repositório UCI (Asuncion and Newman, 2007).

Finalmente, a última classe de soluções dessa categoria está relacionada a propostas e/ou modificações de funcionais risco (função custo). Assumindo a premissa de custos iguais para os erros de classificação, a maioria dos algoritmos de aprendizado existentes são projetados para minimizar o erro global sobre o conjunto de treinamento. Modificações nesse critério, com o objetivo de obter regras de decisão que melhoram o reconhecimento da classe minoritária, têm sido propostas de diferentes formas. A estratégia que tem sido mais usada é considerar a divisão do erro global entre as classes e incorporar funções de penalidade (ou fatores custo) distintas aos diferentes tipos de classificação. Essa técnica é comumente conhecida como *Abordagem Sensível ao Custo* e segue o princípio de minimização do custo esperado (*risco* global) da Teoria de Decisão Bayesiana, apresentada na Seção 2.1. Outras soluções, particularmente no contexto de máquinas de *kernel* (Muller et al., 2001), que envolvem modificações no espaço de características induzido, tais como deslocamento do hiperplano ou aumento da resolução espacial dos exemplos positivos, influenciam diretamente o critério de decisão adotado. Além disso, a divisão do erro global entre as classes tem permitido uma formulação multiobjetivo para o problema do aprendizado.

Nessa revisão, uma maior atenção é dedicada às soluções dessa última classe e assim, uma descrição detalhada dos principais trabalhos propostos no âmbito de máquinas de *kernel* e Redes Neurais Artificiais (RNAs) é fornecida nas seções a seguir. Para facilitar o entendimento do leitor, são apresentadas aqui as principais notações usadas para descrever os métodos. Seja um conjunto de treinamento $T = \{\mathbf{x}_i, \mathbf{y}_i\}_{i=1}^{N}$ consistindo de N exemplos pertencentes a duas classes, onde $\mathbf{y}_i \in \mathbf{Y}$ denota o rótulo para cada vetor de entrada $\mathbf{x}_i \in \mathbb{R}^n$. A natureza do conjunto $\mathbf{Y}$ é dependente da convenção adotada pelo algoritmo de aprendizado. SVMs e outras máquinas de kernel, por exemplo, frequentemente assumem $Y = \{-1, 1\}$ e, assim, y_i torna-se uma simples variável simbólica. Quando necessário, a natureza de $\mathbf{Y}$ será especificada durante a descrição do algoritmo. Considere também que existem N_1 exemplos da classe positiva ou minoritária, T_1 e, N_0 exemplos da classe negativa ou majoritária, T_0. Vetores de entrada arbitrários pertencentes às classes positiva e negativa são denotados, respectivamente, por $\mathbf{x}^1$ e $\mathbf{x}^0$; como antes, λ_{01} e λ_{10} representam os custos referentes aos falsos positivos e falsos negativos, respectivamente.

4.2.1 SVMs com Custos Assimétricos

No contexto de SVMs, Veropoulos et al. (1999) distinguiram os erros entre as classes positiva e negativa através da introdução de diferentes parâmetros de regularização: C^1 e C^0. Assumindo $y_i \in \{-1, 1\}$, os autores propõem a seguinte modificação na função custo do problema primal de SVMs com margens suaves (Cortes and Vapnik, 1995),

$$min_{(\mathbf{w},b,\varepsilon_i)} \quad \frac{1}{2}\|\mathbf{w}\|^2 + C^1 \sum_{i\in T_1} \varepsilon_i + C^0 \sum_{i\in T_0} \varepsilon_i \ . \quad (20)$$

$$s.a. \quad y_i\left(\langle \mathbf{w}\cdot\mathbf{x}_i\rangle + b\right) \geq 1 - \varepsilon_i, \ \forall \ i \in T \ .$$
$$\varepsilon_i \geq 0, \ \forall \ i \in T \ .$$

onde $\mathbf{w}$ e b correspondem aos parâmetros do hiperplano ($\langle \mathbf{w}\cdot\mathbf{x}\rangle + b = 0$) em algum espaço de características $\mathbb{F}$ e as variáveis de folga ε_i são introduzidas para permitir erros de classificação. A formulação dual equivalente é dada por,

$$max_{(\alpha)} \quad \sum_{i=1}^{N} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{N} y_i y_j \alpha_i \alpha_j K(\mathbf{x}_i, \mathbf{x}_j) \ . \quad (21)$$

$$s.a. \quad 0 \leq \alpha_i \leq C^1, \ \forall \ i \in T_1 \ . \quad (22)$$
$$0 \leq \alpha_i \leq C^0, \ \forall \ i \in T_0 \ . \quad (23)$$
$$\sum_{i=1}^{N} \alpha_i y_i = 0 \ . \quad (24)$$

onde $K(\mathbf{x}, \mathbf{x}')$ representa a função de *kernel*. Resolvendo o problema dual, os multiplicadores de *lagrange* α_i, cujos tamanhos são limitados por C^1 e C^0, são estimados; o parâmetro b pode ser obtido a partir de algum exemplo $\mathbf{x}_i$ com α_i não nulo (vetor de suporte). A classificação de um exemplo arbitrário $\mathbf{x}_j$ é dada pela seguinte regra de decisão (mesma regra da SVM original),

$$\text{sgn}\left(\sum_{i=1}^{N} y_i \alpha_i K(\mathbf{x}_i, \mathbf{x}_j) + b\right) \ . \quad (25)$$

A idéia básica do método é compensar o desbalanceamento do conjunto de dados a partir do ajuste da razão $\frac{C^1}{C^0}$. Segundo Veropoulos et al. (1999), se $\frac{C^1}{C^0} > 1$, a estratégia permite aumentar a influência dos vetores de suporte da classe positiva, desde que valores maiores de α_i são obtidos para os exemplos positivos, conforme condições de *KKT* (*karush-kuhn-tucker*) dadas por (22) e (23). Isso faz com que a superfície de decisão fique mais distante da classe minoritária e consequentemente, o número de falsos negativos diminua.

Com o objetivo de equilibrar os custos das classes positiva e negativa, Morik et al. (1999) e Joachims (2002) propuseram que a razão $\frac{C^1}{C^0}$ seja igual a $\frac{N_0}{N_1}$. Em Lin et al. (2002) foi adotada uma estratégia diferente para o ajuste de C^1 e C^0 que além de considerar custos desiguais (para falsos positivos e falsos negativos) também considera *viés de amostragem*. Segundo os autores, *viés de amostragem* ocorre quando

os exemplos não são amostrados de uma maneira completamente aleatória, fazendo com que as proporções de positivos e negativos no conjunto de treinamento não correspondam às atuais proporções na população alvo. Assim, a seguinte técnica para o ajuste de C^1 e C^0 foi proposta,

$$\begin{aligned} C^1 &= \lambda_{10}\,\hat{\pi}_0\,\pi_1\ . \\ C^0 &= \lambda_{01}\,\hat{\pi}_1\,\pi_0\ . \end{aligned}$$

onde $\hat{\pi}_1$ e $\hat{\pi}_0$ correspondem, respectivamente, às proporções de exemplos positivos e negativos no conjunto de treinamento e π_1 e π_0 são essas proporções (probabilidades a priori) na população alvo, na qual a SVM deve ser aplicada.

Críticas à eficiência das AC-SVM (*SVMs com Custos Assimétricos*) foram feitas em Wu and Chang (2003). Baseados nas condições de *KKT*, os autores argumentaram que a restrição (24) impõe equilíbrio na influência total dos vetores de suporte de cada classe. Para que a restrição seja satisfeita, um aumento nos valores de α_i para exemplos positivos também deve acarretar um aumento nos valores de α_i para exemplos negativos. Apesar disso, a estratégia tem apresentado bons resultados em aplicações reais desbalanceadas. Em Akbani et al. (2004), os autores sugerem a combinação de *SMOTE* + AC-SVM, obtendo bons resultados sobre bases de dados desbalanceadas do repositório UCI (Asuncion and Newman, 2007). Recentemente, uma estratégia de *Boosting* que sequencialmente usa AC-SVM como classificadores base foi proposta em Wang and Japkowicz (2008). A cada iteração, um novo conjunto de dados com pesos modificados é aplicado a um AC-SVM com $\frac{C^1}{C^0} = \frac{N_0}{N_1}$. Ao final do processo, as saídas dos classificadores componentes são então combinadas, por um esquema de voto majoritário ponderado, para produzir uma predição final.

4.2.2 SVMs com Margens Desiguais

Em Karakoulas and Shawe-Taylor (1999) foi proposta uma estratégia para diferenciar o tamanho das margens (positiva e negativa) no treinamento de SVMs. Isso pode ser obtido a partir da incorporação do parâmetro τ nas restrições de desigualdade referentes aos exemplos da classe positiva. A formulação do problema primal de SVMs com margens suaves é dada por,

$$\begin{aligned} min_{(\mathbf{w},b,\varepsilon_i)} \quad & \frac{1}{2}\left\|\mathbf{w}\right\|^2 + C\sum_{i=1}^{N}\varepsilon_i\ . \qquad (26) \\ s.a. \quad & y_i\left(\langle\mathbf{w}\cdot\mathbf{x}_i\rangle + b\right) \geq \tau - \varepsilon_i,\ \forall\, i \in T_1\ . \\ & y_i\left(\langle\mathbf{w}\cdot\mathbf{x}_i\rangle + b\right) \geq -1 + \varepsilon_i,\ \forall\, i \in T_0\ . \\ & \varepsilon_i \geq 0,\ \forall\, i \in T\ . \end{aligned}$$

onde $\tau > 1$ corresponde à razão entre a margem positiva e negativa, ou seja, $\tau = \frac{\rho_1}{\rho_0}$. O efeito obtido com o método é o deslocamento paralelo do hiperplano obtido no espaço de características de forma que a margem positiva fique τ vezes maior que a margem negativa.

Os autores também mostraram que o mesmo efeito pode ser obtido a partir da solução do problema original proposto para SVMs (Cortes and Vapnik, 1995), seguido de uma simples mudança no cálculo do parâmetro b (*threshold*) do hiperplano de separação,

$$b = \frac{1}{1+\tau}[\langle\mathbf{w}\cdot\mathbf{x}^1\rangle + \tau\left\langle\mathbf{w}\cdot\mathbf{x}^0\right\rangle]\ . \qquad (27)$$

onde $\mathbf{x}^1$ e $\mathbf{x}^0$ correspondem, respectivamente, a vetores de suporte arbitrários da classe positiva e negativa.

Idéia similar foi apresentada em Li and Shawe-Taylor (2003) porém, a incorporação do parâmetro τ ocorre nas restrições de desigualdade referentes aos exemplos da classe negativa. Nesse trabalho, os autores mostraram a eficiência do método em problemas de categorização de textos que, em geral, são altamente desbalanceados.

4.2.3 Mudanças no *Kernel*

Ainda no contexto de SVMs, Wu and Chang (2003; 2005) sugerem duas abordagens para modificar o *kernel* empregado considerando a distribuição dos dados como informação a priori. O primeiro algoritmo, *Adaptive Conformal Transformation* (ACT) (Wu and Chang, 2003), modifica a função de *kernel* K no espaço de entrada $\mathbb{I}$ e, portanto, depende que os dados possuam uma representação vetorial de dimensão fixa. O segundo, denominado *Kernel Boundary Alignment* (KBA) (Wu and Chang, 2004; Wu and Chang, 2005), modifica diretamente a matriz de *Kernel* $\mathbf{K}$ no espaço de características $\mathbb{F}$, podendo lidar com dados de diferentes dimensões (sequências de DNA, vídeos de monitoramento, etc.).

A idéia básica em ambos os métodos é aumentar o valor da métrica de *Riemann* para dados próximos à fronteira de separação entre as classes. Segundo os autores, a métrica de *Riemann* associada à função de *kernel* $K\left(\mathbf{x},\mathbf{x}'\right)$, mede como

uma área local ao redor de $\mathbf{x}$ em $\mathbb{I}$ é aumentada em $\mathbb{F}$ a partir do mapeamento imposto por $\Phi(\mathbf{x})$. No algoritmo ACT, isso é obtido através de uma transformação *conformal* da função de *kernel* $K(\mathbf{x}, \mathbf{x}')$,

$$\tilde{K}(\mathbf{x}, \mathbf{x}') = D(\mathbf{x})D(\mathbf{x}')K(\mathbf{x}, \mathbf{x}') \quad . \tag{28}$$

onde $D(\mathbf{x})$ é uma função positiva definida que deve ser escolhida para que a nova métrica de *Riemann* associada à nova função $\tilde{K}(\mathbf{x}, \mathbf{x}')$ possua valores maiores em regiões próximas à fronteira de decisão entre as classes. Além disso, para obter uma superfície de decisão mais distante da classe minoritária, os autores propõem que a métrica de *Riemann* seja aumentada de forma mais intensa em regiões próximas à margem da classe positiva. Para isso, eles sugerem o uso de uma família de funções gaussianas para $D(\mathbf{x})$,

$$D(\mathbf{x}) = \sum_{k=1}^{N_{sv}} \exp\left(-\frac{|x - x_k|}{\sigma_k^2}\right) \quad . \tag{29}$$

na qual N_{sv} representa o número total de vetores de suporte (*s.v.*'s) e o parâmetro de largura σ_k^2, deve ser calculado para cada *s.v.* segundo a distribuição espacial de sua vizinhança no espaço de características $\mathbb{F}$. Para detalhes de como esse cálculo é feito veja Wu and Chang (2003). Diferentes fatores são então multiplicados ao parâmetro σ_k^2, dependendo se k corresponde a um *s.v.* positivo ou negativo,

$$\begin{cases} \sigma_k^2 \Leftarrow \eta_1 \, \sigma_k^2 & \text{se } k \text{ é um } \textit{s.v.} \text{ positivo,} \\ \sigma_k^2 \Leftarrow \eta_0 \, \sigma_k^2 & \text{se } k \text{ é um } \textit{s.v.} \text{ negativo} \end{cases} \tag{30}$$

onde $\eta_1 = \frac{N_{SV}^0}{N_{SV}^1}$ e $\eta_0 = \frac{N_{SV}^1}{N_{SV}^0}$ com, N_{SV}^1 e N_{SV}^0, representando os números de vetores de suporte das classes positiva e negativa, respectivamente. Esse ajuste intensifica a resolução espacial em regiões próximas aos *s.v.*'s positivos. Após a obtenção da função transformada $\tilde{K}(\mathbf{x}, \mathbf{x}')$, um novo treinamento permite estimar uma regra de decisão com melhor capacidade discriminativa.

No algoritmo KBA, os autores adotam a estratégia de aumentar a resolução espacial junto a um hiperplano de separação considerado "ideal". Eles partem da hipótese de que, quando o conjunto de dados é desbalanceado, o hiperplano de margem máxima obtido pelas SVMs é desviado em direção à classe minoritária. Assim, a superfície de separação "ideal" deve ficar entre esse hiperplano (central) e o hiperplano representado pela margem da classe majoritária. A localização de um exemplo arbitrário no hiperplano "ideal" é obtida a partir do seguinte procedimento de interpolação, que considera um *s.v.* positivo $\Phi(\mathbf{x}^1)$ e um *s.v.* negativo $\Phi(\mathbf{x}^0)$ no espaço de características $\mathbb{F}$,

$$\Phi(\mathbf{x}_b) = (1 - \beta)\, \Phi(\mathbf{x}^1) + \beta\, \Phi(\mathbf{x}^0), \;\; \frac{1}{2} \leq \beta \leq 1 \quad . \tag{31}$$

O parâmetro β fornece indiretamente a localização do hiperplano "ideal" em $\mathbb{F}$. Seu valor ótimo é obtido a partir da minimização de uma função custo que mede a perda causada por falsos positivos e falsos negativos (vide Wu and Chang (2005) para detalhes). O próximo passo é aumentar a métrica de *Riemann* ao redor do hiperplano "ideal". Para tanto, os autores sugerem $D(\mathbf{x})$ como uma família de gaussianas,

$$D(\mathbf{x}) = \frac{1}{N_b} \sum_{k=1}^{N_b} \exp\left(-\frac{||\Phi(\mathbf{x}) - \Phi(\mathbf{x}_b)||}{\sigma_b^2}\right) \quad . \tag{32}$$

onde σ_b^2 representa a largura da gaussiana associada a um dado *s.v.* interpolado $\Phi(\mathbf{x}_b)$ e, N_b corresponde ao número de s.v.'s interpolados ao longo do hiperplano "ideal". Para um exemplo arbitrário $\mathbf{x}$, $D(\mathbf{x})$ é calculado como a média dessas gaussianas. Desde que o mapeamento $\Phi(\mathbf{x})$ é desconhecido, $||\Phi(\mathbf{x}) - \Phi(\mathbf{x}_b)||$ pode ser obtido por,

$$\begin{aligned} ||\Phi(\mathbf{x}) - \Phi(\mathbf{x}_b)|| &= ||\Phi(\mathbf{x}) - (1 - \beta)\, \Phi(\mathbf{x}^1) - \beta\, \Phi(\mathbf{x}^0)|| \\ &= k_{xx} + (1 - \beta)^2\, k_{x^1x^1} + \beta^2\, k_{x^0x^0} \\ &\quad - 2(1 - \beta)\, k_{xx^1} - 2\beta\, k_{xx^0} \\ &\quad + 2\beta\,(1 - \beta)\, k_{x^1x^0} \quad . \end{aligned}$$

onde $k_{xx'}$ é extraído diretamente da matriz de *kernel* $\mathbf{K}$. Cada elemento de $\mathbf{K}$ é então modificado a partir da transformação conformal descrita a seguir,

$$\tilde{k}_{ij} = D(\mathbf{x}_i)\, D(\mathbf{x}_j)\, k_{ij} \quad . \tag{33}$$

A nova matriz $\tilde{\mathbf{K}}$ obtida é novamente usada pelo algoritmo de treinamento original. Em Wu and Chang (2005), os autores testaram a eficiência de seus métodos em bases desbalanceadas do repositório UCI e obtiveram bons resultados.

Outro algoritmo baseado em modificação do *kernel* foi proposto em Kandola and Shawe-Taylor (2003). Os autores sugeriram uma extensão do algoritmo *Kernel Target Alignment* (Cristianini et al., 2002), atribuindo *targets* de alinhamento de $\frac{1}{N^0}$ para exemplos positivos e $-\frac{1}{N^1}$ para exemplos negativos. Foi observado, no entanto, que o algoritmo não foi eficiente para conjuntos de dados com elevado grau de desbalanceamento.

4.2.4 *Orthogonal Forward Selection*

Hong et al. (2007) apresentaram um novo método para a construção de classificadores binários baseados em *kernels* que, segundo resultados empíricos, têm mostrado bom desempenho em aplicações desbalanceadas. Para tanto, os autores propuseram modificações nos critérios de estimação de parâmetros e seleção de modelos do algoritmo *Orthogonal Forward Selection* (OFS) (Chen et al., 2006).

A cada passo do algoritmo OFS, o método *Regularized Orthogonal Weighted Least Squares* (ROWLS) é usado para estimar os parâmetros dos modelos candidatos através de uma nova função custo que distingue os erros obtidos para cada classe,

$$J = \lambda \sum_{i \in T_0} e_i^2 + \sum_{i \in T_1} e_i^2 \ . \tag{34}$$

onde o erro obtido na saída do classificador para um exemplo arbitrário $\mathbf{x}_i$ é dado por $e_i = y_i - f_i$, com $y_i \in \{-1, 1\}$. O parâmetro de custo $\lambda > 1$, que deve ser escolhido pelo usuário, é usado para atribuir maior peso aos exemplos da classe minoritária; λ tem o efeito de mover o hiperplano para longe da classe minoritária, garantindo que os modelos candidatos sejam apropriados para aplicações desbalanceadas.

Para a seleção do melhor modelo entre os candidatos, os autores propuseram o critério *Leave-One-Out Area Under the ROC Curve* (LOO-AUC). Segundo esse critério, para um dado modelo candidato, os parâmetros são estimados com os $N-1$ exemplos do conjunto de treinamento, e o exemplo restante é usado como validação, sendo a saída do classificador para esse exemplo denotada por $f_i^{(-i)}$. A AUC é então calculada através das saídas de validação obtidas a partir do *LOO-crossvalidation*, através da seguinte Equação,

$$AUC^{(-)} = \frac{1 + TP^{(-)} - FP^{(-)}}{2} \ . \tag{35}$$

onde,

$$\begin{aligned} TP^{(-)} &= \frac{1}{N_1} \sum_{i=1}^{N} IdT(f_i^{(-i)} \times y_i,\ y_i) \ . \\ FP^{(-)} &= \frac{1}{N_0} \sum_{i=1}^{N} IdF(f_i^{(-i)} \times y_i,\ y_i) \ . \end{aligned}$$

na quais as funções indicadoras $IdT(u, v)$ e $IdF(u, v)$ são definidas por,

$$IdT(u, v) = \begin{cases} 1 & \text{se } u \geq 0 \text{ e } v = 1, \\ 0 & \text{caso contrário.} \end{cases} \tag{36}$$

$$IdF(u, v) = \begin{cases} 1 & \text{se } u \leq 0 \text{ e } v = -1, \\ 0 & \text{caso contrário.} \end{cases} \tag{37}$$

Segundo os autores, o critério LOO-AUC não é caro computacionalmente, desde que o método ROWLS possui fórmulas recursivas que algebricamente implementam *LOO-crossvalidation* sem a necessidade de dividir o conjunto de treinamento.

4.2.5 Redes Neurais Sensíveis ao Custo

A estimação de parâmetros de modelos neurais *feed-forward* é comumente obtida através da minimização do funcional somatório dos erros quadráticos, que considera custos uniformes para os diferentes erros de classificação. Essa estratégia tem sido adotada pelos inúmeros algoritmos de aprendizado desenvolvidos para as topologias *Multilayer Perceptron* (MLP) e *Radial Basis Function* (RBF) desde a introdução do algoritmo *Backpropropagation* padrão (Rumelhart and McClelland, 1986). Como visto anteriormente, a premissa de custos uniformes pode prejudicar o aprendizado do grupo que possui menos exemplos no conjunto de treinamento. Além disso, estudos empíricos mostraram que a velocidade de convergência do algoritmo *Backpropropagation* fica comprometida se os grupos são muito desbalanceados (Anand et al., 1993). Para aliviar esses problemas, mudanças na função custo original têm sido propostas, principalmente a partir da introdução de funções de penalidade associadas aos diferentes tipos de erro (Kukar and Kononenko, 1998; Alejo et al., 2006; Castro and Braga, 2009).

O trabalho de Kukar and Kononenko (1998) considera uma rede MLP com codificação 0 de $c-1$, onde c é o número de unidades de saída (classes). Nessa codificação, dado um vetor de entrada $\mathbf{x}_i$ pertencente à classe T_k, o rótulo $\mathbf{y}_i$ associado é um vetor cujo j-ésimo componente $y_i^{(j)} = \delta_{jk}$, onde δ_{jk} é o símbolo delta de *kronecker* definido como: $\delta_{jk} = 1$ se $k = j$ e $\delta_{jk} = 0$ se $k \neq j$, para $k, j = 0, \ldots, c-1$. Usando essa notação, os autores propuseram uma modificação no funcional somatório dos erros quadráticos através da incorporação do fator $\zeta(k, j)$, com k representando a classe desejada (correta) para o i-ésimo exemplo de treinamento e j a classe atual,

$$J = \frac{1}{2} \sum_{i=i}^{N} \sum_{j=0}^{c-1} \left([y_i^{(j)} - f_i^{(j)}]\, \zeta(k, j) \right)^2 \ . \tag{38}$$

onde $y_i^{(j)}$ e $f_i^{(j)}$ correspondem, respectivamente, às saídas desejada e obtida no j-ésimo neurônio de saída devido à apresentação do exemplo $\mathbf{x}_i$; A definição do fator $\zeta(k,j)$ é baseada nos custos λ_{kj} associados aos erros de classificação e, depende de dois aspectos:

- se o neurônio de saída j corresponde à classe correta k do i-ésimo exemplo de treinamento, então a diferença $y_i^{(j)} - f_i^{(j)}$ pode ser interpretada como a probabilidade de classificar o exemplo $\mathbf{x}_i$ em qualquer uma das $c-1$ classes incorretas. Essa probabilidade deve ser ponderada pelo custo esperado do erro para a classe k, descrito pela Equação (39) a seguir.
- para os demais neurônios j, que não correspondem à classe correta k do i-ésimo exemplo, a diferença $y_i^{(j)} - f_i^{(j)}$ pode ser interpretada como a probabilidade de classificar o exemplo $\mathbf{x}_i$ na classe j dado $\mathbf{x}_i$ pertence à k. Nesse caso, ela deve ser ponderada pelo custo λ_{kj}.

$$\zeta(k,j) = \begin{cases} \frac{1}{1-\pi_k} \sum_{\forall\, j \neq k} \pi_j\, \lambda_{kj} & \text{se } k = j, \\ \lambda_{kj} & \text{se } k \neq j. \end{cases} \tag{39}$$

na qual π_k é a probabilidade a priori da classe k.

Em Alejo et al. (2006), uma mudança na função custo original foi proposta para redes RBF. Adotando a codificação 0 de $c-1$ na camada de saída, os autores consideram as contribuições dos erros quadráticos obtidos para cada classe e introduzem a função de perda $\gamma(\cdot)$ para compensar o desbalanceamento,

$$J = \frac{1}{N}\left[\sum_{k=0}^{c-1} \gamma(k) \sum_{i\,\in\, T_k} \sum_{j=0}^{c-1} \left([y_i^{(j)} - f_i^{(j)}]\right)^2\right] \quad . \tag{40}$$

com $\gamma(k) = \frac{\max(N_j)}{N_k}$ representando o custo associado ao se cometer erros para a classe k; $\max(N_j)$ corresponde ao número de exemplos da maior classe. Segundo Alejo et al. (2006), a função $\gamma(\cdot)$ tem o efeito de equilibrar a magnitude (norma euclidiana) dos vetores gradiente obtidos para cada classe, acelerando a convergência e evitando que classes muito pequenas sejam ignoradas no treinamento realizado com o algoritmo *Backpropropagation*.

Limitando o escopo a problemas contendo somente 2 classes, trabalho recente de Castro and Braga (2009) considera redes MLP com uma única unidade de saída, tal que $y_i \in \{-1, 1\}$. Nesse contexto, os autores propuseram uma função custo conjunta descrita pela soma ponderada dos erros quadráticos obtidos para as classes negativa e positiva, respectivamente,

$$J = \frac{1}{2}\left[\frac{1}{\lambda_{01}} \sum_{i\,\in\, T_0} (y_i - f_i)^2 + \frac{1}{\lambda_{10}} \sum_{i\,\in\, T_1} (y_i - f_i)^2\right] \tag{41}$$

onde os fatores de custo $\lambda_{01} \geq 1$ e $\lambda_{10} \geq 1$, são usados para definir os pesos das contribuições dos erros de cada classe na composição de J. Note que quando λ_{01} e λ_{10} assumem valores iguais a 1, a expressão (41) se reduz ao funcional somatório dos erros quadráticos sobre todo o conjunto T.

Em Castro and Braga (2009) foi também conduzida uma análise detalhada sobre o papel dos parâmetros λ_{01} e λ_{10} mostrando que a razão $\lambda_{01}/\lambda_{10}$ influencia diretamente a localização da superfície de decisão estimada. Segundo os autores, regras de decisão com taxas de acerto aproximadamente equilibradas podem ser obtidas ajustando essa razão de acordo com os números de exemplos das classes, i.e., $\lambda_{01}/\lambda_{10} = N_0/N_1$.

4.2.6 Abordagem Multiobjetivo

Com o objetivo de otimizar a curva ROC para classificadores binários baseados em redes MLP, alguns trabalhos na literatura (Kupinski and Anastasio, 1999; Sanchez et al., 2005; Everson and Fieldsend, 2006a; Graening et al., 2006), formularam o problema do aprendizado como um problema de otimização multiobjetivo, da seguinte forma,

$$\arg_{\omega}\ \max\,(\min) \begin{cases} J_0(\omega) \\ J_1(\omega) \end{cases} \tag{42}$$

onde ω é conjunto de parâmetros (pesos) e as funções custo $J_0(\omega)$ e $J_1(\omega)$ correspondem a métricas extraídas da matriz confusão que medem o desempenho obtido pela rede para as classes T_0 e T_1, respectivamente. Em Kupinski and Anastasio (1999), os autores usaram $J_1(\omega) = TPr(\omega)$ e $J_0(\omega) = TNr(\omega)$; em Sanchez et al. (2005) e Everson and Fieldsend (2006a) foram adotados $J_1(\omega) = FNr(\omega)$ e $J_0(\omega) = FPr(\omega)$; e, no trabalho de Graening et al. (2006), foram sugeridos $J_1(\omega) = TPr(\omega)$ e $J_0(\omega) = FPr(\omega)$.

Em todos os trabalhos, algoritmos evolucionários multiobjetivo foram escolhidos para solucionar o problema (42). Ao final do processo de aprendizado, os algoritmos retornam uma estimativa para o conjunto de soluções não dominadas[5] denominado conjunto Pareto-ótimo. Todas as soluções são

[5] Em um problema de otimização multiobjetivo, uma solução é dita ser não dominada, se não existe nenhuma solução com desempenho superior a ela em todos os objetivos.

equivalentes na ausência de qualquer informação referente aos objetivos $J_0(\omega)$ e $J_1(\omega)$ e podem ser interpretadas como pontos de operação de uma curva ROC ótima.

Nos trabalhos supracitados não foi proposta nenhuma estratégia de decisão para a escolha de uma solução (ou ponto de operação) no conjunto Pareto-ótimo. Os autores deixam a cargo do usuário escolher a solução cujo desempenho seja mais apropriado para a tarefa de aprendizado em questão.

Ainda dentro da abordagem multiobjetivo, um algoritmo denominado *Pareto Front Elite* (Ishida and Pozo, 2007) foi proposto para seleção de um subconjunto de regras de classificação que otimiza a AUC (*Area Under the ROC Curve*). *Pareto Front Elite* funciona como uma estratégia de pós-processamento. A partir de um grande conjunto de regras gerado por algum algoritmo de associação, o subconjunto ótimo é selecionado através de testes de não dominância baseados nos critérios TPr (sensibilidade) e TNr (especificidade).

5 DISCUSSÕES E CONCLUSÕES

Esse trabalho teve como objetivo fornecer uma investigação sobre o problema de classe desbalanceadas com foco na abordagem discriminativa do aprendizado supervisionado. Foram descritos aspectos associados à natureza do problema e métricas de avaliação, incluindo os fundamentos da análise ROC. O estado da arte das soluções foi apresentado, com ênfase nas abordagens que modificam a formulação padrão adotada por algoritmos de aprendizado tradicionais.

Uma importante conclusão desse trabalho foi que, embora avanços em aprendizado com dados desbalanceados tenham sido obtidos, especialmente no âmbito das soluções propostas, algumas questões continuam em aberto ou não foram completamente resolvidas. Existe uma carência por estudos teóricos (ou empíricos), fundamentados nas Teorias do Aprendizado, que permitam um melhor entendimento das causas e consequências do problema. Tais estudos são essenciais para justificar e guiar o desenvolvimento de soluções. A análise descrita na Seção 2 foi desenvolvida para contribuir um pouco nesse sentido. Foi demonstrado, no contexto de modelos discriminativos, que o viés imposto pelo grupo dominante é uma consequência direta da minimização de um critério baseado no erro global, tendo como principal atenuante o nível de incerteza (ruído) da tarefa de classificação.

Existem ainda outras características relacionadas ao problema de classes desbalanceadas que têm sido observadas, especialmente no contexto de classificadores baseados em árvores de decisão, e que necessitam de mais investigações. Entre elas, destacam-se aspectos associados à falta de representatividade do grupo minoritário, tais como o desbalanceamento entre *subclusters* pertencentes à uma mesma classe e o problema de *small disjuncts* (Japkowicz, 2003; Jo and Japkowicz, 2004; Prati et al., 2004a; Machado, 2007; Machado and Ladeira, 2007a). Com base nas informações levantadas até o momento, entende-se que soluções promissoras para o problema deveriam tratar ambos fatores: viés causado pela diferença entre as probabilidades a priori das classes e a falta de representatividade da distribuição minoritária.

Outra questão importante está relacionada às soluções propostas para o problema. Abordagens de *pré-processamento de dados* frequentemente são usadas para balancear o conjunto de treinamento visando aumentar o número de exemplos positivos corretos e diminuir a discrepância entre as taxas de acertos das classes. Alguns trabalhos, no entanto, têm argumentado que nem sempre a distribuição balanceada produz os melhores resultados (Weiss and Provost, 2003; Estabrooks et al., 2004; Prati et al., 2008b). Nesse contexto, surge uma questão importante: Dado um problema desbalanceado, qual é a proporção (razão) ideal entre o número de exemplos das classes no conjunto de treinamento para a maximização do desempenho do classificador? Os trabalhos de Weiss and Provost (2003) e Prati et al. (2008b) tentaram responder a essa pergunta analisando os desempenhos obtidos para diferentes razões de desbalanceamento em inúmeras bases de dados reais. Com base nos resultados obtidos, ambos os estudos sugeriram que se a *AUC* é selecionada como métrica de desempenho, a melhor distribuição fica próxima da balanceada. Por outro lado, Weiss and Provost (2003) observaram que se a *acurácia* é escolhida, a melhor proporção tende a ser próxima da distribuição natural dos dados.

Problema correlato pode ser visto no escopo dos métodos da *abordagem Sensível ao Custo* que incorporam parâmetros (ou funções) de custo (representados nesse estudo por λ_{10} e λ_{01}) à formulação padrão do problema de aprendizado. Embora na maioria desses métodos, exista somente a recomendação para que o custo associado ao grupo minoritário (λ_{10}) seja maior que o custo associado ao grupo dominante (λ_{01}), o ajuste adequado (ideal) para a razão $\lambda_{10}/\lambda_{01}$ é desconhecido e ainda considerado um problema em aberto.

Com base em resultados reportados na literatura (Weiss and Provost, 2003; Tang et al., 2009; Castro and Braga, 2009), especula-se que os ajustes adequados para as proporções (razões) entre as classes e custos, devem ser dependentes da medida (ou critério) de desempenho que se deseja maximizar/minimizar. Sob esse ponto de vista, procedimentos de otimização poderiam ser incorporados aos algoritmos de aprendizado para efetuar uma busca dos valores dessas razões (classes e custos) segundo a métrica selecionada: *acurácia*, *G-mean*, *F-measure*, *AUC*, etc. Além disso, uma alternativa promissora nesse campo, é propor novos algoritmos de aprendizado baseados em funcionais risco específicos

para otimizar diretamente a métrica desejada, conforme apresentado nos trabalhos de Joachims (2005), Herschtal et al. (2006) e, Castro and Braga (2008).

Ainda sobre a questão do ajuste dos parâmetros, alguns métodos apresentados na Seção 4.2 têm proposto um ajuste para a razão entre os custos de acordo com o inverso da razão entre os números de exemplos das classes, i.e., $\frac{\lambda_{10}}{\lambda_{01}} = \frac{N_0}{N_1}$. Analisando essa estratégia com base nos fundamentos teóricos apresentados na Seção 2, é possível especular sobre as propriedades das soluções buscadas por esses métodos. Note que a incorporação dos parâmetros de custo à formulação original desses algoritmos, que é baseada na probabilidade do erro global de classificação (6), leva a um novo funcional *risco*, dado pela seguinte expressão,

$$\begin{aligned} R &= \lambda_{01} \int_{\mathbf{R}_1} p(\mathbf{x}|y=0)P(y=0)\, d\mathbf{x} \\ &\quad +\lambda_{10} \int_{\mathbf{R}_0} p(\mathbf{x}|y=1)P(y=1)\, d\mathbf{x} \ . \end{aligned} \quad (43)$$

Substituindo em (43) as probabilidades a priori $P(y = k)$ pelas proporções de exemplos N_k/N no conjunto de treinamento e, usando a sugestão proposta: $\lambda_{10} = N_0$ e $\lambda_{01} = N_1$; é possível mostrar que a solução ótima f^* que minimiza o novo funcional (43) é aquela que atribui cada exemplo de entrada $\mathbf{x}$ à classe C_k para o qual a densidade condicional $p(\mathbf{x}|y = k)$ é maior, i.e.,

$$f^*(\mathbf{x}) = \begin{cases} 1 & \text{se } \frac{p(\mathbf{x}|y=1)}{p(\mathbf{x}|y=0)} > 1, \\ 0 & \text{caso contrário.} \end{cases} \quad (44)$$

Observe a partir de (44), que a estratégia que ajusta $\frac{\lambda_{10}}{\lambda_{01}}$ de acordo com $\frac{N_0}{N_1}$, busca uma solução que desconsidera a influência das probabilidades a priori das classes, confiando somente na informação associada às características observadas, i.e., nas verossimilhanças $p(\mathbf{x}|y = k)$. Nesse caso, se as matrizes de covariância para as densidades condicionais $p(\mathbf{x}|y = k)$ forem iguais, a solução ótima corresponde ao ponto de operação (na Curva ROC) que equilibra as probabilidades de erro/acerto para cada classe, i.e., $P(\mathbf{x} \in \mathbf{R}_0|y = 1) = P(\mathbf{x} \in \mathbf{R}_1|y = 0)$, conforme ilustrado pela Figura 8. Esse ponto de operação é conhecido na literatura como *break-even point* (Duda et al., 2000), e se caracteriza por produzir taxas iguais de verdadeiros positivos (sensibilidade) e de verdadeiros negativos (especificidade).

Finalmente, é importante ressaltar que embora a investigação realizada nesse trabalho tenha se concentrado em tarefas de classificação com somente duas classes (binárias), problemas desbalanceados contendo $c > 2$ classes são comuns na

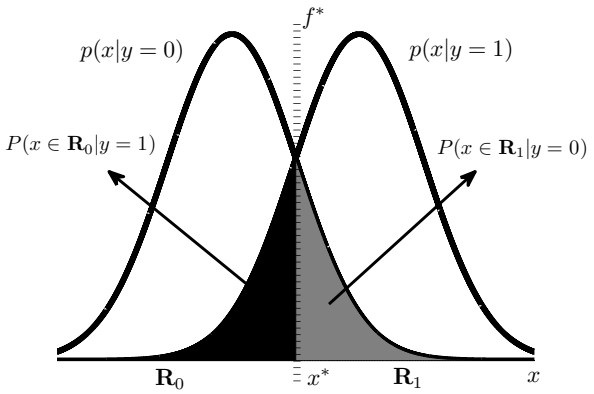

Figura 8: Solução ótima f^* equilibrando as probabilidades de erro (áreas em cinza e preto) quando as variâncias das distribuições são iguais.

prática. Assim, com o objetivo de complementar essa investigação, os parágrafos a seguir fornecem uma breve discussão a respeito de como os principais conceitos aqui apresentados podem ser estendidos para o contexto de aprendizado multiclasse.

Enquanto que para o caso binário, o objetivo da formulação padrão do problema do aprendizado é a minimização da probabilidade do erro de classificação (conforme apresentado na Seção 2.2), para o caso geral de c classes, esse objetivo torna-se a maximização da probabilidade de um exemplo ser corretamente classificado[6], dada pela seguinte expressão (Berger, 1985; Duda et al., 2000),

$$\begin{aligned} P\,(\text{Correto}) &= \sum_{k=1}^{c} P(\mathbf{x} \in \mathbf{R}_k, y = k) \\ &= \sum_{k=1}^{c} P(\mathbf{x} \in \mathbf{R}_k|y = k)P(y = k) \\ &= \sum_{k=1}^{c} \int_{\mathbf{R}_k} p(\mathbf{x}|y = k)P(y = k)d\mathbf{x}. \end{aligned} \quad (45)$$

Note que para maximizar a probabilidade de estar correto deve-se atribuir cada vetor $\mathbf{x}$ à classe k que fornece o maior integrando em (45). Isso leva à seguinte expressão para a regra de decisão ótima (multiclasse) (Duda et al., 2000),

$$f^*(\mathbf{x}) = \arg_k \max\, p(\mathbf{x}|y = k)P(y = k), \quad (46)$$

com $k = 1, \ldots, c$.

[6] A definição do funcional *risco* em termos das probabilidades de acerto das classes é mais simples, uma vez que para o caso multiclasse existem mais formas de se errar do que de se acertar (Duda et al., 2000).

A expressão (46) define a solução alvo das máquinas de aprendizado para o caso multiclasse. Uma análise das propriedades dessa solução em situações controladas (com distribuições de probabilidade conhecidas) poderia ser conduzida para se prover um melhor entendimento das causas/efeitos do problema geral de classes desbalanceadas. Metodologia equivalente foi adotada na Seção 2.3 para o caso binário. Nessa análise seria possível demonstrar, por exemplo, que a regra de decisão ótima (multiclasse) favorece as classes com maior probabilidade de ocorrência, quando o cenário é desbalanceado. Observe a partir de (46) que se uma ambiguidade surge na classificação de um exemplo particular $\mathbf{x}$, devido aos valores similares observados para as densidades condicionais de todas as c classes, i.e., $p(\mathbf{x}|y = k) \approx p(\mathbf{x}|y = j)$ para todo $j \neq k$, a regra f^* deverá atribuir $\mathbf{x}$ à classe majoritária.

Em relação às métricas de avaliação descritas na Seção 3, algumas delas podem ser estendidas para o caso geral de c classes. Nesse cenário, o desempenho de um classificador pode ser descrito por uma matriz de confusão c x c normalizada[7], com os c elementos da diagonal principal representando as taxas de classificação corretas e, os $c^2 - c$ elementos fora dessa diagonal, representando as taxas de erro. No âmbito da análise ROC multiclasse, essa matriz define um único ponto em um espaço de dimensão $c^2 - c$, cujos eixos representam as taxas de erro entre as classes. Os elementos da diagonal principal não precisam ser representados, uma vez que os mesmos são equivalentes ao complemento da soma das taxas de erro de cada linha, i.e., $e_{k,k} = 1 - \sum_{j=1}^{c} e_{k,j},\ k \neq j$ (Fawcett, 2006). A geração da superfície ROC multidimensional envolve a ponderação das saídas do classificador por todas as possíveis combinações de custos entre classes (*thresholds*). Esse procedimento possui elevado custo computacional (exponencial), podendo limitar o uso da análise ROC quando o número de classes c é muito grande (Landgrebe and Duin, 2008). Apesar dessa limitação, alguns trabalhos têm investigado os aspectos associados ao projeto de gráficos ROC para domínios multiclasse (Lane, 2000; Fawcett, 2006; Everson and Fieldsend, 2006b; Landgrebe and Duin, 2008). De acordo com Fawcett (2006), uma estratégia simples que poderia ser aplicada é gerar c curvas ROC, sendo uma para cada classe. Nesse caso, a k-ésima curva mostra o desempenho de classificação, considerando a classe k como a classe positiva e todas as demais classes como a classe negativa (abordagem *one-against-all*). No trabalho de Landgrebe and Duin (2008), um algoritmo eficiente foi proposto para a geração de aproximações acuradas para superfícies ROC multidimensionais.

[7]Na matriz de confusão normalizada, os elementos de cada linha (veja Tabela 1 na Seção 3) são divididos pelo número total de exemplos da classe que a linha representa.

Similarmente, extensões multiclasse das métricas AUC e *G-mean* têm sido discutidas (Hand and Till, 2001; Everson and Fieldsend, 2006b; Landgrebe and Duin, 2006; Sun et al., 2007). A formulação proposta em Hand and Till (2001) para o cálculo do VUS (*Volume Under the ROC Surface*) é baseada na agregação dos valores de AUC estimados para todos os pares de classes. Tal formulação é eficiente para valores elevados de c. No trabalho de Everson and Fieldsend (2006b), uma generalização do coeficiente *Gini*, análogo à métrica AUC, foi proposta para quantificar o desempenho multiclasse de um modelo em relação ao classificador aleatório, i.e., que faz predições aleatórias para os exemplos. Além disso, uma extensão simples da métrica *G-mean* pode ser encontrada em Sun et al. (2007). Todos esses trabalhos reforçam a importância de se possuir um métrica global de avaliação que, ao contrário da acurácia (ou taxa de erro), não produza falso sentimento em cenários desbalanceados.

Por último, ao se considerar as abordagens propostas para solucionar o problema de classes desbalanceadas na Seção 4, entende-se que os métodos de *reamostragem* da categoria de *pré-processamento de dados* (Seção 4.1) poderiam ser mais facilmente adaptados para domínios com $c > 2$ classes, uma vez que os mesmos atuam somente no espaço de entrada (dados) sendo, portanto, independentes da formulação do algoritmo de aprendizado. Adicionalmente, alguns métodos, tais como *sobreamostragem* com substituição, SMOTE (Chawla et al., 2002), *subamostragem* aleatória e OSS (Kubat and Matwin, 1997) poderiam ser aplicados individualmente sobre cada classe, para aumentar/diminuir a representatividade (custos) dos grupos minoritários/majoritários no conjunto de treinamento. Essa estratégia foi adotada, por exemplo, no trabalho de Zhou and Liu (2006).

Na categoria de *adaptações em algoritmos de aprendizado* (Seção 4.2), a extensão das soluções para problemas com $c > 2$ classes depende da formulação original dos algoritmos de aprendizado sobre os quais essas soluções foram construídas e por isso, não é uma tarefa trivial. Algoritmos baseados em *kernel* como as SVMs, por exemplo, foram originalmente concebidos com base nos fundamentos da Teoria do Aprendizado Estatístico (Vapnik, 1995). Nessa teoria, limites superiores para o erro de generalização das máquinas de aprendizado são definidos com base na medida de complexidade VC das funções estimadas (regras de decisão) e, no tamanho do conjunto de dados. No caso da tarefa de classificação de padrões, esses limites são derivados no âmbito de problemas com duas classes. Isso explica a natureza binária das formulações propostas para SVMs e outras máquinas de *kernel* e também, a dificuldade associada à extensão natural desses algoritmos para problemas com $c > 2$ classes. No caso de SVMs, por exemplo, as abordagens mais adotadas para aprendizado multiclassse são baseadas na combinação de múltiplas SVMs binárias (Bishop, 2006). Nessa linha,

uma alternativa para a extensão das soluções fundamentalmente binárias apresentadas na Seção 4.2, é a decomposição de um problema de classificação com $c > 2$ classes dentro de múltiplos problemas com duas classes. As abordagens mais comuns para efetuar essa decomposição são *one-against-all* e *one-against-one*. Tais abordagens são independentes do algoritmo de aprendizado, sendo mais populares entre SVMs, *Boosting* e árvores de decisão. Para detalhes sobre o funcionamento desses métodos, recomenda-se os trabalhos de Vapnik (1998) e Bishop (2006).

Diferentemente das SVMs, a formulação original apresentada por modelos neurais *feed-forward*, tais como redes MLP e redes RBF, permite uma extensão natural para problemas com $c > 2$ classes. Um esquema comumente adotado para realizar essa extensão, é a codificação 1 de c, onde c é o número de unidades de saída do modelo (Bishop, 2006). Os trabalhos de Kukar and Kononenko (1998) e, Alejo et al. (2006) apresentados no contexto de *redes reurais sensíveis ao custo* (Seção 4.2.5) usam essa codificação e portanto, permitem diretamente o aprendizado multiclasse.

AGRADECIMENTOS

O presente trabalho foi realizado com o apoio da Fundação de Amparo à Pesquisa do Estado de Minas Gerais - FAPEMIG.

REFERÊNCIAS

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

Alejo, R., García, V., Sotoca, J., Mollineda, R. A. and Sánchez, J. (2006). Improving the performance of the rbf neural networks trained with imbalanced samples, *Proc. of Intell. Data Eng. Autom. Learn.*, Vol. 7 of *Lecture Notes in Computer Science*, Springer, pp. 720–747.

Anand, R., Mehrotra, K., Mohan, C. and Ranka, S. (1993). An improved algorithm for neural network classification of imbalanced training sets, *IEEE Transactions on Neural Networks, 6(4):962-969* .

Asuncion, A. and Newman, D. (2007). UCI machine learning repository.

Bather, J. (2000). *Decision Theory: An Introduction to Dynamic Programming and Sequential Decisions*, Wiley.

Batista, G. E. A. P. A., Prati, R. C. and Monard, M. C. (2004). A study of the behavior of several methods for balancing machine learning training data, *SIGKDD Explor. Newsl.* **6**(1): 20–29.

Batista, G. E., Prati, R. C. and Monard, M. C. (2005). Balancing strategies and class overlapping, *Advances in Intelligent Data Analysis VI*, Vol. 3646 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 24–35.

Beckmann, M. (2010). *Algoritmos genéticos como estratégia de pré-processamento em conjuntos de dados desbalanceados*, Master's thesis, Programa de Pós-Graduação em Engenharia Civil - COPPE - UFRJ.

Beckmann, M. and Lima, B. (2009). Algoritmos genéticos como estratégia de pré-processamento para o aprendizado de máquina em conjuntos de dados desbalanceados, *Anais do XXX Congresso Ibero Americano de Métodos Computacionais em Engenharia*, pp. 210–223.

Bergamini, C., Oliveira, L. S., Koerich, A. L. and Sabourin, R. (2009). Combining different biometric traits with one-class classification, *Signal Process.* **89**: 2117–2127.

Berger, J. (1985). *Statistical Decision Theory and Bayesian Analysis*, second edn, Springer.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer.

Boser, B. E., Guyon, I. M. and Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers, *COLT '92: Proceedings of the fifth annual workshop on Computational learning theory*, ACM, New York, NY, USA, pp. 144–152.

Bradley, A. P. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms, *Pattern Recognition* **30**(7): 1145–1159.

Braga, A. P., Horta, E. G., Natowicz, R., Rouzier, R., Incitti, R., Rodrigues, T. S., Costa, M. A., Pataro, C. D. M. and Çela, A. (2008). Bayesian classifiers for predicting the outcome of breast cancer preoperative chemotherapy., *ANNPR*, Vol. 5064 of *Lecture Notes in Computer Science*, Springer, pp. 263–266.

Breiman, L., Friedman, J., Stone, C. J. and Olshen, R. A. (1984). *Classification and Regression Trees*, Chapman & Hall/CRC.

Carvalho, A., Pozo, A., Vergilio, S. and Lenz, A. (2008). Predicting fault proneness of classes trough a multiobjective particle swarm optimization algorithm, *Proceedings of the 2008 20th IEEE International Conference on Tools with Artificial Intelligence - Volume 02*, IEEE Computer Society, pp. 387–394.

Castro, C. and Braga, A. (2008). Optimization of the area under the roc curve, *Proceedings of the 10th Brazilian Symposium on Neural Networks (SBRN '08)*, IEEE Computer Society, Washington, DC, USA, pp. 141–146.

Castro, C. and Braga, A. (2009). Artificial neural networks learning in roc space, *Proc. of the 1st International Conference on Neural Computation (ICNC'09)*, INSTICC, pp. 219–224.

Castro, C. L., Carvalho, M. A. and Braga, A. P. (2009). An improved algorithm for svms classification of imbalanced data sets, *Engineering Applications of Neural Networks*, Vol. 43 of *Communications in Computer and Information Science*, Springer Berlin Heidelberg, pp. 108–118.

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

Chawla, N. V., Japkowicz, N. and Kotcz, A. (2003). *Proceedings of the ICML'2003 Workshop on Learning from Imbalanced Data Sets, ICML*.

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

Chen, S., Wang, X., Hong, X. and Harris, C. (2006). Kernel classifier construction using orthogonal forward selection and boosting with fisher ratio class separability measure, *IEEE Transactions on Neural Networks* **17**(6): 1652–1656.

Cherkassky, V. and Mulier, F. (2007). *Learning from data*, 2 edn, John Wiley and Sons.

Cortes, C. and Mohri, M. (2004). Auc optimization vs. error rate minimization, *Advances in Neural Information Processing Systems 16*, MIT Press, Cambridge, MA.

Cortes, C. and Vapnik, V. (1995). Support-vector networks, *Mach. Learn.* **20**(3): 273–297.

Cristianini, N., Kandola, J., Elisseeff, A. and Shawe-Taylor, J. (2002). On kernel-target alignment, *Advances in Neural Information Processing Systems 14*, Vol. 14, pp. 367–373.

Drummond, C. and Holte, R. (2003). C4.5, class imbalance, and cost sensitivity: Why under-sampling beats over-sampling, *Working Notes of the ICML Workshop Learning from Imbalanced Data Sets*.

Duda, R. O., Hart, P. E. and Stork, D. G. (2000). *Pattern Classification (2nd Edition)*, Wiley-Interscience.

Egan, J. P. (1975). *Signal Detection Theory and ROC Analysis*, Academic Press.

Elkan, C. (2001). The foundations of cost-sensitive learning, *Proceedings of the Seventeenth International Joint Conference on Artificial Intelligence, IJCAI*, pp. 973–978.

Estabrooks, A., Jo, T. and Japkowicz, N. (2004). A multiple resampling method for learning from imbalanced data sets, *Computational Intelligence* **20**(1): 18–36.

Everson, R. M. and Fieldsend, J. E. (2006a). Multi-class roc analysis from a multi-objective optimisation perspective, *Pattern Recogn. Lett.* **27**(8): 918–927.

Everson, R. M. and Fieldsend, J. E. (2006b). Multi-objective optimisation for receiver operating characteristic analysis, *Multi-Objective Machine Learning*, pp. 533–556.

Fan, W., Stolfo, S. J., Zhang, J. and Chan, P. K. (1999). Adacost: misclassification cost-sensitive boosting, *Proceedings of IEEE International Conference on Machine Learning*, Morgan Kaufmann, pp. 97–105.

Fawcett, T. (2004). Roc graphs: Notes and practical considerations for researchers, *Technical report*, HP Laboratories, Palo Alto, USA.

Fawcett, T. (2006). An introduction to roc analysis, *Pattern Recogn. Lett.* **27**(8): 861–874.

Fawcett, T. and Provost, F. (1997). Adaptive fraud detection, *Data Min. Knowl. Discov.* **1**(3): 291–316.

Gallinari, P., Thiria, S., Badran, F. and Fogelman-Soulie, F. (1991). On the relations between discriminant analysis and multilayer perceptrons, *Neural Netw.* **4**(3): 349–360.

Gao, Y., Wang, S. and Liu, Z. (2009). Automatic fault detection and diagnosis for sensor based on kpca, *Proceedings of International Symposium on the Computational Intelligence and Design*, IEEE Computer Society, pp. 135–138.

Girosi, F., Jones, M. and Poggio, T. (1995). Regularization theory and neural networks architectures, *Neural Comput.* **7**(2): 219–269.

Graening, L., Jin, Y. and Sendhoff, B. (2006). Generalization improvement in multi-objective learning, *International Joint Conference on Neural Networks*, IEEE Press, pp. 9893–9900.

Han, H., Wang, W.-Y. and Mao, B.-H. (2005). Borderline-smote: A new over-sampling method in imbalanced data sets learning, *Advances in Intelligent Computing*, Vol. 3644 of *Lecture Notes in Computer Science*, Springer Berlin, Heidelberg, pp. 878–887.

Hand, D. and Till, R. (2001). A simple generalisation of the area under the roc curve for multiple class classification problems, *Mach. Learn.* **45**: 171–186.

Hanley, J. A. and Mcneil, B. J. (1982). The meaning and use of the area under a receiver operating characteristic (roc) curve., *Radiology* **143**(1): 29–36.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*, Macmillan, New York.

He, H., Bai, Y., Garcia, E. A. and Li, S. (2008). Adasyn: Adaptive synthetic sampling approach for imbalanced learning, *Proceedings of the International Joint Conference on Neural Networks, IJCNN 2008*, pp. 1322–1328.

He, H. and Garcia, E. A. (2009). Learning from imbalanced data, *IEEE Transactions on Knowledge and Data Engineering* **21**(9): 1263–1284.

He, H. and Shen, X. (2007). A ranked subspace learning method for gene expression data classification, *Proceedings of the 2007 International Conference on Artificial Intelligence, ICAI 2007, Volume I, June 25-28, 2007, Las Vegas, Nevada, USA*, pp. 358–364.

Herschtal, A., Raskutti, B. and Campbell, P. K. (2006). Area under roc optimisation using a ramp approximation, *Proceedings of the Sixth SIAM International Conference on Data Mining*, pp. 1–11.

Hodge, V. and Austin, J. (2004). A survey of outlier detection methodologies, *Artif. Intell. Rev.* **22**(2): 85–126.

Hong, X., Chen, S. and Harris, C. (2007). A kernel-based two-class classifier for imbalanced data sets, *IEEE Transactions on Neural Networks* **18**(1): 28–41.

Ishida, C. and Pozo, A. (2007). Optimization of the auc criterion for rule subset selection, *Proceedings of the Seventh International Conference on Intelligent Systems Design and Applications*, IEEE Computer Society, pp. 497–502.

Japkowicz, N. (2000a). *Proceedings of the AAAI'2000 Workshop on Learning from Imbalanced Data Sets, AAAI Tech Report WS-00-05*.

Japkowicz, N. (2000b). Learning from imbalanced data sets: A comparison of various strategies, *AAAI Conference on Artificial Intelligence*, AAAI Press, pp. 10–15.

Japkowicz, N. (2001). Supervised versus unsupervised binary-learning by feedforward neural networks, *Mach. Learn.* **42**(1-2): 97–122.

Japkowicz, N. (2003). Class imbalances: Are we focusing on the right issue?, *Proceedings of the International Conf. Machine Learning, Workshop on Learning from Imbalanced Data Sets II*.

Japkowicz, N. and Stephen, S. (2002). The class imbalance problem: A systematic study, *Intell. Data Anal.* **6**(5): 429–449.

Jo, T. and Japkowicz, N. (2004). Class imbalances versus small disjuncts, *SIGKDD Explor. Newsl.* **6**(1): 40–49.

Joachims, T. (2002). *Learning to Classify Text Using Support Vector Machines: Methods, Theory and Algorithms*, Kluwer Academic Publishers, Norwell, MA, USA.

Joachims, T. (2005). A support vector method for multivariate performance measures, *ICML'05: Proceedings of the 22nd international conference on Machine learning*, ACM, New York, NY, USA, pp. 377–384.

Kandola, J. and Shawe-Taylor, J. (2003). Refining kernels for regression and uneven classification problems, *Proceedings of International Conference on Artificial Intelligence and Statistics*, Springer-Verlag, Berlin Heidelberg.

Karakoulas, G. and Shawe-Taylor, J. (1999). Optimizing classifiers for imbalanced training sets, *Proceedings of Conference on Advances in Neural Information Processing Systems II*, MIT Press, Cambridge, MA, USA, pp. 253–259.

Khoshgoftaar, T. M., Hulse, J. V. and Napolitano, A. (2010). Supervised neural network modeling: An empirical investigation into learning from imbalanced data with labeling errors, *IEEE Trans. on Neural Networks* **21**(5): 813–830.

Kohonen, T., Schroeder, M. R. and Huang, T. S. (eds) (2001). *Self-Organizing Maps*, 3rd edn, Springer-Verlag New York, Inc., Secaucus, NJ, USA.

Kubat, M., Holte, R. C. and Matwin, S. (1998). Machine learning for the detection of oil spills in satellite radar images., *Machine Learning* **30**(2-3): 195–215.

Kubat, M. and Matwin, S. (1997). Addressing the curse of imbalanced training sets: one-sided selection, *Proc. 14th International Conference on Machine Learning*, Morgan Kaufmann, pp. 179–186.

Kukar, M. and Kononenko, I. (1998). Cost-sensitive learning with neural networks, *Proceedings of the 13th European Conference on Artificial Intelligence (ECAI-98)*, John Wiley and Sons, pp. 445–449.

Kupinski, M. A. and Anastasio, M. A. (1999). Multiobjective genetic optimization of diagnostic classifiers with implications for generating receiver operating characterisitic curves, *IEEE Transactions on Medical Imaging* **18**: 675–685.

Landgrebe, T. and Duin, R. (2006). A simplified extension of the area under the roc to the multiclass domain, *Proceedings of the Seventeenth Annual Symposium of the Pattern Recognition Association of South Africa, PRASA 2006*, pp. 241–245.

Landgrebe, T. and Duin, R. (2008). Efficient multiclass roc approximation by decomposition via confusion matrix perturbation analysis, *IEEE Trans. Pattern Anal. Mach. Intell.* **30**: 810–822.

Lane, T. (2000). Extensions of roc analysis to multi-class domains, *Dietterich, T., Margineantu, D., Provost, F., Turney, P. (Eds.), ICML-2000, Workshop on Cost-Sensitive Learning*.

Lasko, T. A., Bhagwat, J. G., Zou, K. H. and Ohno-Machado, L. (2005). The use of receiver operating characteristic curves in biomedical informatics, *Journal of Biomedical Informatics* **38**(5): 404–415.

Laurikkala, J. (2001). Improving identification of difficult small classes by balancing class distribution, *AIME '01: Proceedings of the 8th Conference on AI in Medicine in Europe*, Springer-Verlag, London, UK, pp. 63–66.

Lawrence, S., Burns, I., Back, A. D., Tsoi, A. C. and Giles, C. L. (1998). Neural network classification and prior class probabilities, *Neural Networks: Tricks of the Trade, this book is an outgrowth of a 1996 NIPS workshop*, Springer-Verlag, London, UK, pp. 299–313.

Lee, H. and Cho, S. (2006). The novelty detection approach for different degrees of class imbalance, *Neural Information Processing*, Vol. 4233 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 21–30.

Li, Y. and Shawe-Taylor, J. (2003). The svm with uneven margins and chinese document categorization, *Proceedings of the 17th Pacific Asia Conference on Language, Information and Computation*, pp. 216–227.

Lin, Y., Lee, Y. and Wahba, G. (2002). Support vector machines for classification in nonstandard situations, *Mach. Learn.* **46**(1-3): 191–202.

Liu, X.-Y., Wu, J. and Zhou, Z.-H. (2009). Exploratory undersampling for class-imbalance learning, *IEEE Trans. on Sys. Man Cyber. Part B* **39**(2): 539–550.

Machado, E. (2007). *Um estudo de limpeza em base de dados desbalanceada e com sobreposição de classes*, Master's thesis, Curso de Mestrado em Informática - Universidade de Brasilia.

Machado, E. and Ladeira, M. (2007a). Dealing with rare cases and avoiding overfitting: Combining cluster-based oversampling and smote, *Proceeding of IX Argentine Symposium on Artificial Intelligence - ASAI*, pp. 47–55.

Machado, E. and Ladeira, M. (2007b). Um estudo de limpeza em base de dados desbalanceada com sobreposição de classes, *VI Encontro Nacional de Inteligência Artificial - Anais do XXVII Congresso da Sociedade Brasileira de Computação*, pp. 330–340.

Maloof, M. A. (2003). Learning when data sets are imbalanced and when costs are unequal and unknown, *Proceedings of the International Conf. Machine Learning, Workshop on Learning from Imbalanced Data Sets II*.

Manevitz, L. M. and Yousef, M. (2002). One-class svms for document classification, *J. Mach. Learn. Res.* **2**: 139–154.

Manevitz, L. and Yousef, M. (2007). One-class document classification via neural networks, *Neurocomput.* **70**(7-9): 1466–1481.

Markou, M. and Singh, S. (2003). Novelty detection: A review - part 2: Neural network based approaches, *Signal Processing* **83**: 2499–2521.

Mease, D., Wyner, A. J. and Buja, A. (2007). Boosted classification trees and class probability/quantile estimation, *J. Mach. Learn. Res.* **8**: 409–439.

Milaré, C., Batista, G. and Carvalho, A. (2010). A hybrid approach to learn with imbalanced classes using evolutionary algorithms, *Logic Journal of IGPL* .

Monard, M. and Batista, G. (2002). Learning with skewed class distribution, *Advances in Logic, Artificial Intelligence and Robotics*, IOS Press, pp. 173–180.

Morik, K., Brockhausen, P. and Joachims, T. (1999). Combining statistical learning with a knowledge-based approach - a case study in intensive care monitoring, *Proceedings of the Sixteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 268–277.

Moturu, S. T., Johnson, W. G. and Liu, H. (2010). Predictive risk modelling for forecasting high-cost patients: a real-world application using medicaid data, *International Journal of Biomedical Engineering and Technology* **2**(1): 114–132.

Muller, K. R., Mika, S., Ratsch, G., Tsuda, K. and Scholkopf, B. (2001). An introduction to kernel-based learning algorithms, *IEEE Trans. on Neural Networks* **12**(2): 181–201.

Natowicz, R., Incitti, R., Horta, E. G., Charles, B., Guinot, P., Yan, K., Coutant, C., Andre, F., Pusztai, L. and Rouzier, R. (2008). Prediction of the outcome of preoperative chemotherapy in breast cancer by dna probes that convey information on both complete and non complete responses, *BMC Bioinformatics* **9**: 149–166.

Pazzani, M., Merz, C., Murphy, P., Ali, K., Hume, T. and Brunk, C. (1994). Reducing misclassification costs, *Proceedings of the 11th International Conference on Machine Learning, ICML*, Morgan Kaufmann, pp. 217–225.

Pearson, P., Goney, G. and Shwaber, J. (2003). Imbalanced clustering for microarray time-series, *Proc. 20th International Conference on Machine Learning (ICML'03)*.

Prati, R., Batista, G. and Monard, M. (2004a). Learning with class skews and small disjuncts, *Advances in Artificial Intelligence, SBIA 2004*, Vol. 3171 of *Lecture Notes in Computer Science*, Springer Berlin, Heidelberg, pp. 1119–1139.

Prati, R., Batista, G. and Monard, M. (2008a). Evaluating classifiers using roc curves, *Latin America Transactions, IEEE (Revista IEEE America Latina)* **6**(2): 215 – 222.

Prati, R., Batista, G. and Monard, M. (2008b). A study with class imbalance and random sampling for a decision tree learning system, *Artificial Intelligence in Theory and Practice II*, Vol. 276 of *IFIP International Federation for Information Processing*, Springer Boston, pp. 131–140.

Prati, R. C., Batista, G. E. A. P. A. and Monard, M. C. (2004b). Class imbalances versus class overlapping: An analysis of a learning system behavior, *MICAI 2004: Advances in Artificial Intelligence, Third Mexican International Conference on Artificial Intelligence*, Vol. 2972 of *Lecture Notes in Computer Science*, Springer, pp. 312–321.

Provost, F. and Fawcett, T. (1997). Analysis and visualization of classifier performance: Comparison under imprecise class and cost distributions, *Proceedings of the Third International Conference on Knowledge Discovery and Data Mining*, AAAI Press, pp. 43–48.

Provost, F. and Fawcett, T. (1998). Robust classification systems for imprecise environments, *AAAI '98/IAAI '98: Proceedings of the fifteenth national/tenth conference on Artificial intelligence/Innovative applications of artificial intelligence*, American Association for Artificial Intelligence, Menlo Park, CA, USA, pp. 706–713.

Provost, F. and Fawcett, T. (2001). Robust classification for imprecise environments, *Mach. Learn.* **42**(3): 203–231.

Provost, F. J., Fawcett, T. and Kohavi, R. (1998). The case against accuracy estimation for comparing induction algorithms, *ICML'98: Proceedings of the Fifteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 445–453.

Raskutti, B. and Kowalczyk, A. (2004). Extreme rebalancing for svms: a case study, *SIGKDD Explor. Newsl.* **6**(1): 60–69.

Rodrigues, J., Barros, F. and Prudencio, R. (2009). B-boost: Uma extensão do método de boosting para conjuntos de treinamento desbalanceados, *VII Encontro Nacional de Inteligência Artificial - Anais do XXIX Congresso da Sociedade Brasileira de Computação*, pp. 1039–1048.

Rumelhart, D. E. and McClelland, J. L. (1986). *Parallel distributed processing: Explorations in the microstructure of cognition*, Vol. 1: Foundations, MIT Press.

Sanchez, M. S., Ortiz, M. C., Sarabia, L. A. and Lleti, R. (2005). On pareto-optimal fronts for deciding about sensitivity and specificity in class-modelling problems, *Analytica Chimica Acta* **544**(1-2): 236 – 245.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J. C., Smola, A. J. and Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution, *Neural Comput.* **13**(7): 1443–1471.

Silva, C., Silva, A., Netto, S., Paiva, A., Junior, G. and Nunes, R. (2009). Lung nodules classification in ct images using simpsons index, geometrical measures and one-class svm, *Machine Learning and Data Mining in Pattern Recognition*, Vol. 5632 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 810–822.

Souza, M. R. P., Cavalcanti, G. D. C. and Tsang, I. R. (2010). Off-line signature verification: An approach based on combining distances and one-class classifiers, *Proceedings of the 22nd IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2010, Arras, France*, IEEE Computer Society, pp. 7–11.

Spackman, K. A. (1989). Signal detection theory: valuable tools for evaluating inductive learning, *Proceedings of the sixth international workshop on Machine learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 160–163.

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

Swets, J. A., Dawes, R. M. and Monahan, J. (2000). Better decisions through science., *Scientific American* **283**(4): 82–87.

Tamee, K., Rojanavasu, P., Udomthanapong, S. and Pinngern, O. (2008). Using self-organizing maps with learning classifier system for intrusion detection, *PRICAI 2008: Trends in Artificial Intelligence*, Vol. 5351 of *Lecture Notes in Computer Science*, Springer Berlin / Heidelberg, pp. 1071–1076.

Tan, P.-N., Steinbach, M. and Kumar, V. (2005). *Introduction to Data Mining, (First Edition)*, Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.

Tang, Y. and Zhang, Y.-Q. (2006). Granular svm with repetitive undersampling for highly imbalanced protein homology prediction, *Proceedings of International Conference on Granular Computing*, pp. 457–460.

Tang, Y., Zhang, Y.-Q., Chawla, N. V. and Krasser, S. (2009). Svms modeling for highly imbalanced classification, *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics* **39**(1): 281–288.

Teixeira, R., Braga, A., Takahashi, R. and Saldanha, R. (2000). Improving generalization of mlps with multi-objective optimization, *Neurocomputing* **35**(1-4): 189–194.

Ting, K. M. (2000). A comparative study of cost-sensitive boosting algorithms, *Proceedings of the 17th International Conference on Machine Learning*, Morgan Kaufmann, pp. 983–990.

Tomek, I. (1976). Two modifications of cnn, *IEEE Trans. Systems, Man, and Cybernetics* **6**(11): 769–772.

Vapnik, V. (1998). *Statistical Learning Theory*, Wiley-Interscience.

Vapnik, V. N. (1995). *The nature of statistical learning theory*, Springer-Verlag New York, Inc.

Veropoulos, K., Campbell, C. and Cristianini, N. (1999). Controlling the sensitivity of support vector machines, *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 55–60.

Wang, B. X. and Japkowicz, N. (2008). Boosting support vector machines for imbalanced data sets, *Proceedings of the 17th ISMIS 2008 - International Symposium on Foundations of Intelligent Systems*, Vol. 4994 of *Lecture Notes in Computer Science*, Springer, pp. 38–47.

Weiss, G. M. (2004). Mining with rarity: a unifying framework, *SIGKDD Explor. Newsl.* **6**(1): 7–19.

Weiss, G. M. (2005). Mining with rare cases, *The Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practioners and Researchers*, Springer, pp. 765–776.

Weiss, G. M. and Provost, F. (2003). Learning when training data are costly: The effect of class distribution on tree induction, *Journal of Artificial Intelligence Research* **19**: 315–354.

Wilson, D. (1972). Asymptotic properties of nearest neighbor rules using edited data, *IEEE Trans. Systems, Man, and Cybernetics* **2**(3): 408–421.

Wu, G. and Chang, E. Y. (2003). Adaptive feature-space conformal transformation for imbalanced-data learning, *Proceedings of IEEE International Conference on Machine Learning*, pp. 816–823.

Wu, G. and Chang, E. Y. (2004). Aligning boundary in kernel space for learning imbalanced dataset, *ICDM '04: Proceedings of the Fourth IEEE International Conference on Data Mining*, IEEE Computer Society, Washington, DC, USA, pp. 265–272.

Wu, G. and Chang, E. Y. (2005). Kba: Kernel boundary alignment considering imbalanced data distribution, *IEEE Transactions on Knowledge and Data Engineering* **17**(6): 786–795.

Zadrozny, B. and Elkan, C. (2001). Obtaining calibrated probability estimates from decision trees and naive bayesian classifiers, *ICML'01: Proceedings of the Eighteenth International Conference on Machine Learning*, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, pp. 609–616.

Zhang, J. and Mani, I. (2003). Knn approach to unbalanced data distributions: A case study involving information extraction, *Proceedings of the ICML'2003 workshop on learning from imbalanced datasets*.

Zhou, Z.-H. and Liu, X.-Y. (2006). Training cost-sensitive neural networks with methods addressing the class imbalance problem, *IEEE Transactions on Knowledge and Data Engineering* **18**(1): 63–77.