TESE DE DOUTORADO Nº 239

UMA NOVA ABORDAGEM PARA APRENDIZADO DE MÚLTIPLAS INSTÂNCIAS, BASEADA EM SELEÇÃO DE INSTÂNCIAS VIA ESTIMADOR DE DENSIDADE POR KERNEL

Alexandre Wagner Chagas Faria

DATA DA DEFESA: 12/08/2016

Universidade Federal de Minas Gerais

Escola de Engenharia

Programa de Pós-Graduação em Engenharia Elétrica

UMA NOVA ABORDAGEM PARA APRENDIZADO DE MÚLTIPLAS INSTÂNCIAS, BASEADA EM SELEÇÃO DE INSTÂNCIAS VIA ESTIMADOR DE DENSIDADE POR KERNEL

Alexandre Wagner Chagas Faria

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

Orientador: Prof. Antônio de Pádua Braga

Belo Horizonte - MG

Agosto de 2016

F224n

Faria, Alexandre Wagner Chagas.
Uma nova abordagem para aprendizado de múltiplas instâncias, baseada em seleção de instâncias via estimador de densidade por Kernel [manuscrito] / Alexandre Wagner Chagas Faria. - 2016.
x, 81 f., enc.: il.

Orientador: Antônio de Pádua Braga.
Coorientador: André Paim Lemos.

Tese (doutorado) Universidade Federal de Minas Gerais, Escola de Engenharia.

Bibliografia: f. 73-81.

1. Engenharia elétrica - Teses. 2. Aprendizado de múltiplas instâncias - Teses. 3. Kernel, Funções de - Teses. I. Braga, Antônio de Pádua. II. Lemos, André Paim. III. Universidade Federal de Minas Gerais. Escola de Engenharia. IV.Título.

CDU: 621.3(043)

"Uma Nova Abordagem para Aprendizado de Múltiplas Instâncias, Baseada em Seleção de Instâncias via Estimador de Densidade por Kernel"

Alexandre Wagner Chagas Faria

Tese de Doutorado submetida à Banca Examinadora designada pelo Colegiado do Programa de Pós-Graduação em Engenharia Elétrica da Escola de Engenharia da Universidade Federal de Minas Gerais, como requisito para obtenção do grau de Doutor em Engenharia Elétrica.

Aprovada em 12 de agosto de 2016.

Por:

Prof. Dr. Antônio de Pádua Braga
DELT (UFMG) - Orientador

Prof. Dr. André Paim Lemos
DELT (UFMG) - Coorientador

Prof. Dr. Frank Sill Torres
DELT (UFMG)

Prof. Dr. Cristiano Leite de Castro
DEE (UFMG)

Prof. Dr. David Menotti Gomes
Departamento de Informática (UFPR)

Prof. Luis Enrique Zárate
Departamento de Informática (PUC-MG)

Resumo

Aprendizado de Múltiplas Instâncias (AMI) é uma generalização do aprendizado supervisionado. AMI tem sido utilizado em inúmeras aplicações, em que a rotulação para cada instância na fase de aprendizado é, às vezes, impossível ou não eficiente de ser realizada de forma prática.

A fim de lidar com essa família de problemas, o AMI propõe um novo paradigma, assinalando um único rótulo (positivo ou negativo) para um conjunto de instâncias, chamado de *bags*.

Mais formalmente, um *bag* é rotulado como positivo se contém, ao menos, uma instância positiva e rotulado como negativo se é conhecido, com certeza, que todas as suas instâncias são negativas.

Embora exista, na literatura, um número expressivo de propostas para trabalhar com AMI, poucos métodos fornecem resultados competitivos para a maioria das bases de dados avaliadas. Além disso, falta uma análise mais aprofundada entre os métodos existentes.

Neste trabalho são propostos dois novos algoritmos, baseados em seleção de instâncias por meio do cálculo da probabilidade, utilizando-se Estimador de Densidade por *Kernel*. O método utiliza o algoritmo *LogitBoost* como classificador. As abordagens baseadas em seleção de instâncias visam identificar as instâncias mais representativas em cada *bag* positivo, eliminando possíveis ruídos dentro desses *bags* para, assim, realizar um treinamento mais robusto.

Testes estatísticos demonstraram que as abordagens propostas são equivalentes aos melhores algoritmos da literatura, superando todos em algumas das bases avaliadas.

Também foi desenvolvida nesta tese uma nova aplicação, com base na metodologia proposta, com objetivo de selecionar pacientes que melhor representem cada classe em uma base de dados conhecida de Leucemia. Os experimentos mostram que com o emprego das abordagens propostas, foi possível reduzir, pela metade, os pacientes de treinamento e encontrar resultados ligeiramente melhores em relação àqueles que aqueles quando empregados todos os pacientes da base.

Abstract

Multiple Instance Learning (MIL) is a generalization of the supervised learning. MIL has been used in numerous applications where the instance labeling for individual instance, for the learning step, is sometimes not possible or unfeasible in practical way.

For dealing with this family of problem, MIL proposes a new paradigm by assigning a single label (positive or negative) to a set of instances, called *bag*. More formally, a *bag* is labeled positive if it contains at least one positive instance, and it is labeled negative if all instances are certainly negative.

Although there is a considerable number of algorithms to work with MIL in the literature, few works provides balanced outcomes for the majority of the datasets. Furthermore, a deeper analysis, among the existing methods, is not available.

In this work are proposed two new algorithms based on instance selection by likelihood computation, using Kernel Density Estimation. The approach uses the LogitBoost algorithm as classifier. The instance selection approach aim to identify the most representative instances in each positive *bag*, eliminating possible instance noise inside those *bags*, in this way, perform a more robust learning step.

Statistical tests, have demonstrated that the proposal methods are comparable with the best literature algorithms, overcoming all in some datasets.

It is also developed in this work a new application based on the proposed method in order to select patients that best represent each class in a Leukemia dataset. After experiments, it was possible to reduce the training patients by half, and find slightly better results than those when is used all patients in the dataset.

Dedico esse trabalho aos meus Avós Maria e Benedito, a minha Mãe Silezia, e aos meus filhos Letícia e Bernardo, as maiores alegrias e realizações da minha vida.

Agradecimentos

Primeiramente, ao meu inseparável DEUS, muito obrigado pela dádiva da minha vida, por ter segurado minha mão em todos os momentos da minha vida, por minhas maiores virtudes: perseverança e humildade, que graças a elas me trouxeram até aqui. Agradeço a DEUS por ter ouvido minhas preces e, por algumas vezes, ter me ouvido sem mesmo eu falar. Agradeço pelos amigos "irmãos" que foram colocados em minha vida e por iluminar meu caminho. Agradeço a ti meu DEUS, por ter colocado em minha vida as melhores pessoas do mundo para me criar: meus AVÓS, que me ensinaram meus bens mais preciosos: caráter e honestidade.

Agradeço aos meus Avós: Vovó Maria e Vovô Bené e minha Mãe Silézia, a vocês agradeço pela minha vida, agradeço por tudo que fizeram por mim, por tudo que me ensinaram, pelos valores que foram me passado, agradeço por ser quem sou. Novamente, este é mais um daqueles momentos que tanto gostaria de compartilhar com a presença de vocês ao meu lado, espero que se sintam orgulhosos por mais uma vitória nossa.

Aos meus filhos Letícia Caroline e Bernardo Alexandre, os seus sorrisos e carinhos foram um dos maiores incentivos e fonte de forças para eu chegar ao fim dessa jornada tão difícil.

À minha esposa Keila, obrigado por estar ao meu lado, pela paciência, acreditando e sempre me incentivando a superar meus desafios.

À minha irmã Jaqueline Andrea, por sempre confiar em mim e demonstrar que me ama incondicionalmente.

Aos amigos da Gazeta Brandão e FSOP, vocês são meus grandes amigos de fé, meus irmãos camaradas.

Aos meus amigos da FIAT CRHYSLER AUTOMÓVEIS, pelo apoio, pelas palavras de incentivos e por serem tão generosos.

Aos amigos do LITC, pelas conversas, discussões técnicas, ajudas e incentivos, em especial: Luiz Carlos Bambirra Torres, Frederico Gualberto Coelho, Euler Guimarães Horta, Leonardo Jose Silvestre e Frederico Bastos Dantas.

Aos amigos: Kenedy Marconi, Alisson Marques da Silva e Emerson da Silva Almeda.

Aos professores de altíssimo nível que tive durante essa jornada: Antônio de Pádua Braga, Davies William de Lima Monteiro, Guilherme Augusto Silva Pereira, Hani Camille Yehia e João Antônio de Vasconcelos. Agradeço a vocês por compartilharem do conhecimento de vocês e por me guiarem neste caminho de grande reflexão e tanto aprendizado.

Agradeço à UFMG, pela excelência em ensino e por ter me proporcionado durante estes dez anos, entre Disciplinas Isoladas, Mestrado e Doutorado, conhecimentos de tão alto nível.

Ao meu Orientador Professor Antônio de Pádua Braga, obrigado por me acolher como seu orientando, por seu voto de confiança, pelas oportunidades e pelas orientações precisas, que me ajudaram a encontrar meu caminho e a conquistar este grande sonho.

Ao meu Co-orientador Professor André Paim Lemos, muito obrigado pela orientação e pelas discussões técnicas.

Ao meu grande amigo David Menotti, pelas discussões precisas e inspiradoras, pelas diversas ajudas nos momentos difíceis, por sempre me encorajar a cumprir minha meta .

Aos Professores membros da banca: Antônio de Pàdua Braga, André Paim Lemos, Cristiano Leite de Castro, Frank Sill Torres, David Menotti Gomes e Luis Enrique Zárate Gálvez, que contribuíram fortemente com seus valiosos comentários.

E por fim a todos que de alguma maneira contribuíram, incentivaram ou acreditaram neste trabalho.

Sumário

1 Introdução **2**

1.1 O Problema de Aprendizado de Múltiplas Instâncias 4
1.2 Primeiro trabalho de Aprendizado de Múltiplas Instâncias 7
1.3 Definição do problema 8
1.4 Motivações, objetivos e contribuições 8
1.5 Organização deste trabalho 10

2 Revisão Bibliográfica **11**

2.1 Formalização 11
2.2 Abordagens baseadas em modelos generativos 12
2.2.1 Retângulos de Eixos Paralelos - *Axis Parallel Rectangle* (APR) 12
2.2.2 Densidade Diversa - *Diverse Density* (DD) 13
2.2.3 Maximização da Expectativa da Densidade Diversa - *Expectation-Maximization Diverse Density (EM-DD)* 16
2.3 Abordagens baseadas em modelos discriminativos 17
2.3.1 Citação dos K Vizinhos mais Próximos - *Citation kNN* 17
2.3.2 Máquinas de Vetores de Suporte para Aprendizado de Múltiplas Instâncias (mi-SVM and MI-SVM) 19
2.4 Abordagens baseadas em seleção de instâncias 20
2.4.1 SMILE 20
2.4.2 MILIS 21
2.4.3 MILDS 22
2.5 Outras abordagens 22
2.6 Aplicações 23
2.6.1 Recuperação de imagem 24
2.6.2 Aplicações médicas 25
2.6.3 Previsão de mercado de ações 28
2.6.4 Caracterização de texto 28
2.7 Conclusões do capítulo 28

3 Seleção de Instâncias Positivas, via Estimador de Densidade por *Kernel* 30
3.1 Algoritmos propostos . . . 31
3.1.1 Cálculo de probabilidade via Estimador de Densidade por *Kernel* . . . 32
3.1.2 Passo 1: Identificação da Instância Mais Positiva . . . 37
3.1.3 Passo 2: Seleção das instâncias positivas - Qualificação . . . 37
3.1.4 Passo 3: Repescagem . . . 38
3.1.5 LogitBoost . . . 39
3.2 Conclusões do capítulo . . . 40

4 Experimentos, Resultados e Discussões 41
4.1 Base de dados . . . 41
4.1.1 Base de dados MUSK, Elefante, Tigre e Raposa . . . 41
4.1.2 Base de dados COREL . . . 44
4.2 Análises Estatísticas . . . 48
4.2.1 Intervalo de Confiança . . . 48
4.2.2 O teste de Friedman . . . 49
4.3 Resultados . . . 50
4.3.1 Resultados para as Bases MUSK, Elefante, Raposa e Tigre. . . . 50
4.3.2 Resultados para a base COREL . . . 53
4.4 Discussões . . . 57
4.5 Conclusões do capítulo . . . 59

5 Seleção de Instâncias em Expressões Gênicas 60
5.1 O problema de classificação do tipo de leucemia . . . 60
5.2 Aplicação de Aprendizado de Múltiplas Instâncias para o problema de classificação de leucemia . . . 63
5.3 Experimento . . . 64
5.3.1 Resultados . . . 65
5.4 Conclusões do capítulo . . . 68

6 Conclusões e Trabalhos Futuros 69
6.1 Conclusões . . . 69
6.2 Trabalhos futuros . . . 71
6.3 Produção ciêntifica . . . 71

Referências Bibliográficas 73

Lista de Figuras

1.1 Imagem de um quintal contendo um cachorro. 3
1.2 Figura esquemática do Aprendizado Supervisionado - Adaptado de Braga et al. (2011). 4
1.3 Examplo de *bag* e instâncias, aplicado em imagem. 5
1.4 Examplo de *bag* e instâncias, onde *f1*, *f2* e *f3* representam três dimensões (características) para o exemplo. 6
1.5 A forma da molécula muda conforme a rotação das ligações internas. Adaptada de Dietterich et al. (1997). 7

2.1 Exemplo do conceito de Densidade Diversa: a) Cada linha representa uma forma diferente que uma molécula pode assumir, a região A corresponde a interseção dos *bags* positivos onde a molécula apresenta a mesma forma (região de alta densidade de instâncias positivas). b) A região B apresenta um lugar com alta densidade, porém a região A é uma área de alta Densidade Diversa, por estar mais distante das instâncias negativas. Figura adaptada de Maron e Lozano-Perez (1998). 14
2.2 Uma ilustração da Distância de Housdorff Mínima. 18
2.3 Comparação entre as soluções de MI-SVM e mi-SVM, para um espaço de característica 2D. Fonte: Extraída de Babenko (2008). 20
2.4 Ilustração do método para detecção de sintomas de Parkinson, formulado como AMI, Figura adaptada de Das et al. (2012). 26
2.5 Ilustração da análise do ECG por AMI - cada trecho de ECG é considerado um *bag*. 27
2.6 Exemplos de *bags* e instâncias: a) *Bag* positivo (imagem com câncer) e b) *Bag* negativo (imagem sem câncer). Quadrados brancos indicam as instâncias positivas (tecidos com a doença) e quadrados pretos são instâncias negativas (tecidos sem a doença) - Figura extráida de Xu et al. (2012). 27

3.1 Três *bags* positivos ("cachorro"), com instâncias negativas (regiões que não contém o cachorro) não correlatas. 31
3.2 Metodologia proposta. 33

3.3 Dados sintéticos, uniformemente, distribuídos dentro dos limites 0 e 100 em ambas as direções. Amostras positivas são aquelas que estão contidas dentro do quadrado central, centrado em $(50, 50)$ e lados de 5 ± 3, nas duas dimensões. 34
3.4 Probabilidade de uma amostra, dado todos os *bags* positivos: $P(\mathbf{x}_i|B_1^+, B_2^+, \cdots, B_{N^+}^+)$. 35
3.5 Probabilidade de uma amostra, dado todos os *bags* negativos: $P(\mathbf{x}_i|B_1^-, B_2^-, \cdots, B_{N^-}^-)$. 35
3.6 Projeção de todos os dados de entrada no espaço de probabilidades $\prod_j P(\mathbf{x}_i|B_j^-) \times \prod_j P(\mathbf{x}_i|B_j^+)$. Pontos em vermelhos são aqueles dentro do quadrado central, como mostrado na Figura 3.3. 36
3.7 Perfil de diferença de probabilidades para a base MUKS1. 37

4.1 Exemplos de caracteristicas geométricas, extraída para cada molécula - *bag* (Figura extraída de Dietterich et al. (1997).) 42
4.2 Exemplos de imagens da base: Elefante, Tigre e Raposa. 42
4.3 Distribuição média das instâncias para cada *bag*, em cada classe. 43
4.4 Histogramas da quantidade de instâncias por *bags* para as bases MUSK1, MUSK2, Elefante, Tigre e Raposa. 45
4.5 Exemplos da base COREL. Cada linha contém dois exemplos de imagens de duas categorias distintas, são elas: Africa, Praias, Prédios, Ônibus, Dinossauros, Elefantes, Flores, Cavalos, Montanhas, Comidas, Cachorros, Largatos, Modelos, Por do sol, Carros, Cascatas, Móveis antigos, Navios de batalhas, Ski e Desertos. 47
4.6 Intervalo de Confiança - Gráfico resumo da Tabela 4.4. As áreas hachuradas representam os resultados para as metodologias MILKDE e MILKDE2, para as bases de dados MUSK1 e MUSK2. 54
4.7 Intervalo de Confiança - Gráfico resumo dos cálculos do Intervalo de Confiança para os resultados da Tabela 4.5. As áreas hachuradas representam os resultados para a metodologia MILKDE aplicada à base COREL. 56
4.8 Gráfico radar comparativo entre a metodologia proposta MILKDE e alguns algoritmos da literatura. 58

5.1 Evolução de uma célula sanguínea. Figura inspirada na original disponível no Instituto Nacional do Câncer. 61
5.2 Adaptação do problema de Leucemia para abordagem MILKDE. 64

Lista de Tabelas

4.1 Base de dados: MUSK, Elefante, Raposa e Tigre. 43
4.2 Características da base de dados COREL. 46
4.3 Resultados para o MILKDE, MILKDE2 e outros algoritmos disponíveis na literatura: acurácia média (%), para 10 × 10 *fold* de validação cruzada. 52
4.4 Resultados: acurácia média (%), desvio padrão (%) para 10 × 10 *fold* de validação cruzada (em parênteses) e o intervalo de confiança de 95 % (em colchete). 53
4.5 Acurácia Média (%) e Desvio Padrão (%) para a Base COREL 2000 Imagens. . . . 55
4.6 Resultados para a base COREL: Acurácia média (%) para base com 2000 Imagens. 55
4.7 Análise das diferenças de *Rank*, de acordo com teste de Friedmam para a base Corel 2000 imagens. 56

5.1 Resultados para a metodologia proposta. Base de teste - 34 pacientes (20 ALL e 14 AML). Para o primeiro teste é apresentado a Acurácia (%) e para o segundo a Acurácia média (%) e o desvio padrão (%). 66
5.2 Resultados da literatura. Base de teste - 34 pacientes (20 ALL e 14 AML) 66
5.3 50 Genes mais relevantes, selecionados pelo método de *Gini*. Os genes em negritos estão também presentes no trabalho de Golub et al. (1999). 67

Capítulo 1

Introdução

prendizagem pode ser definida como o ato de aprender, de adquirir conhecimento, competências ou habilidades. O processo de aprendizagem estabelece ligações entre estímulos de entradas e suas respectivas respostas, com impacto direto ao aumento da adaptação e percepção do ser ao seu meio. A aprendizagem é capaz de mudar o indivíduo em virtude da experiência à qual este foi submetido.

Tanto seres humanos, animais e sistemas inteligentes artificiais são capazes de aprender por meio de exemplos, cada um com seu próprio nível de complexidade. Para os seres humanos, de acordo com o filósofo inglês John Locke (1632-1704) (Locke, 1690), todo conhecimento humano provém de suas experiências que são obtidas por meio dos seus sentidos. Segundo Locke, todos os seres humanos, ao nascerem, são como uma folha em branco *i.e.*"tábula rasa". E essa folha é escrita com o tempo, na medida em que experimentamos e sentimos o mundo. Ainda de acordo com Locke, DEUS nos concedeu somente as faculdades, ou habilidades, de adquirir conhecimento, de inferir regras, com base em exemplos e experiências, e, por meio desse dom, temos a oportunidade de aprender

Já nos primeiros meses de vida, os seres humanos começam a aprender por meio de exemplos que são lhes mostrados diariamente: "mamãe", "papai", "mamadeira", "bola" *etc.* E, para cada palavra, o bebê é estimulado com uma imagem única e bem distinta para sua fácil percepção e compreensão. À medida que os bebês se desenvolvem, formas de aprendizado mais complexas lhes serão apresentadas: em vez de um objeto único relacionado com uma palavra, lhe são apresentadas situações ou ambientes, nos quais um objeto está inserido, por exemplo: "um cavalo no campo","um carro na rua, "um cachorro no quintal", "na televisão está passando Peppa" *etc.* Este é um passo muito importante para o aprendizado, pois necessita de uma percepção e de uma interpretação mais complexa do cérebro. Quando lhe é mostrada, pela primeira vez, um quintal com um cachorro, como apresentado na Figura 1.1, o que, naquela cena, realmente, é o cachorro? A grama? A casinha do cachorro? A árvore? A cerca? O objeto branco de quatro patas e uma cauda? Todas as respostas estariam corretas para aqueles

Figura 1.1: Imagem de um quintal contendo um cachorro.

que fossem apresentados, pela primeira vez, aquele cenário. Com o passar do tempo e com base na observação de outros cachorros, a criança começa a inferir o que, naquelas prévias cenas, que ela viu sobre cachorro, têm em comum.

Podemos chamar de ambígua essa última forma de aprendizagem, pois o objeto de aprendizado não apresenta uma especificação única e distinta, quando é vista sozinha. Essa forma de aprendizagem, ambígua, não é exclusividade dos seres humanos e podemos extrapolar esses conceitos, também, para o ambiente computacional.

Na área de Inteligência Artificial, o subcampo de Aprendizado de Máquina é responsável pelo desenvolvimento de algoritmos que permitem o processo de aprendizagem para sistemas computacionais. Os algoritmos de Aprendizado de Máquina buscam inferir um modelo, com base nos dados que lhe são apresentados e buscam, com base nesse modelo, serem capazes de predizer os rótulos de dados futuros de entrada.

Entre as principais abordagens de aprendizado de máquina, podem ser destacadas três abordagens: aprendizado supervisionado, aprendizado não supervisionado e aprendizado por reforço. Dentre esses, o modelo que é realizado com base em exemplos e seus rótulos é chamado de aprendizado supervisionado. Em relação a esse tipo de aprendizado, existe o papel do "professor" que é responsável por apresentar para o modelo exemplos (vetores de características) e seus respectivos rótulos, conforme apresentado na Figura 1.2, adaptada de Braga et al. (2011). A diferença entre as duas respostas: a desejada e a obtida pelo modelo, é

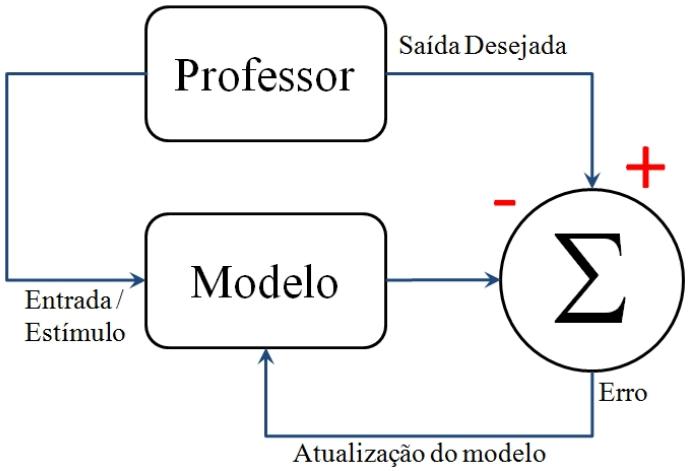


Figura 1.2: Figura esquemática do Aprendizado Supervisionado - Adaptado de Braga et al. (2011).

utilizada pelo professor para que se faça ajustes nos parâmetros internos do modelo, de forma que ele possa melhorar sua previsão nos estímulos futuros. Esses modelos devem ser robustos o bastante para que possam generalizar as informações e para que sejam capazes de, para exemplos jamais vistos, predizer seus rótulos.

Quando se trabalha com aprendizado supervisionado (AS), cada amostra $\mathbf{x}_i$, no conjunto de treinamento $D_L = \{\mathbf{x}_i, y_i\}_{i=1}^{N}$ é associada a um rótulo de classe $y_i \in \{-1, +1\}$, onde $\mathbf{x}_i \in \mathcal{R}^p$, na qual p é o número de características que define uma instância e N o número de instâncias. Assim, existe uma relação uma a uma entre os vetores de entrada e os rótulos da classe.

No entanto, existem situações em que esse tipo de treinamento não é totalmente factível e rotular cada instância, na fase de treinamento, não é possível. Aprendizado de Múltiplas Instâncias (AMI), ou *Multiple Instance learning (MIL)* é o paradigma para formalizar esse tipo de problema. Em AMI (Dietterich et al., 1997), um conjunto de instâncias (*bags*) é rotulado, em vez de instâncias individuais, o que representa um desafio maior para os algoritmos de aprendizados tradicionais, já que a maioria deles recaem na relação $\mathbf{x}_i \rightarrow y_i$, a fim de induzir funções de mapeamento.

1.1 O Problema de Aprendizado de Múltiplas Instâncias

Nem sempre a associação entre uma única instância e seu rótulo é factível nos trabalhos de classificação. Essa situação nos leva a um novo paradigma de aprendizado supervisionado chamado Aprendizado de Múltiplas Instâncias (AMI).

Esse novo paradigma de aprendizagem foi descrito, originalmente, por Dietterich et al. (1997), no contexto do problema de previsão de atividades de drogas e, posteriormente, apli-

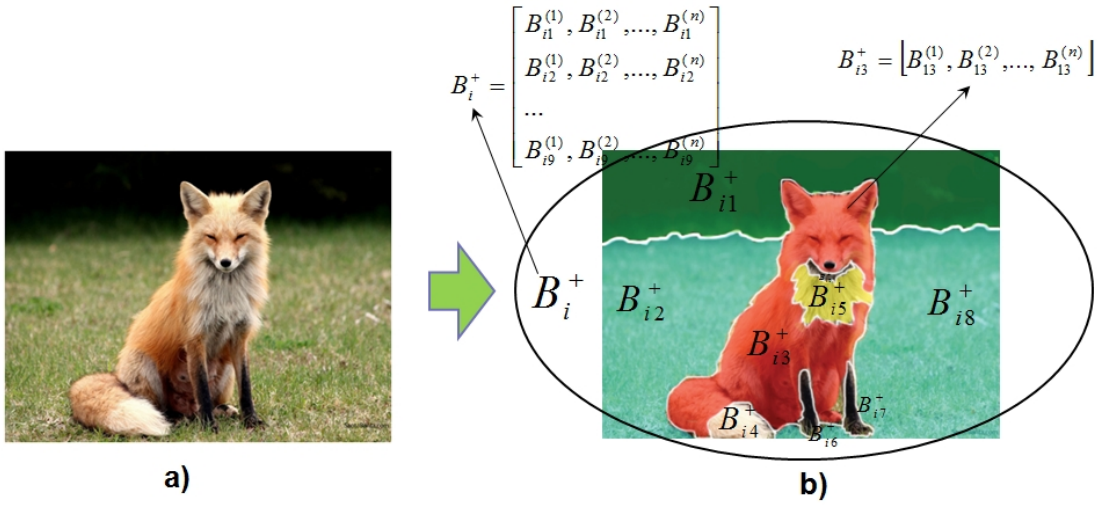


Figura 1.3: Examplo de *bag* e instâncias, aplicado em imagem.

cado em diversos problemas como: classificação e recuperação de imagem (Maron e Ratan, 1998; Li et al., 2012a; Wang et al., 2012; Yao et al., 2010; Liu et al., 2010; Zhang et al., 2005; Chen e Wang, 2004a), áreas médicas (Das et al., 2012; Sun et al., 2012; Mahnot e Popescu, 2012), mercado de ações (Maron e Lozano-Perez, 1998), classificação e reconhecimento de texto (Andrews et al., 2003; Keshtkar e Inkpen, 2011; Wei e Yu, 2009; S. e M., 2005), entre outros.

Em Andrews et al. (2003); Maron e Lozano-Perez (1998); Babenko et al. (2011), AMI é definido como uma generalização do aprendizado supervisionado, na qual rótulos são associados a um conjunto de instâncias, ou *bags* e não instâncias individuais. A Figura 1.4 exemplifica essa situação, na qual os rótulos são associados aos *bags* e não às instâncias individualmente. A ideia principal para o AMI é a forma como o *bag* é rotulado: Um *bag* é rotulado como negativo se é conhecido que todas suas instâncias são negativas e o *bag* é rotulado positivo se é de conhecimento prévio que pelo menos uma instância, dentro daquele *bag*, é positiva.

Um exemplo mais prático, sobre *bags* e instâncias, é apresentado na Figura 1.3. A Figura 1.3 apresenta uma imagem de uma raposa (1.3 a), que ao ser analisada sobre a ótica de Aprendizado de Múltiplas Instâncias (1.3 b), podemos considerá-la como sendo um *bag* positivo, pois a mesma contém o objeto de interesse *i.e.* Raposa, inserido na foto. Podemos considerar esta imagem ou *bag*, como uma coleção de instâncias ou regiões, cada uma identificada de uma cor em específica (1.3 b). A rotulação das regiões são assim distribuídas: Positivas, aquelas regiões que contém parte do animal (3, 4, 5, 6 e 7) e Negativas, aquelas que não contém o animal (1, 2 e 8). Ainda, cada uma destas instâncias é formada por um vetor de característica, que neste caso são descritores de imagem. É válido relembrar que neste caso didático, conhecemos a rotulação das instâncias dentro do *bag*, fato que não acontece em bases reais.

Então, para um conjunto de *bags* rotulados como positivos ou negativos, o classificador

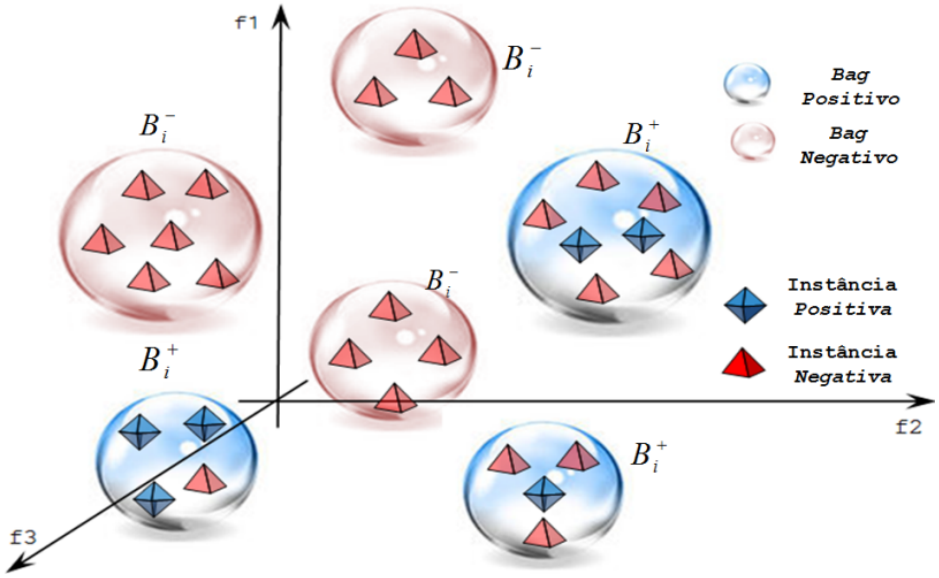


Figura 1.4: Examplo de *bag* e instâncias, onde *f1*, *f2* e *f3* representam três dimensões (características) para o exemplo.

busca inferir uma regra ou uma função que determina a classe de um *bag* não conhecido. Uma solução para esse problema seria identificar uma ou mais características comuns entre as instâncias do *bag* positivo.

No AMI, um *bag* positivo B_i^+ contém ao menos uma instância positiva, e *bags* negativos B_i^- contém somente instâncias negativas. Assim, instâncias negativas dentro dos *bags* negativos são de rótulos conhecidos, enquanto os rótulos das instâncias dos *bags* positivos não são conhecidas.

Com o intuito de melhor compreender essa problema, vamos ilustrar a definição de AMI, considerando-se o seguinte exemplo adaptado de Babenko (2008) e Dietterich et al. (1997): "Em uma empresa existem diversos empregados, e cada um possui um chaveiro com inúmeras chaves presas a este chaveiro e estas chaves abrem diversas portas. É de conhecimento que alguns empregados possuem uma chave especial que permite acesso à sala de café e outros empregados não possuem acesso a esta sala. Vamos assumir que um Chaveiro profissional é contratado para tentar inferir qual é a chave que abre a sala de café. O que torna difícil o trabalho do Chaveiro, é que os empregados não são muito cooperativos. Em vez de entregar ao Chaveiro a chave exata que abre a sala de café, eles simplesmente entregam o molho de chave, e cabe ao Chaveiro descobrir qual ou quais chaves abrem a porta da sala de café". Para resolver essa família de problema, é necessário descobrir qual ou quais chaves são comuns em todos os molhos de chave positivo (que contêm a chave que abre a porta desejada).

Na literatura podemos encontrar diversos algoritmos para tratar o problema de AMI, podendo destacar os seguintes: *Axis Parallel Concepts* (APR) (Dietterich et al., 1997), *Diverse*

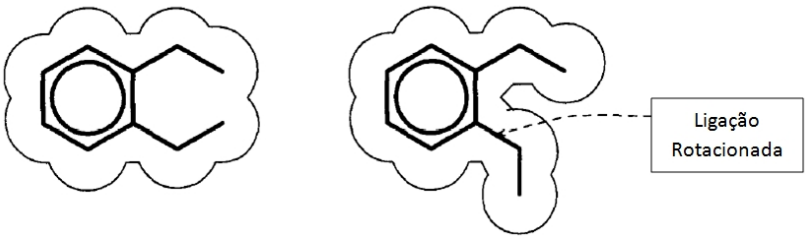


Figura 1.5: A forma da molécula muda conforme a rotação das ligações internas. Adaptada de Dietterich et al. (1997).

Density (DD) (Maron e Lozano-Perez, 1998), *Expectation-Maximization Diverse Density* (EM-DD) (Zhang e Goldman, 2001), *Citation k Nearest Neighbor* (kNN) (Wang e Zucker, 2000), Máquinas de Vetores de Suporte para AMI (mi-SVM e MI-SVM) (Andrews et al., 2003), *Multiple Instance Learning via Embedded Instance Selection* (MILES) (Chen et al., 2006a), *Similarity Based Multiple Instance Learning* (SMILE) (Xiao et al., 2010a, 2014), *Multiple Instance Learning with Instant Selection* (MILIS) (Fu et al., 2011), *Multiple Instance Learning via Dominant Set* (MILDS) (Erdem e Erdem, 2011), *Elipsoidal Multiple Instance Learning* (eMIL) (Krummenacher e Ong, 2013), *Diverse Density with Support Vector Machine* (DD SVM) (Chen e Wang, 2004a), e *Optimization on Multiple Instance Learning via Embedded Instance Selection* (OptMILES) (Zhao et al., 2013). Alguns desses algoritmos serão explorados no próximo capítulo.

1.2 Primeiro trabalho de Aprendizado de Múltiplas Instâncias

O conceito de Aprendizado de Múltiplas Instâncias foi originalmente proposto por Dietterich et al. (1997), quando realizava trabalhos relacionados à previsão de atividades em drogas. O objetivo era determinar se uma molécula era qualificada para realizar uma determinada droga ou não, dado um conjunto de moléculas. De acordo com Dietterich et al. (1997), uma molécula pode possuir diversas formas, devido a rotações de suas ligações internas, como apresentado na Figura 1.5. Cada uma dessas formas pode apresentar diferentes propriedades, como o potencial de energia que é determinado pelas interações entre os átomos, que determinam se a molécula pode ser "MUSK" ou "NÃO MUSK".

Para o problema de atividade de droga, moléculas ativas irão emitir um odor. Dessa forma a molécula ativa irá emitir um cheiro de Almíscar [1] somente se uma de suas formas emitir esse cheiro. No entanto qual forma da molécula que é responsável por ela ser MUSK não é conhecida. Então, o problema de aprendizado é predizer se um conjunto de moléculas irá emitir ou não este cheiro.

[1] do inglês MUSK

Baseado nessa formulação do problema, Dietterich et al. (1997) propuseram o paradigma de Aprendizado de Múltiplas Instâncias para resolver o problema. Dessa forma, cada molécula seria um *bag*, cada uma das suas formas seria uma instância e as dimensões seriam as características extraídas de cada uma das suas formas. Essas características foram determinadas como raios provindos de sua origem, após padronizar a posição e orientação da molécula, de forma a mostrar mais uniformemente a molécula. No caso dessas moléculas, foram extraídas 162 distâncias (dimensões) mais quatro características para representar a posição de um átomo de oxigênio na molécula.

1.3 Definição do problema

De acordo com Maron e Lozano-Perez (1998), Aprendizado de Múltiplas Instâncias pode ser considerado como um tipo de aprendizado ambíguo, o qual os exemplos não são perfeitamente ou completamente rotulados, gerando, assim, ambiguidades nas classes das instâncias. Essa característica de AMI faz com que seja um grande desafio desenvolver algoritmos para resolver essa classe de problema, pois os rótulos das instâncias, separadamente, não são conhecidos. Como será visto no capítulo 2, várias abordagens de algoritmos são propostas na literatura para trabalhar com AMI, todas com objetivo de encontrar uma estratégia para trabalhar com essa ambiguidade. Assim, podemos definir o problema a ser tratado nesta tese como:

"Dado um conjunto de dados, rotulados de forma ambígua, ou seja, os rótulos são atribuídos a um conjunto de instâncias (bags) e não para cada uma separadamente, construir um modelo que seja capaz de prever se um novo bag é positivo ou negativo, com base nas premissas de Aprendizado de Múltiplas Instâncias ".

1.4 Motivações, objetivos e contribuições

Ao analisar os resultados das metodologias existentes na literatura, apresentado na Tabela 4.3 no capítulo Experimentos, constata-se que embora exista uma quantidade expressiva de metodologias para se tratar o problema de AMI, nenhum deles possuem resultados que se destacam em mais de uma base de dados. Temos algoritmos com o SMILE, MILDS, MILES, MILIS, GPMIL *etc.*, que possuem bons resultados, porém só conseguiram o melhor resultado para uma base em específico. Isso nos leva a assumir que a grande maioria das metodologias são mais específicos e não se destacam para mais de um tipo de base de dados.

Essa análise nos conduz a procurar metodologias mais generalistas que possam atingir resultados competitivos em mais de uma base de dados. Esta é a grande motivação deste trabalho, desenvolver uma metodologia competitiva com as já apresentados na literatura e que possa alcançar resultados promissores em um maior número de bases.

Outro aspecto das metodologias já existentes na literatura é que muitas se baseiam na distribuição da classe negativa com intuito de inferir um padrão nessa distribuição e assim

escolher melhor as instâncias realmente positivas. Essa abordagem, embora tenha se mostrado eficiente nas bases testadas, são paradoxais com o conceito original de Aprendizado de Múltiplas Instâncias. Esse paradoxo é devido à natureza das instâncias das classes negativas que podem representar qualquer coisa e não, necessariamente, ter coerência entre si.

Portanto, o propósito principal deste trabalho consiste em desenvolver uma metodologia que seja mais generalista, atingindo resultados competitivos e equilibrados em mais de uma base de dados e não em uma classe específica de problema. Além disso, a metodologia deve respeitar melhor os conceitos de AMI, utilizando a classe negativa para realizar estudos da distribuição da base, mas não usar essa distribuição como critério único para seleção de instâncias.

Entre os principais objetivos deste trabalho, destacam-se:

- Pesquisar, conhecer e estudar o problema de Aprendizado de Múltiplas Instâncias, bem como suas hipóteses e premissas. O objetivo principal deste estudo é fornecer embasamento teórico para propor uma nova metodologia para tratar o problema;
- Uma revisão das principais metodologias para Aprendizado de Múltiplas Instâncias, suas abordagens principais e suas aplicações práticas;
- Desenvolver um algoritmo consistente que possa contribuir, de forma eficaz, para a solução de problemas envolvendo Aprendizado de Múltiplas Instâncias, com resultados competitivos com os melhores algoritmos apresentados da literatura. Esta metodologia deverá apresentar resultados satisfatório para a maioria das bases de dados envolvendo AMI;
- Empregar a metodologia proposta para Aprendizado de Múltiplas Instâncias em uma nova aplicação, voltada à problemas na área de Bioinformática.

Como principais contribuições deste trabalho, podem-se destacar:

- Nesta tese foram propostos duas novas metodologias para Aprendizado de Múltiplas Instâncias, baseados em seleção de instâncias positivas, que demostraram ser uma boa abordagem, com resultados competitivos aos da Literatura;
- Foi realizado, durante o trabalho, um estudo estatístico não presente em nenhum trabalho na literatura, entre as principais metodologias da literatura;
- Utilizando a metodologia desenvolvido nesta tese, foi proposta uma nova aplicação para AMI em uma base conhecida de Leucemia. Com a utilização da proposta foi possível selecionar instâncias (pacientes) mais representantes para tipo de leucemia, durante a fase de treinamento. Os resultados obtidos comprovaram que, com o uso da metodologia, foi possível atingir resultados ligeiramente melhores, utilizando a metade dos pacientes de treinamento;

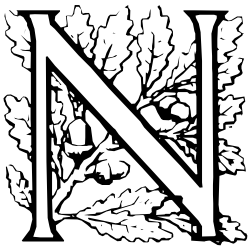
- Partes dos desenvolvimentos tratados nesta tese foram publicadas em Faria et al. (2014) (aprensentada no *European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning* Bélgica/2014) e também uma publicação na revista JCB (Faria et al., 2015). Além desses artigos, outro foi submetido à revista *Engineering Applications of Artificial Intelligence*, pendente ainda de aceitação.

1.5 Organização deste trabalho

O restante deste trabalho está organizado da seguinte forma: No capítulo 2, são apresentadas e discutidas as principais abordagens existentes na literatura visando resolver os problemas de AMI, bem como algumas aplicações do método. A metodologia proposta juntamente com suas premissas serão tratadas no capítulo 3. Toda a etapa de experimentos e resultados da metodologia proposta em confronto com os algoritmos existentes na literatura está descrita no capítulo 4. O capítulo 5 apresenta o problema de classificação de tipos de Leucemia e uma adaptação do método proposto visando selecionar os pacientes mais representativos para cada classe. Para finalizar o trabalho, são relatadas no capítulo 6 as conclusões finais e trabalhos futuros a serem desenvolvidos.

Capítulo 2

Revisão Bibliográfica



este capítulo são apresentadas e discutidas as principais abordagens visando o Aprendizado de Múltiplas Instâncias (AMI). Nos últimos anos, diversos métodos para resolver problemas de AMI têm sido desenvolvidos utilizando-se diversas técnicas e inspirações. Buscando agrupar esses métodos de maneira a representar a sua maioria, podem-se destacar três grupos: Abordagens baseadas em Modelos Generativos, baseadas em Modelos Discriminativos e Seleção de Instâncias. Esses três grupos já foram propostos e são bem utilizados na literatura (Erdem e Erdem, 2011; Fu et al., 2011; Xiao et al., 2010b; Zafra et al., 2010).

Inicialmente, a Seção 2.1 apresenta a formalização e notações utilizadas para descrever os métodos apresentados neste trabalho, as Seções 2.2, 2.3 e 2.4 descrevem os principais algoritmos baseados em Modelos Generativos, Discriminativos e os algoritmos baseados em Seleção de Instâncias, respectivamente. Outras abordagens são apresentadas na Seção 2.5 e, finalmente, a Seção 2.6 descreve algumas das principais aplicações para AMI.

2.1 Formalização

Esta Seção apresenta a formalização e as notações utilizadas neste trabalho para o problema de Aprendizado de Múltiplas Instâncias.

Seja definido B_i como o i-ésimo *bag* do conjunto de dados B, *i.e.*, $B = \{B_1, B_2, ..., B_l\}$, com $1 \leq i \leq l$, em que l representa o número de *bag*s no conjunto de dados. Também seja definido B_{ij} como a j-ésima instância do *bag* B_i, *i.e.*, $B_i = \{B_{i1}, B_{i2}, ..., B_{ib_i}\}$, com $1 \leq j \leq b_i$, e b_i como o número de instâncias do *bag* B_i. Define-se, ainda, B_{ijk} como a k-ésima característica da instância B_{ij}.

Um *bag* positivo é representado por B_i^+ e um *bag* negativo por B_i^-. Uma instância positiva e uma instância negativa são representadas por B_{ij}^+ and B_{ij}^- respectivamente. Então, $B =$

$\{B_1^+, B_2^+, \ldots, B_n^+, B_1^-, B_2^-, \ldots, B_m^-\}$ representa um conjunto de *bags* positivos e negativos, e n e m o número de *bags* positivos e negativos, respectivamente, em que $n + m = l$.

Mais formalmente, o rótulo de cada *bag* é representado por $L(B_i) \in \{+1, -1\}$. Em outras palavras, $L(B_i) = +1$ (ou B_i^+) se existir ao menos uma instância positiva em B_i (B_{ij}^+), e $L(B_i) = -1$ (or B_i^-) caso contrário. Isto é, $L(B_i) = +1, \exists j : B_{ij}^+, 1 < j < b_i$, e $L(B_i) = -1$, caso contrário.

Uma propriedade de grande relevância para a formulação do Aprendizado de Múltiplas Instâncias consiste no conceito que cada *bag* B_i pode ser composto por um número diferente de instâncias, porém cada instância deve possuir a mesma dimensão, *i.e.*, número de características.

2.2 Abordagens baseadas em modelos generativos

Os modelos generativos buscam identificar um ponto conceito, determinando uma região no espaço de características, a qual visa conter todas as instâncias realmente positivas, enquanto afasta todas as instâncias negativas dessa região. Entre os principais métodos com essa abordagem, encontrados na literatura, se destacam o Retângulos de Eixos Paralelos (APR) (Dietterich et al., 1997), Densidade Diversa (DD) (Maron e Lozano-Perez, 1998) e a Maximização da Expectativa para Densidade Diversa (EM-DD) (Zhang e Goldman, 2001), detalhados a seguir.

2.2.1 Retângulos de Eixos Paralelos - *Axis Parallel Rectangle* (APR)

O algoritmo Retângulo de Eixos Paralelos ou *Axis Parallel Hyper Rectangle* foi a primeira classe de algoritmo proposto para resolver problemas de AMI. Esse algoritmo foi, originalmente, proposto por Dietterich et al. (1997), aplicado ao problema de predição de atividade de drogas. A ideia principal do algoritmo é encontrar um Hiper Retângulo de Eixos Paralelos (APR), no espaço de característica para representar o ponto conceito. Esse APR deve conter ao menos uma instância de cada *bag* positivo, enquanto todas as instâncias dos *bags* negativos estejam fora. Dietterich et al. (1997) propôs três algoritmos diferentes para encontrar esse APR.

- Algoritmo Padrão: Também conhecido como Naïve (ingênuo) APR, essa abordagem encontra o menor APR que contenha todas as intâncias dos *bags* positivos.
- Algoritmo *Outside-in*: essa abordagem constrói o menor APR que contenha todas as instâncias dos *bags* positivos e começa a encolher o APR, de modo a excluir todas as instâncias negativas.

- Algoritmo *Inside-out*: começa com um ponto inicial (*seed*), no espaço de característica, e começa a crescer o APR objetivando encontrar o menor APR que contenha pelo menos uma instância de cada *bag* positivo e nenhuma instância do *bag* negativo.

Dietterich et al. (1997) descrevem que, devido ao método "*inside-out*" ser hábil de identificar mais características relevantes, este é melhor que o método "*outside-in*". Assim o método "*inside-out*" obtém melhores resultados entre os demais métodos. O Algoritmo Padrão que procura ignorar o problema de múltiplas instâncias obteve os piores resultados, com isso reforçando que o conceito de Problemas de Múltiplas Instâncias (PMI) deve ser considerado

2.2.2 Densidade Diversa - *Diverse Density* (DD)

Buscando, também, identificar uma região no espaço de características, surgiu o algoritmo Densidade Diversa-DD originalmente proposto por Maron e Lozano-Perez (1998). Em seu trabalho, Maron e Lozano-Perez (1998) descreve DD como uma medida de intersecção dos *bags* positivos menos a união dos *bags* negativos.

A principal ideia do DD (Maron e Ratan, 1998) (Yang, 2005) é encontrar um ponto conceito no espaço de característica, que seja próximo de no mínimo uma instância de cada *bag* positivo enquanto seja distante das instâncias dos *bags* negativos. Yang (2005) define o ponto conceito ótimo como o ponto dentro da máxima DD, o qual é a medida de quantos *bags* positivos diferentes possuem instâncias próximo ao ponto conceito e o quão distante das instâncias negativas estão distante daquele ponto. O algoritmo, então, procura, no espaço de características, dentre todos os pontos, aquele ponto que possua a máxima Densidade Diversa. Uma vez encontrado o ponto, com máxima DD, um novo *bag* é classificado positivo se pelo menos uma instância estiver próxima do ponto de máximo DD. Segundo Chen et al. (2006b), DD mede a co-ocorrência de instâncias similares de *bags* positivos diferentes. Então, o conceito desejado é encontrado pela maximização da função DD.

Em seu artigo original Maron e Lozano-Perez (1998), define seu algoritmo por meio de um exemplo de uma molécula: "Suponha que a forma de uma molécula, candidata a realizar uma ligação a uma dada proteína, possa ser descrita como um vetor de características. Uma instância da molécula então, é representada com um ponto no espaço n-dimensional de característica. Como cada molécula muda sua forma, será projetado suas características no espaço de característica, a qual essa projeção representará um caminho, ou uma curva, para cada molécula."

Se uma molécula candidata é rotulada positiva, sabe-se que ao menos um lugar do seu 'caminho" no espaço de características assumirá a forma correta para ajustar a proteína desejada. Se a molécula é rotulada negativa, em nenhuma dos seus pontos sobre seu "caminho" será permitida a ligação com a proteína desejada. Para esse exemplo, assume-se que existe somente uma forma de molécula que ligará com a proteína desejada. Então, a forma correta

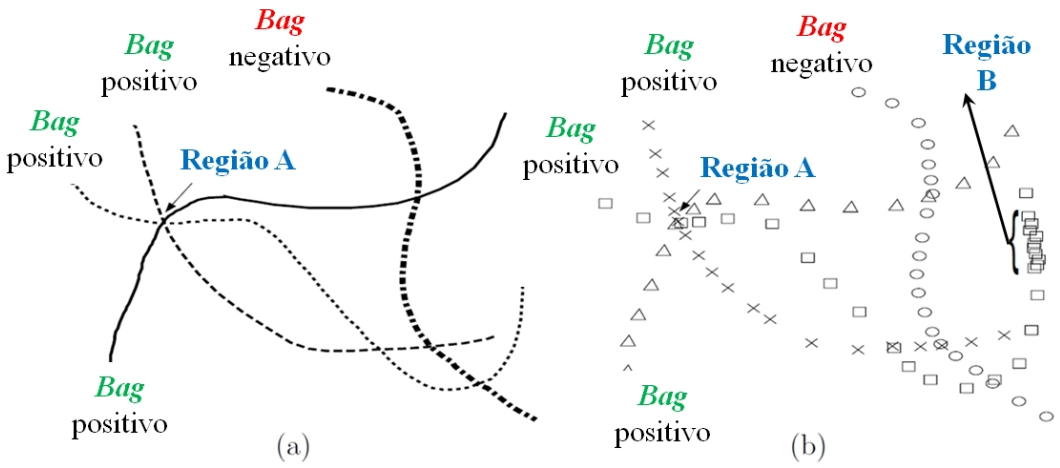


Figura 2.1: Exemplo do conceito de Densidade Diversa: a) Cada linha representa uma forma diferente que uma molécula pode assumir, a região A corresponde a interseção dos *bags* positivos onde a molécula apresenta a mesma forma (região de alta densidade de instâncias positivas). b) A região B apresenta um lugar com alta densidade, porém a região A é uma área de alta Densidade Diversa, por estar mais distante das instâncias negativas. Figura adaptada de Maron e Lozano-Perez (1998).

estará onde todas as características, dos diversos caminhos das moléculas positivas se interseccionam sem interseccionar nenhuma característica negativa. Assim, o problema de descobrir a intersecção dos caminhos altera-se para um problema relativo à busca de uma região onde exista alta densidade de pontos positivos e baixa densidade de pontos negativos.

Portanto, Densidade Diversa é definido como um ponto que mede o quanto *bags* positivos diferentes possuem instâncias próximos a esse ponto conceito, ou região, e o quão distante de instâncias negativas está esse ponto, como ilustrado na Figura 2.1, adaptada de Maron e Lozano-Perez (1998).

Embora DD seja aplicado em diversos trabalhos na literatura, Fu e Kelly (2009) ressaltam que um problema com esse método é o *trade-off* entre eficiência e desempenho. Em seu trabalho, Fu e Kelly (2009) relatam que métodos generativos como DD e EM-DD (o algiritmo EM-DD será descrito na próxima seção), são bem eficientes no aprendizado, mas ambos assumem que todas as instâncias positivas formam um *cluster* compacto no espaço de características. O que, em caso de aplicações reais, não é necessariamente verdadeiro, pois as distribuições das instâncias positivas podem ser arbitrárias e provavelmente multimodais.

Formulação do Densidade Diversa

Nessa seção, é descrita a formulação do algoritmo Densidade Diversa, usando a mesma nomenclatura utilizada por Maron e Lozano-Perez (1998): os *bags* positivos são descritos por B_i^+, a instância j-ésima contido no *bag* como B_{ij}^+, e o valor da característica k-ésima desse ponto

como B^+_{ijk}. Da mesma forma, B^-_{ij} representa uma instância de um *bag* negativo. Assumindo que o ponto conceito é um ponto único t, pode-se encontrá-lo por meio da maximização da probabilidade: $P(x = t|B^+_1, ..., B^+_n, B^-_1, ..., B^-_m)$, sobre todos os pontos $x_i \in \mathbb{R}^d$ no espaço de características.

Utilizando a regra de Bayes (Duda et al., 2000) sobre a localização do ponto conceito, isto é equivalente a maximizar a probabilidade $P(B^+_1, ..., B^+_n, B^-_1, ..., B^-_m|x = t)$. Realizando suposições adicionais, que os *bags* são condicionalmente independentes dado um ponto conceito *"target"* t, a melhor hipótese é $arg \max_x \prod_i P(B^+_i|x = t) \prod_i P(x = t|B^-_i)$. Utilizando-se a regra de Bayes, novamente, obtém-se a seguinte equação:

$$arg \max_x \prod_i P(x = t|B^+_i) \prod_i \P(x = t|B^-_i) \tag{2.1}$$

A Equação 2.1 é a definição geral da máxima Densidade Diversa, porém é necessário definir os termos em produtos para poder instanciá-los. Uma possibilidade é o modelo *noisy-or* (Parsons e Bigham, 1996), para modelar a probabilidade de uma instância em um *bag* ser o ponto conceito, que é dada por: $P(x = t|B^+_i) = P(x = t|B^+_{i1}, B^+_{i2}, B^+_{i3}, ...) = 1 - \prod_j (1 - P(x = t|B^+_{ij}))$, e da mesma forma $P(x = t|B^-_i) = P(x = t|B^-_{i1}, B^-_{i2}, B^-_{i3}, ...) = \prod_j (1 - P(x = t|B^-_{ij}))$.

Se for assumido que cada instância em um *bag* possui seu rótulo próprio e que cada instância pode ser vista como uma causa independente da rotulação do *bag*, o modelo *noisy-or* pode ser aplicado. Analisando-se o problema, a probabilidade de um *bag* ser positivo é equivalente a probabilidade que nem todas as instâncias dentro do *bag* seja negativa. E a probabilidade de um *bag* ser negativo é equivalente à probabilidade que todas as instâncias dentro de um *bag* ser negativa. Então, a contribuição de um *bag* positivo é dada por: $P(x = t|B^+_i) = P(x = t|B^+_{i1}, B^+_{i2}, B^+_{i3}, ...) = 1 - \prod_j (1 - P(x = t|B^+_{ij}))$, e a contribuição de um *bag* negativo é dada por $P(x = t|B^-_i) = P(x = t|B^-_{i1}, B^-_{i2}, B^-_{i3}, ...) = \prod_j (1 - P(x = t|B^-_{ij}))$.

Assim, o desafio é modelar a probabilidade de cada instância $P(x = t|B_{ij})$. Os autores propõe modelar a probabilidade associada à distância entre o ponto conceito e a instância em análise, no espaço de característica. Densidade Diversa utiliza a distância euclidiana como métrica para essa tarefa. Dessa forma, se uma instância é próxima do ponto conceito t, a probabilidade $P(x = t|B_{ij})$ é alta.

Assim o desafio é modelar a probabilidade de cada instância $P(x = t|B_{ij})$. Os autores propõe modelar a probabilidade associada a distância entre o ponto conceito e a instância em análise, no espaço de característica. Densidade Diversa utiliza a distância euclidiana como métrica para esta tarefa, dessa forma, se uma instância é próxima do ponto conceito t, a probabilidade $P(x = t|B_{ij})$ é alta.

A probabilidade de cada instância é dada pela Equação 2.2

$$P(x = t|B_{ij}) = exp(-||B_{ij} - x||^2) \tag{2.2}$$

na qual

$$||B_{ij} - x||^2 = \sum_k s_k^2 (B_{ij} - x_k)^2 \tag{2.3}$$

E k representa a k-ésima caracteristica, B_{ijk} denota o valor da k-ésima característica no ponto B_{ij}, x é o ponto conceito e s_k representa a contribuição (peso) de cada instância.

2.2.3 Maximização da Expectativa da Densidade Diversa - *Expectation-Maximization Diverse Density (EM-DD)*

Com objetivo de aprimorar o algoritmo Densidade Diversa, principalmente em relação à velocidade de operação, foi proposto o algoritmo Maximização da Expectativa da Densidade Diversa EM-DD. EM-DD foi, originalmente, proposto por Zhang e Goldman (2001) e é uma extensão do algoritmo *Diverse Density* padrão. Embora EM-DD utiliza a mesma base teórica que o DD, EM-DD possui um método diferente de encontrar o ponto conceito mais provável. EM-DD combina o algoritmo Maximização da Expectativa junto com o DD original, utilizando o *most-likely-cause*, sugerido por Maron e Lozano-Perez (1998).

O algoritmo EM-DD inicializa com uma semente para o ponto conceito, que pode ser utilizado o algoritmo DD padrão para esse fim e, repetidamente, realiza dois passos:

- Passo E: uma instância é selecionada de cada *bag* positivo, que seja a mais provável de ser responsável pelo rótulo do *bag*, dada uma hipótese inicial. Para isso é utilizado o modelo *most-likely-cause*. A hipótese atual do ponto conceito t é usada para selecionar a instância mais provável de cada *bag*.

- Passo M: é calculado o gradiente ascendente baseado na seleção das instâncias, realizados no Passo E, a fim de encontrar um novo h' que maximize a função. A hipótese atual h é atualizada com o valor de h'. O novo ponto conceito t' é estimado, maximizando a DD, sobre as instâncias selecionadas no passo E.

Dessa forma, o ponto conceito anterior t é substituído pelo novo ponto conceito t', e os dois passos anteriores descritos são repetidos até o algoritmo convergir. O teste de convergência é feito a cada iteração. Se o valor de DD, na iteração atual, é menor ou igual ao valor da iteração anterior, é considerado que o algoritmo convergiu.

Um ponto interessante dessa abordagem é o fato que EM-DD pode transformar os dados de múltiplas instâncias em dados de única instância, uma vez que somente uma instância é deixada em cada *bag*.

De acordo com os autores, o algoritmo EM-DD é, aproximadamente, dez vezes mais rápido que o DD tradicional, considerando-se a base de dados MUSK1. E, aproximadamente, cem vezes mais rápido que o DD tradicional, para a base de dados MUSK2. Além do bom

desempenho, é atestado pelos autores que EM-DD apresenta resultados melhores que o DD original.

2.3 Abordagens baseadas em modelos discriminativos

As abordagens baseadas em Modelos Discriminativos buscam a generalização dos métodos de aprendizado supervisionado tradicional, realizando adaptações para paradigma de AMI. Entre os principais métodos dessa abordagem estão: Citation-kNN (Wang e Zucker, 2000), miSVM (Andrews et al., 2003), MISVM (Andrews et al., 2003), MIKernel (Thomas et al., 2002), *etc.* Os algoritmos principais são detalhados neste capítulo.

2.3.1 Citação dos K Vizinhos mais Próximos - *Citation kNN*

O *Citation kNN* (Wang e Zucker, 2000) é um algoritmo baseado no algoritmo k Vizinhos mais Próximos (Cover e Hart, 1967), incluindo os conceitos de citação e referência usados na literatura científica, com objetivo de aumentar o desempenho do algoritmo. Nesse sentido, um *bag* é rotulado não somente de acordo com seus vizinhos, mas também com os *bags* que consideram esse *bag* um vizinho.

A ideia-chave por traz do *Citation kNN*, aplicado para Aprendizado de Múltiplas Instâncias, é como transformar a medida entre instâncias (como no kNN tradicional) em medidas entre *bags*. A seguir é apresentado como a similaridade (distância) entre *bag*s é feita no contexto de AMI, utilizando-se a distância Housdorff (Blumberg, 1920) e, também, a noção de Citação e Referência nos *bag*s, visando a sua classificação.

Distância Housdorff

O algoritmo kNN original (Cover e Hart, 1967) pressupõe que todas as instâncias são pontos no espaço de característica e os vizinhos mais próximos são calculados usando a distância euclidiana. No entanto, para usar o kNN em problemas de AMI, é necessário que a distância seja entre *bags* e não entre instâncias. Para resolver esse problema, Wang e Zucker (2000) propõe-se utilizar a distância de Housdorff (Blumberg, 1920) para medir a distância de dois *bags* de instâncias.

Por definição ((Edgar, 1995) apud (Wang e Zucker, 2000)), dois conjuntos de pontos A e B estão dentro de uma Distância de Housdorff, somente se cada ponto de A estiver dentro da distância d de ao menos um ponto de B e vice-versa. Assim, dados dois conjuntos de dados $A = \{a_1, ..., a_m\}$ e $B = \{b_1, ..., b_n\}$, a distância Hausdorff é definida como:

$$H(A, B) = max\{h(A, B), h(B, A)\} \tag{2.4}$$

na qual

$$h(A, B) = \max_{a \in A} \min_{b \in B} ||a - b|| \tag{2.5}$$

Em Wang e Zucker (2000), os autores relatam que a Distância Housdorff tradicional é muito sensível à *outliers*, comprometendo os resultados. A fim de evitar esta situação e aumentar a robusteza da medida de distância, Wang e Zucker (2000) propõem uma modificação no algoritmo original da Distância de Housdorff. Em vez de pegar a distância máxima, Wang e Zucker (2000) ordenam as distâncias e, posteriormente, escolhem o valor, assim a medida se torna:

$$h_k(A, B) = \underset{a \in A}{\mathrm{K}} \min_{b \in B} ||a - b|| \tag{2.6}$$

na qual K representada o k-ésimo valor ordenado. Em Wang e Zucker (2000), é indicado que o uso de $K = 1$, de forma que a Distância de Housdorff mínima seja usada e representa a menor distância entre duas instâncias, de *bag*s diferentes. A Figura 2.2 ilustra a Distância de Housdorff Mínima.

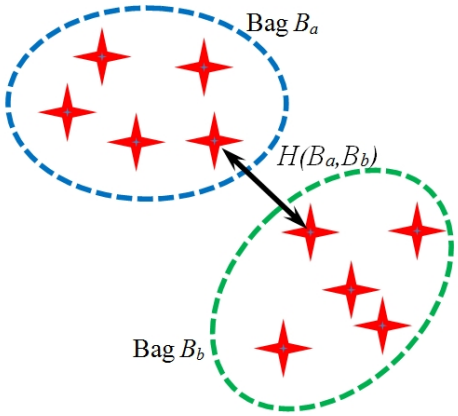


Figura 2.2: Uma ilustração da Distância de Housdorff Mínima.

Citações e referências

Após adotar o *kNN* tradicional para trabalhar com AMI, os autores concluíram que somente com o uso da Distância de Housdorff não foi suficiente para obter resultados promissores, então concluíram que outros melhoramentos nos algoritmos eram necessários. Isso é devido ao fato de alguns *bags* positivos conterem inúmeras instâncias negativas, de modo que é previsível que alguns *bags* negativos escolham esses *bags* como seus vizinhos mais próximos.

Para resolver esse inconveniente, os autores propuseram o uso da noção de citação da literatura (Garfield, 1979). Nesse caso, além de considerar os *bags* como vizinhos mais próximos (referências) de algum *bag* B_i, consideram-se, também, os *bags* que reconhecem B_i como seus

vizinhos (citação). Então, para a tarefa de classificação, a referência e os rótulos das citações são considerados. Por exemplo, vamos assumir que, para o R-próximas referências, positivas e negativas, representado por R_p e R_n, respectivamente e C-próximas citações, positivas e negativos, representado por C_p e C_n, respectivamente, um *bag* desconhecido, será classificado como positivo se $(R_p + C_p) > (R_n + C_n)$, e negativo caso contrário.

2.3.2 Máquinas de Vetores de Suporte para Aprendizado de Múltiplas Instâncias (mi-SVM and MI-SVM)

Um dos principais algoritmos de aprendizado supervisionado tradicional, que sofreu várias adaptações e utilização em AMI foi as Máquinas de Vetores de Suporte ou *Support Vector Machines* (SVM) (Boser et al., 1992). A principal questão para se considerar em adaptar as Máquinas de Vetores de Suporte para trabalhar com AMI é como definir a margem de separação entre as classes (Babenko, 2008).

Para os *bags* negativos, todas as instâncias são conhecidas como negativas, então a margem é definida com o caso tradicional. Porém, para os *bags* positivos, essa questão é mais complicada, pois esses *bags* contêm tanto instâncias positivas quanto negativas.

A adaptação de SVM para tratar problemas de AMI foi proposta por Andrews et al. (2003) em 2003. Em seu trabalho Andrews et al. (2003) propõe duas novas formulações para SVM, uma chamada MI-SVM: para tratar de AMI em nível de *bag*, então é procurada uma margem que maximize a distância entre *bags* e mi-SVM: para tratar AMI em nível de instâncias, que procura maximizar a distância entre instâncias.

No algoritmo mi-SVM, os rótulos das instâncias nos *bags* positivos são definidos inicialmente como todos sendo positivas, a partir daí o classificador é treinado repetidamente até que todos os *bags* tenha pelo menos uma instância rotulada como positiva (Xiao et al., 2010b). No algoritmo MI-SVM, a cada interação, somente uma instância de cada bag positivo é escolhida e, juntamente às instâncias dos bags negativos, são utilizadas no aprendizado. O objetivo, então, do MI-SVM é encontrar um hiperplano de margem máxima, na qual todas as instâncias, nos *bags* negativos, estejam posicionada de um lado do hiperplano, enquanto ao menos uma instância de cada *bag* positivo esteja localizada do outro lado.

A Figura 2.3 apresenta um exemplo simples que ilustra a diferença entre MI-SVM e mi-SVM. A maior diferença é que, na abordagem MI-SVM, as instâncias negativas no *bag* positivo são ignoradas e somente uma instância em cada *bag* positivo contribui para otimização do hiperplano. E, em mi-SVM, as instâncias negativas nos *bags* positivos, como também as múltiplas instâncias de um *bag* podem ser vetores de suporte.

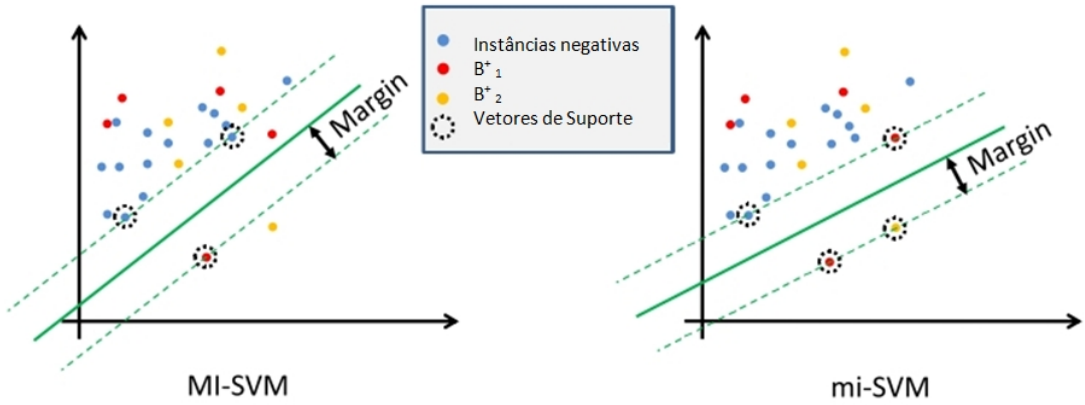


Figura 2.3: Comparação entre as soluções de MI-SVM e mi-SVM, para um espaço de característica 2D. Fonte: Extraída de Babenko (2008).

2.4 Abordagens baseadas em seleção de instâncias

Diferentemente das abordagens anteriores, os métodos baseados em seleção de instâncias têm por objetivo identificar instâncias que são mais ou menos relevantes, dependendo do ponto de vista do método e buscam tratar essas instâncias de forma específica. Este estudo minucioso das instâncias contribui tanto para o aumento da precisão dos métodos como, também, para a velocidade do processo de treinamento (Erdem e Erdem, 2011). Com essa abordagem, procura-se transformar os problemas de AMI em problemas de aprendizado supervisionado tradicional. Entre os diversos algoritmos nessa linha, se destacam principalmente: SMILE (Xiao et al., 2014), MILIS (Fu et al., 2011), MILDS (Erdem e Erdem, 2011), DD SVM Chen e Wang (2004b) E MILD-B Li e Yeung (2010), alguns desses são descritos a seguir.

2.4.1 SMILE

Os autores em Xiao et al. (2014) propõem uma nova abordagem chamada SMILE- *Similarity Based Multiple Instance Learning*. De acordo com os autores, o algoritmo SMILE utiliza a ambiguidade das instâncias, dos *bags* positivos, para aumentar a acurácia de AMI. Ao contrário de outros métodos que excluem as instâncias consideradas ambíguas, no *bag* positivo, SMILE trabalha com essa ambiguidade, atribuindo a essas instâncias dois pesos: o primeiro é associado à similaridade com as do *bag* positivo e a segunda com o *bag* negativo.

Inicialmente, os dados de treinamento são separados em três subgrupos: S_p^+ representado por todas as instâncias candidatas a serem positivas, S_a^+ que representa as demais instâncias dos *bags* positivos que não foram selecionadas como S_p^+ e, finalmente, S^- que são todas as instâncias dos *bags* negativos.

A seleção das instâncias candidatas a serem positivas S_p^+, é baseada no método de Similaridade de Conjunto. Então, dada uma instância x e um conjunto S, a Similaridade de

Conjunto é representada pela Equação 2.7.

$$R(x,S) = \frac{1}{|S|}\sum_{x_i \in S} e^{-||\phi(x)-\phi(x_i)||^2} \tag{2.7}$$

em que $|S|$ é o tamanho da amostra de S e $\phi(x_i)$ é a imagem da instância x_i no espaço de característica e $||\phi(x)-\phi(x_i)||^2 = K(x,x)+K(x_i,x_i)-2K(x_i,x_i)$, em que $K(..,..)$ é a função de *Kernel* de Mercer (Bernhard e J., 2001).

Assim, as instâncias mais similares às instâncias em outros *bags* positivos e, ao mesmo tempo, aquelas que posuírem a menor similaridade com os *bags* negativos serão definidas como instâncias positivas candidatas. Após a escolha das instâncias candidatas, os pesos são atribuídos às instâncias do grupo S_a^+. Para cada instância desse grupo, é calculada a similaridade em direção aos *bags* positivos e aos *bags* negativos.

Então, essas informações são incorporadas na fase de treinamento, visando construir um classificador robusto baseado em SVM. O algoritmo sempre atualiza as instâncias candidatas a serem positivas e os pesos das demais. Mais precisamente, com os trabalhos de Fu e A. (2009), os autores selecionam uma instância de cada *bag* positivo, como candidata principal e, dessa forma, conseguem transformar o problema de AMI em um problema de aprendizado supervisionado tradicional.

2.4.2 MILIS

Em Fu e A. (2009) e Fu et al. (2011), é proposto um novo algoritmo baseado em seleção de instâncias, o MILIS - *Multiple Instance Learning Instance Selection*. Esse algoritmo é baseado na seleção adaptativa de instâncias. Em seu trabalho, é feito uma otimização alternando os passos de seleção de instâncias e aprendizagem, em que, segundo os autores, garante a convergência do algoritmo. A seleção da instância é baseada na distribuição das instâncias negativas, que em Fu e A. (2009) é chamada de "*Instance Prunning*". Nessa etapa é utilizado o um estimador de densidade sobre todos os *bags* negativos, em que se sabe que todas as instâncias são negativas.

Os autores atestam que a grande vantagem em se modelar as instâncias negativas ocorre em virtude da grande quantidade dessas instâncias e como elas se sobressaem em relação ao conjunto de instâncias positivas.

Com base nessa estimação de distribuição das instâncias negativas, são escolhidas as instâncias mais negativas em cada *bag* negativo e é escolhida a instância menos negativa (ou mais positiva) em cada *bag* positivo. As instâncias escolhidas nos *bags* positivos são chamadas de instâncias Protótipos e somente uma é escolhida de cada *bag* positivo. Dessa forma os autores, como feito por Xiao et al. (2010b), transformam o problema de AMI em um problema de aprendizado supervisionado tradicional. Como classificador os autores utilizam uma SVM linear.

2.4.3 MILDS

Em Erdem e Erdem (2011) a intenção dos autores é, novamente, transformar o problema de AMI em um problema de aprendizado tradicional, usando a seleção de características. No artigo é proposto o método MILDS *Multiple Instance Learning via Dominant Set*. A inovação do método, segundo os autores, está na estratégia de seleção das instâncias mais representativas em cada *bag*, positivo e negativo, que é baseado no algoritmo de "*Pairwise Clustering*" (Pavan e Pelillo, 2007) via Conjunto Dominante. Esse algoritmo é utilizado para modelar a distribuição das instâncias negativas e selecionar um subconjunto de instâncias protótipos.

Segundo os autores, ao contrário de métodos como MILIS, para selecionar o conjunto representativo de instâncias, eles não usam função de densidade de probabilidade tampouco probabilidade condicional, mas sim modelam os dados negativos, agrupando os dados dos *bags* negativos, formando *clusters*. Porém, a estratégia para seleção das instâncias nos *bags* positivos deve ser diferente, devido a sua característica de possuir instâncias negativas incluídas nesses *bags* e isso afetaria encontrar *clusters* corretos. Para contornar esse problema, os autores utilizam uma medida de similaridade entre os *clusters* encontrados nos *bags* negativos e as instâncias de cada *bag* positivo. Assim, as instâncias com menor similaridade são escolhidas nos *bags* positivos. Entre as principais contribuições do trabalho, segundo os autores, se destaca a construção de um espaço de características de baixa dimensão.

2.5 Outras abordagens

Trabalhando na mesma linha de seleção de instâncias, mas também com redução de dimensionalidade, Zafra et al. (2010) propõem um algoritmo utilizando um filtro de seleção, aplicando o algoritmo ReliefF-MI, que é uma variação do ReliefF original (Kira e Rendell, 1992), desenvolvido especialmente para problemas de AMI. Em seu trabalho é apresentado uma série de experimentos em 17 tipos de algoritmos para AMI, utilizando de 10% a 70% (passos de 10%) das bases de dados mais usadas para AMI. A abordagem determina pesos para cada característica, de modo que as características são ranqueadas e, posteriormente, um limiar é utilizado para selecionar um percentual da base. Segundo os autores, os pesos representam a habilidade das características em distinguir as classes.

Propondo uma abordagem diferente, utilizando SVM, os autores em Krummenacher e Ong (2013) desenvolvem um novo método chamado eMIL *Elipsoidal Multiple Instance* baseado em margens largas, utilizando elipsoides. Segundo os autores, o método permite que os *bags* positivos cruzem a margem de classificação, uma vez que não são conhecidas quais instâncias nesses *bags* são realmente positivas. No método proposto, os *bags* são representados por elipsoides, que, de acordo com os autores, representam bem o problema, uma vez que é assumindo que um *bag* é uma variável aleatória com média e covariância finita. O método é baseado em

duas etapas: estimar o elipsoide para cada *bag*, e, posteriormente, a otimização da margem de classificação.

Uma abordagem interessante, focada mais no tempo de execução dos algoritmos de AMI, é proposta por Li et al. (2012b). Neste trabalho, os autores afirmam que trabalhar com a ambiguidade de AMI (nos *bags* positivos) é muito cara, computacionalmente, e, por essa razão, as aplicações de AMI em visão computacional são reduzidas. Com objetivo de lidar com essa situação, os autores propuseram um "Framework" chamado de BAMIL *Batch Mode Adaptative Multiple Instance Learning*, a fim de otimizar a velocidade dos métodos de AMI. Em vez de usar todos os *bags* de treinamento ao mesmo tempo, os autores dividem os *bags* de treinamento em diversos outros subconjuntos de *bags*, associando grupos de *bags*. Para cada grupo de *bags*, é encontrada uma fronteira de classificação, que é utilizada pelo próximo grupo de *bags*, adaptando a resposta do anterior e, assim, sucessivamente. De acordo com os autores, esse tipo de abordagem consegue acelerar os métodos de AMI, o que é especialmente desejado para aplicações em condições dinâmicas.

Uma tarefa difícil e crítica em AMI é, segundo os autores em Zhao et al. (2013), aprender, de forma efetiva, o mapeamento que converta uma tarefa de AMI em uma tarefa de aprendizado supervisionado tradicional. No artigo é apresentado o algoritmo OptMILES, que tem por objetivo principal aprender a transformação ótima para a medida de similaridade entre *bag* e instância, explorando, de forma aprofundada, a métrica de distância entre instâncias. De acordo com os autores, a distância Euclidiana adotada por muitos métodos na literatura para mensurar a distância entre as instâncias não é apropriada, assim eles propõe a distância de Mahalanobis em vez da distância Euclidiana. O método proposto é baseado no mapeamento de características originais de MILES (Chen et al., 2006a). Assim, é aprendida a métrica de distância, com a utilização da distância de Mahalanobis e combinada a métrica na formulação do SVM, usando iterações para aprender as amostras ótimas e o classificador, ao mesmo tempo.

2.6 Aplicações

Aprendizado de Múltiplas Instâncias foi introduzido, originalmente, por Dietterich et al. (1997) em 1997 no campo de predição de atividades de drogas e desde então vem sendo aplicado em diversos outros campos. Entre as principais aplicações para AMI, podem-se destacar: classificação e recuperação de imagem (Maron e Ratan, 1998; Li et al., 2012a; Wang et al., 2012; Yao et al., 2010; Liu et al., 2010; Zhang et al., 2005), predição de atividade de drogas (Maron, 1998; Huang et al., 2003), aplicações médicas (Das et al., 2012; Sun et al., 2012; Mahnot e Popescu, 2012), previsão de mercado de ações (Maron e Lozano-Perez, 1998), classificação de texto (Andrews et al., 2003; Keshtkar e Inkpen, 2011; Wei e Yu, 2009) *etc.* Nesta seção são descritas algumas dessas aplicações.

2.6.1 Recuperação de imagem

Uma das grandes aplicações de AMI é para a recuperação de imagens, seja pela internet ou em bancos de imagens *off-line*. Dentre as formas em que se pode realizar uma recuperação de imagem, o usuário apresenta uma série de imagens desejadas e o sistema é capaz de retornar imagens relacionadas com as que foram apresentadas inicialmente. Existem outras variações do tipo de entrada do usuário que, em vez de fornecer algumas imagens relacionadas, o usuário pode entrar textualmente com o tipo da imagem desejada ou seu significado semântico e o sistema, novamente, busca as imagens relacionadas com a entrada requisitada.

No contexto de recuperação de imagens, cada imagem representa um *bag*, as diversas regiões da imagem representam as instâncias, e os descritores (momentos, cores, histogramas, texturas, forma, *etc.*.) de cada região são as características de cada instância. A imagem, ou *bag*, é então classificada como positiva, se contém em pelo menos uma das suas regiões, ou instâncias, o objeto desejado ou que caracteriza aquela imagem com seu significado.

De acordo com os autores em Zhang et al. (2005), o objetivo da recuperação de imagem é encontrar uma região semanticamente correta com o interesse do usuário. Como cada imagem é composta por diversas regiões, e cada região pode ser descrita como uma instância, o problema de recuperação de imagem, pode ser facilmente transformado em um problema de AMI. Em Zhang et al. (2005), os autores utilizam SVM para resolver o problema de recuperação de imagem, utilizando AMI. A escolha desse classificador é justificada, pelos autores, devido ao fato de todas as instâncias positivas serem similares, enquanto as negativas, poderem assumir qualquer outra forma. Então, é correto afirmar, segundo os autores, que todas instâncias positivas pertencem a uma classe, enquanto as negativas podem ser consideradas como pontos fora da classe positiva.

Ainda no campo de recuperação de imagens, um seguimento que vem crescendo bastante nos últimos anos é a Busca semântica da imagem. No trabalho de Yao et al. (2010), é proposta a aplicação de AMI para busca semântica da imagem. Segundo os autores, em uma busca semântica de uma imagem, é altamente desejado desenvolver um método automático para extrair os vários significados que uma imagem pode conter. O método adota AMI para realizar o mapeamento das características de baixo nível em significados semânticos simples, que é relacionado com uma ou várias regiões da imagem, mas não com a imagem inteira. Então, quando existe pelo menos uma região da imagem relacionada com a semântica procurada, ela é rotulada como positiva. No método é utilizado o algoritmo Densidade Diversa como classificador.

Em Maron e Ratan (1998) é aplicado AMI para classificação de cenas naturais. No trabalho, o usuário rotula uma imagem como positiva, se esta contém em alguma região um conceito que descreve a cena daquela imagem (no trabalho foi utilizado: montanhas, campos e cachoeiras). Para o método, cada imagem é um *bag*, as várias regiões da imagem são as instâncias. Então, para uma base de imagens, contendo *bag*s positivos (contendo pelo menos

uma região conceito) e negativos, o método dos autores é capaz de aprender esses conceitos e recuperar a imagem com a cena natural desejada. Também, nesse trabalho é utilizado o algoritmo Densidade Diversa.

Existem, também, trabalhos que modelam a busca de recuperação de conteúdo em vídeos como um problema de AMI (Liu et al., 2010), em que o vídeo corresponde a um *bag* e cada *frame* corresponde a uma instância.

2.6.2 Aplicações médicas

Uma das aplicações mais crescentes para AMI é no campo da Medicina, em situações que a rotulação de cada evento separadamente, nem sempre é possível. Nessa seção são apresentados alguns exemplos de aplicação de AMI na área médica.

Os autores em Das et al. (2012) utilizam o paradigma de AMI para detectar sintomas de mal de Parkinson, em ambientes não controláveis, como a própria casa do indivíduo. Visando esse objetivo, os autores formularam o problema de detecção dos sintomas como um problema de Aprendizado de Múltiplas Instâncias. AMI foi escolhido para esse caso devido ao fato de o AMI não necessitar da rotulação precisa de cada instância, ou, para essa aplicação, a rotulação precisa, para cada pequeno intervalo de tempo, se ocorreu ou não o sintoma. Aplicando o conceito de AMI, é necessário somente identificar se em um intervalo maior de tempo ocorreu ou não sintoma em análise. Os autores monitoraram duas pessoas com a doença por quatro dias e cada paciente utilizando um acelerômetro triaxial para coletar as informações. Para o trabalho de Das et al. (2012), foi utilizado uma variante do algoritmo APR (Dietterich et al., 1997). Para essa aplicação, cada hora foi formulada como um *bag*, cada um dos sinais dos acelerômetros as instâncias e as características foram os sinais obtidos por meio das respostas dos acelerômetros. A Figura 2.4, ilustra essa situação.

De acordo com Mahnot e Popescu (2012), diversas aplicações em ciências médicas possuem conjunto de dados que possuem classificação em classe de forma ambígua. Em muito desses casos, a rotulação individual é muito complexa e cara, o que torna a rotulação de grupos muito mais interessante, dessa forma sendo propícia para a utilização de AMI. Neste trabalho os autores se preocupam com a detecção de doenças em pessoas idosas que vivem sozinhas em suas casas. Os autores propõem um novo algoritmo, chamado FUMIL, que é uma variação do algoritmo *Diverse Density* (Maron e Lozano-Perez, 1998), utilizando conceitos de lógica *Fuzzy*. Os dados foram coletados em uma casa especialmente desenvolvida, em que existem diversos sensores espalhados por toda parte da casa, coletando sinais de: Presença (no quarto, banheiro, cozinha, sala), padrão da utilização da cama, temperatura do fogão, *etc.* Formulando esse problema na ótica AMI, a atividade diária do paciente foi identificada como *bag* e cada hora como uma instância. As características de cada instância são os diversos sinais dos sensores.

Uma aplicação interessante para AMI foi proposto por Sun et al. (2012). Em seu trabalho

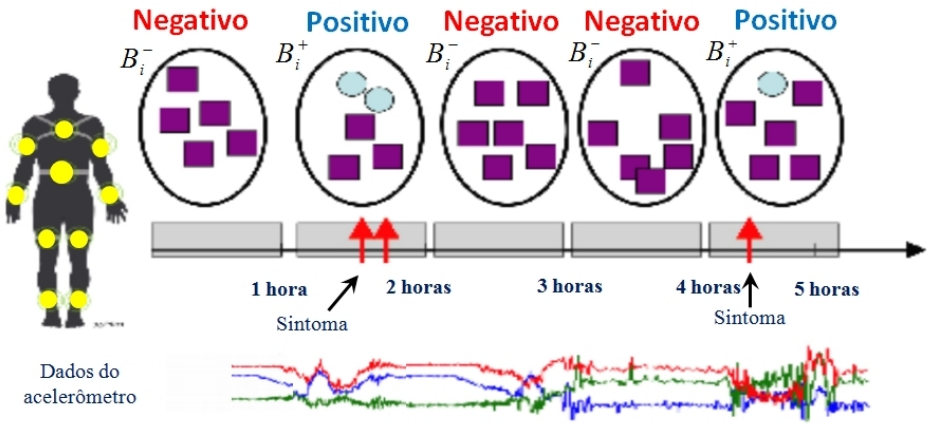


Figura 2.4: Ilustração do método para detecção de sintomas de Parkinson, formulado como AMI, Figura adaptada de Das et al. (2012).

é apresentado um novo algoritmo para automaticamente detectar Enfarte do Miocárdio (EM) a partir de ECG (Eletro Cardiograma), utilizando o paradigma de AMI. De acordo com os autores, o custo de rotular manualmente os batimentos do coração em pacientes é muito alto devido à enorme quantidade de batidas presentes em um ECG. Para contornar esse problema, os autores propõe um algoritmo, chamado de *Latent Topic MIL*, para, automaticamente, classificar um ECG. O método usa um filtro DCT (*Discrete Cossine Transform*), para eliminar ruídos do sinal, e, posteriormente, extração de características do sinal do ECG, pegando os principais pontos da forma de onda do ECG. Nesse trabalho, cada trecho ECG foi considerado como um *bag*, cada batida do coração uma instância e as propriedades da forma de onda para cada batida de coração as características das instâncias, como apresentado na Figura 2.5. Como classificador foi utilizado Otimizado de Enxame de Partículas para Pesos de Variáveis ou *Particle Swarm Optimizer for Variable Weighting* (PSOVW).

Em Xu et al. (2012) é apresentado uma aplicação para Aprendizado de Múltiplas Instâncias com o objetivo de avaliar tecidos cancerígenos, baseados em imagens histopatológicas. Em seu trabalho, os autores descrevem que é de grande importância rotular corretamente imagens histopatológicas que contenham regiões cancerosas ou não. É descrito, também, que a rotulação detalhada dessas imagens é, geralmente, um trabalho ambíguo e desafiador. Segundo os autores, realizar uma anotação geral de uma imagem histopatológica é uma atividade relativamente mais simples que uma análise detalhada em cada região da imagem. Portanto, o conceito de AMI pode ser aplicado para esse problema: cada imagem é considerada como um *bag* (positivo se tem a doença e negativo caso não tenha) e as diversas regiões no interno da imagem são suas instâncias, em cada imagem foram extraídas 64 regiões (instâncias). A Figura 2.6.2 apresenta um exemplo de *bag* e instâncias. O objetivo do trabalho consiste em realizar a classificação de uma nova imagem em dois níveis: em nível de imagem (imagem com câncer ou não) e, também, em nível de *pixels* (regiões do tecido com câncer ou não). O

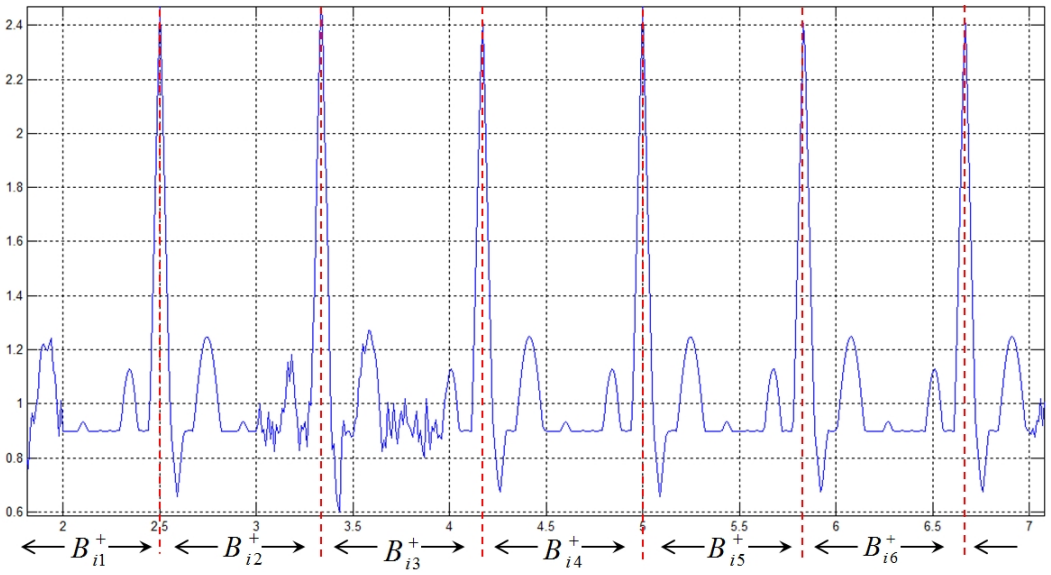
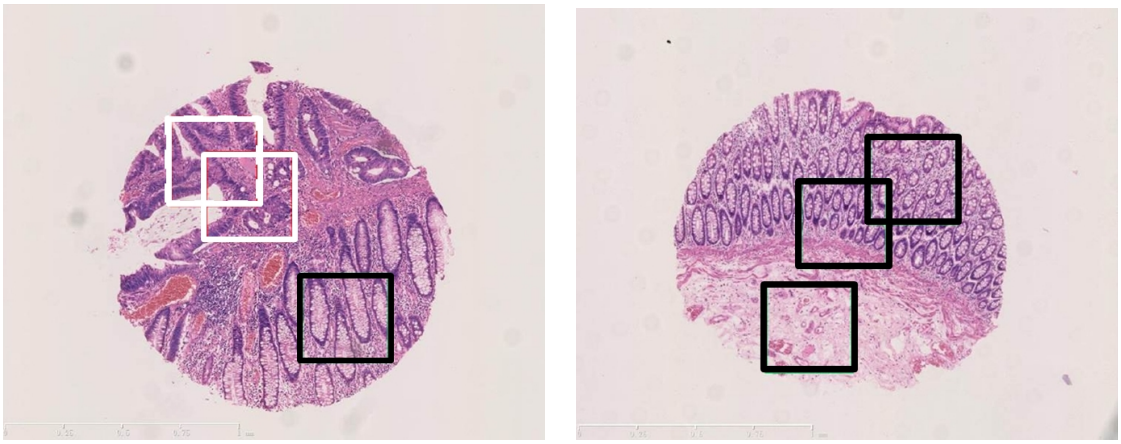


Figura 2.5: Ilustração da análise do ECG por AMI - cada trecho de ECG é considerado um *bag*.

algoritmo proposto se chama MCIL *Multiple Clustered Instance Learning*. De acordo com os autores, utilizando a abordagem de Aprendizado de Múltiplas instâncias, o método proposto foi capaz de superar diversos algoritmos tradicionais para as mesmas bases.



(a) (b)

Figura 2.6: Exemplos de *bags* e instâncias: a) *Bag* positivo (imagem com câncer) e b) *Bag* negativo (imagem sem câncer). Quadrados brancos indicam as instâncias positivas (tecidos com a doença) e quadrados pretos são instâncias negativas (tecidos sem a doença) - Figura extráida de Xu et al. (2012).

2.6.3 Previsão de mercado de ações

No mercado de ações, um dos grandes interesses dos investidores é ter a habilidade de poder prever se determinadas ações irão se valorizar ou se depreciados no tempo, a fim de decidir suas estratégias de negócios. Esse mercado é de extrema volatilidade e incontáveis variáveis determinam as mudanças de valores no mercado de ações. No artigo original do *Diverse Density* (Maron e Lozano-Perez, 1998), os autores descrevem duas possíveis aplicações para AMI. Uma das aplicações é relacionada ao problema de seleção de ações, em que o objetivo é selecionar ações que irão ter bom resultados.

O problema foi formulado da seguinte forma pelos autores: para cada mês, foram pegas 100 ações que tiveram os melhores resultados e estas foram rotuladas como *bags* positivos, assumindo que pelo menos uma dessas ações foram bem por razões fundamentais. Assim a medida que os *bags* positivos crescem, é mais provável conter uma instância realmente verdadeira. Os *bags* negativos foram criados com base nas cinco piores ações em cada mês. As instâncias das ações foram selecionadas características como momento, relação preço atual e preço justo, projeção futura, histórico, volatilidade, entre outras. De acordo com os autores, corretoras como Grantham, Mayo, Van Otterloo forneceram a base de dados, contendo as 600 maiores ações comercializada nos Estados Unidos, desde 1978 até 1995. Outra aplicação sugerida pelos autores se baseia em aprender a descrição de uma pessoa, dentre uma série de imagens, que são rotuladas como positivas se elas contém a pessoa e negativa caso não contenha.

2.6.4 Caracterização de texto

Uma aplicação com grande potencial para AMI é a categorização de texto, na qual o objetivo é determinar rótulos semânticos para documentos. Para esse aplicação, um documento é representado como um *bag* e instâncias são obtidos pela divisão do texto em pequenos trechos. As características são determinadas como: ocorrência de palavras, frequências dentre outras.

Os autores em Andrews et al. (2003) geraram uma base de dados para caracterizar texto, baseado no paradigma de AMI. A base foi construída a partir da base de dados pública OHSUMED (TREC9). A base OHSHUMED é uma coleção de mais de 300 mil referências da MEDLINE (*on line medical information database*), que consiste em títulos e/ou resumos de mais de 270 periódicos da área de medicina. Os autores dividiram os documentos em passagens, com janelas de no máximo 50 palavras cada. Para essa tarefa, foram utilizados os algoritmos mi-SVM e MI-SVM, aplicando-se um *kernel* polinomial.

2.7 Conclusões do capítulo

Neste capítulo foi realizada uma revisão bibliográfica sobre as principais metodologias para Aprendizado de Múltiplas Instâncias encontrados na literatura. As metologias apresentadas

foram divididas em três principais categorias: abordagens baseadas em modelos generativos, aprendizado discriminativo e seleção de instâncias. Foram descritos também outros metodologias encontrados na literatura.

Na segunda parte do capítulo, foram apresentadas algumas das principais aplicações para AMI, com um maior destaque para aplicações envolvendo Recuperação de Imagens e Aplicações Médicas.

Capítulo 3

Seleção de Instâncias Positivas, via Estimador de Densidade por *Kernel*

As abordagens para Aprendizado de Múltiplas Instâncias descritas no capítulo 2 apresentam uma forte tendência para os algoritmos baseadas em Seleção de Instâncias, mais utilizados nos últimos anos. Analisando-se seus resultados, observa-se que estes estão entre as abordagens que apresentam os melhores resultados da literatura, apontando para uma direção nessa área. Alguns desses algoritmos (MILIS (Fu et al., 2011), MILDS (Erdem e Erdem, 2011) *etc*) buscam encontrar um padrão ou estrutura na classe negativa com intuito de encontrar um modelo que possa ser aplicado aos *bags* positivos e, assim, escolher a instância menos negativa, em cada *bag*. Utilizar a classe negativa, por si só, para criar um modelo, pode ser um caminho perigoso, especialmente para Aprendizado de Múltiplas Instâncias, em que os *bags* negativos podem representar qualquer situação, não necessariamente correlacionadas entre si. Então, uma dada metodologia pode-se sair bem em uma base de dados em estudo, por existir uma coerência entre as instâncias dos *bags* negativos, mas essa mesma regra pode não ser aplicada às demais base de dados. Esse mesmo pensamento é válido para os *bags* positivos, em que as instâncias negativas de cada *bag* podem não ter nenhuma relação entre si. A Figura 3 apresenta um exemplo de três *bags* positivos, em que as instâncias negativas não possuem relação aparente entre si. As três figuras, embora possuam um "cachorro" em comum em cada uma, não apresentam nenhuma semelhança entre as regiões contidas no *background* das três imagens.

Nesse capítulo, dois novos algoritmos para tratar o problema de Aprendizado de Múltiplas Instâncias são propostos. A metodologia adotada para o desenvolvimento dos algoritmos se baseia na abordagem de Seleção das Instâncias dos *bags* positivos, que consiste em encontrar aquelas instâncias que, realmente, são mais positivas, *i.e.* aquelas que mais se diferenciam estatisticamente das instâncias negativas e são capazes de representar melhor cada *bag* positivo,

(B_1^+) (B_2^+) (B_3^+)

Figura 3.1: Três *bags* positivos ("cachorro"), com instâncias negativas (regiões que não contém o cachorro) não correlatas.

buscando, dessa forma, realizar um melhor treinamento, minimizando o ruído provocado pelas instâncias negativas presentes nesses *bags*. Porém, a Seleção de Instância aplicada nas metodologias propostas contemplam análises tanto dos *bags* positivos quanto dos *bags* negativos, tornando a metodologia mais generalista e robusta. A abordagem proposta, além de contar com elementos bem característicos de Seleção de Instâncias, comprovadamente eficiente de acordo com a literatura, combina alguns conceitos de algoritmos da abordagem Generativa (Densidade Diversa (Maron e Lozano-Perez, 1998)), para o mapeamento das probabilidades iniciais de cada instância.

O primeiro algoritmo proposto, denominado MILKDE, utiliza Estimador de Densidade por *Kernel* para encontrar a instância mais positiva entre todos os *bags* positivos, e em seguida emprega duas etapas: Classificação e Repescagem para compor a classe positiva, que juntamente com todas as instâncias dos *bags* negativos irão compor o treinamento do classificador. O Segundo algoritmo, denominado MILKDE2, utiliza os mesmos passos do primeiro algoritmo, porém sem a etapa da Repescagem.

A Seção 3.1 descreve a abordagem proposta e, em cada subseção subsequente, são explicados os passos que compõe a metodologia.

3.1 Algoritmos propostos

Nesta seção são apresentados os detalhes das duas metodologias propostas a fim de lidar com os problemas de Aprendizado de Múltiplas Instâncias.

Como enunciado inicialmente na Introdução, uma das principais premissas para Aprendizado de Múltiplas Instancias é que "Em um *bag* positivo, existe no mínimo, uma instância realmente positiva". Fundamentado nessa afirmativa, foram proposto as duas metodologias a serem apresentados neste capítulo, a fim de identificar a instância mais positiva em cada *bag* positivo.

As abordagens propostas identificam, inicialmente, a instância mais positiva dentre todas as instâncias dos *bags* positivos e negativos. Essa etapa é chamada de passo 1. Posterior-

mente é identificada uma instância que seja a mais representativa de cada *bag* positivo (passo 2), em relação à instância escolhida no passo 1, essa etapa é chamada de Qualificação. A proposta, ainda, utiliza uma terceira etapa, em que instâncias que não foram selecionadas no passo 2 concorrem entre si para compor o grupo das instâncias positivas. Essa etapa (passo 3) é chamada de Repescagem. O conjunto resultante das instâncias positivas selecionadas na Qualificação e Repescagem, em adição com todas as instâncias negativas, constituem os dados para a construção do classificador. Essa metodologia descreve o primeiro algoritmo proposto e foi nomeado de MILKDE, ela contempla todas as etapas aqui descritas. A segunda metodologia, chamada MILKDE2, contempla todos os passos até a etapa de Qualificação, não existindo a etapa de Repescagem.

A identificação da instância mais positiva dentre todo conjunto de instâncias emprega o cálculo da probabilidade de cada instância $\mathbf{x}$, por meio do Estimador de Densidade por *Kernel* (KDE).

Um esquema simplificado da metodologia MILKDE pode ser observado na Figura 3.2. Nessa Figura, os triângulos representam as instâncias dos *bags* positivos e os losangos representam as instâncias dos *bags* negativos. Ainda, os triângulos preenchidos são as instâncias mais positivas de cada *bag* positivo, os triângulos sem preenchimento representam as instâncias candidatas à repescagem e, finalmente, os triângulos pontilhados são aquelas instâncias positivas escolhidas durante a etapa de repescagem.

No esquema ilustrado, observamos que, na etapa de Qualificação, é escolhida somente uma instância de cada *bag* positivo e, durante a Repescagem, as instâncias não escolhidas são ranqueadas de acordo com sua proximidade com a instância mais positiva e, posteriormente, somente um percentual dessas instâncias é escolhido. A Figura 3.2 mostra, também, que o conjunto de instâncias positivas, entregue ao treinamento do classificador, é composto pelas instâncias selecionadas na Qualificação mais aquelas selecionadas na Repescagem. Os conjuntos de instâncias negativas não sofrem qualquer modificação, sendo utilizado na sua íntegra para compor o treinamento do classificador.

Nas próximas seções são detalhados cada etapa das metodologias propostas.

3.1.1 Cálculo de probabilidade via Estimador de Densidade por *Kernel*

Com o propósito estudar a eficiência da metodologia proposta, um experimento envolvendo dados sintéticos foi realizado. O conjunto de dados sintéticos foi gerado utilizando o mesmo procedimento descrito no artigo de Maron e Lozano-Perez (1998). Para a criação dessa base sintética, foram criados 5 *bags* positivos e 5 *bags* negativos. Cada *bag* contém 50 instâncias uniformemente distribuídos em um *range* de $[0, 100]$ para cada variável. A diferença entre os *bags* positivos e negativos é que os *bags* positivos possuem no mínimo uma instância dentro de um quadrado central $(50, 50)$, como apresentado na Figura 3.3.

A probabilidade condicional de uma instância $\mathbf{x}$ dado os *bags* positivos e negativos é des-

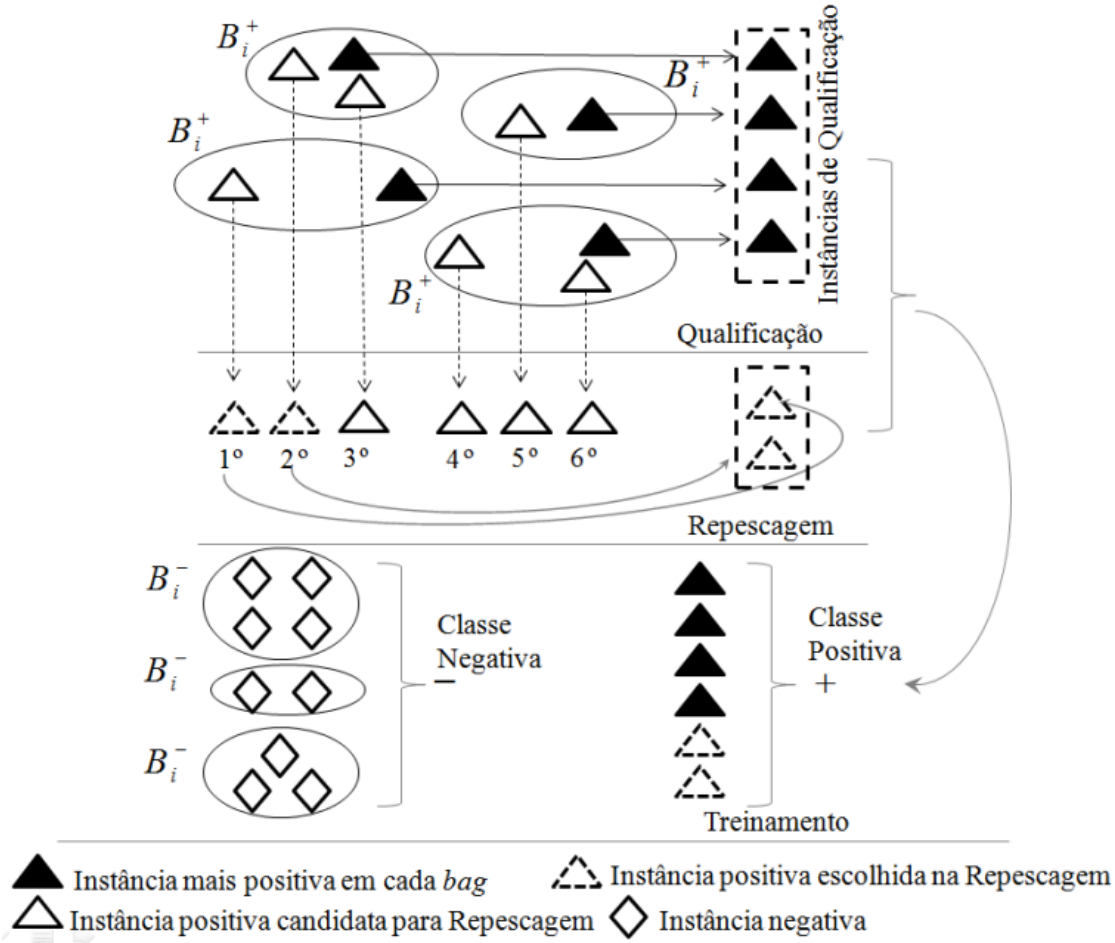


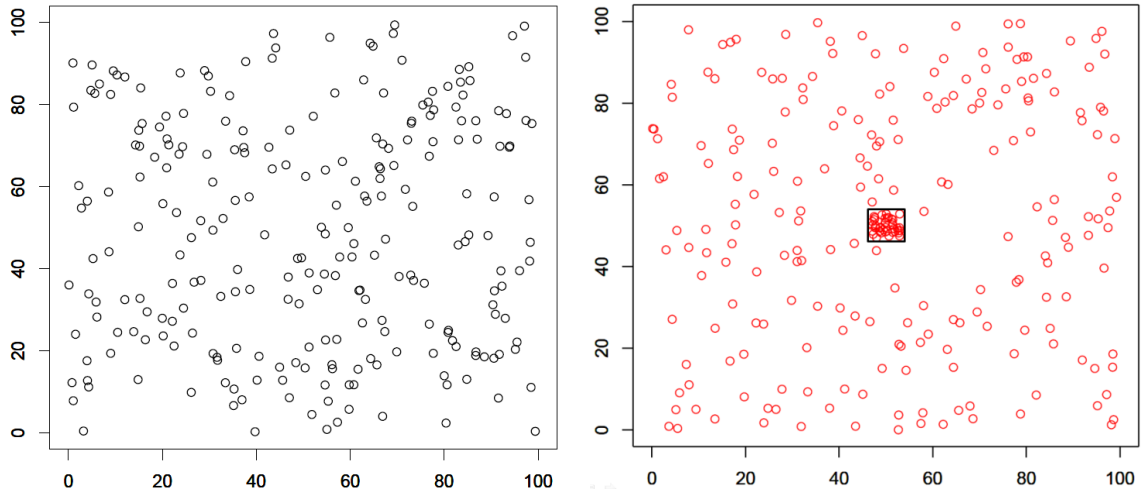
Figura 3.2: Metodologia proposta.

crita pela probabilidade $P(\mathbf{x}|B_1^+, B_2^+, ..., B_n^+, B_1^-, B_2^-, ..., B_m^-)$ onde n e m representam, respectivamente, o números de *bags* positivos e negativos. Como é assumido que os *bags* são independente em relação aos outros, essa probabilidade pode ser fatorada em $P(\mathbf{x}|B_1^+, B_2^+, ..., B_n^+)$ e $P(\mathbf{x}|B_1^-, B_2^-, ..., B_m^-)$, e finalmente expressa pela Equação 3.1, como feito pelo algoritmo Densidade Diversa (Maron e Lozano-Perez, 1998).

$$P(\mathbf{x}^+) = \prod_{i=1}^{n} P\left(\mathbf{x}|B_i^+\right) \tag{3.1a}$$

$$P(\mathbf{x}^-) = \prod_{i=1}^{m} P\left(\mathbf{x}|B_i^-\right) \tag{3.1b}$$

Em uma situação na qual o conjunto de amostras atende aos critérios de normalidade, sua distribuição pode ser estimada por meio de parâmetros como desvio padrão, correlação e



(Instâncias dos *bags* negativos) (Instâncias dos *bags* positivos)

Figura 3.3: Dados sintéticos, uniformemente, distribuídos dentro dos limites 0 e 100 em ambas as direções. Amostras positivas são aquelas que estão contidas dentro do quadrado central, centrado em $(50, 50)$ e lados de 5 ± 3, nas duas dimensões.

média. Porém, nem sempre o conjunto de amostras se apresentará de forma bem controlada ou comportada o suficiente para ser modelado por modelos paramétricos, como a distribuição Normal. Então, mesmo em situações como essas é possível estimar a densidades dos pontos por meio de um modelo não paramétrico. Um desses modelos é o Estimador de Densidade por *Kernel* (KDE) (Parzen, 1962), que será utilizada na metodologia proposta. A estimativa utilizando KDE é computada por meio de superposição de funções de densidades centradas em cada um dos pontos do conjunto de amostras.

Então, a probabilidade de cada instância $\mathbf{x}$ dados os *bags* positivos e negativos pode ser estimada por meio da utilização do Estimador de Densidade por *Kernel* (KDE). Para a metodologia proposta, foi utilizado um KDE Gaussiano, como apresentado na Equação 3.2, onde $K(\cdot)$ é a função *Kernel* Gaussiano, $\mathbf{x_j}$ representa uma instância em particular no espaço de amostra, $\mathbf{x}$ o ponto (instância) de interesse da estimação da verossimilhança, e b_i é o tamanho da amostra do *bag* B_i.

$$\widehat{f}_h(\mathbf{x}) = \frac{1}{b_i} \sum_{j=1}^{b_i} K_h(\mathbf{x} - \mathbf{x}_j) = \frac{1}{b_i h} \sum_{j=1}^{b_i} K(\frac{\mathbf{x} - \mathbf{x}_j}{h}) \tag{3.2}$$

A distribuição da probabilidade é uma função *Kernel* com largura de banda h, que é um parâmetro de suavização. Dessa forma, a identificação do parâmetro h ótimo passa a ser um problema de otimização. Para encontrar esse h ótimo, foi utilizado o algoritmo de Evolução Diferencial (Price et al., 2005).

Após empregar o Estimador de Densidade por *Kernel* (KDE) sobre os dados gerados

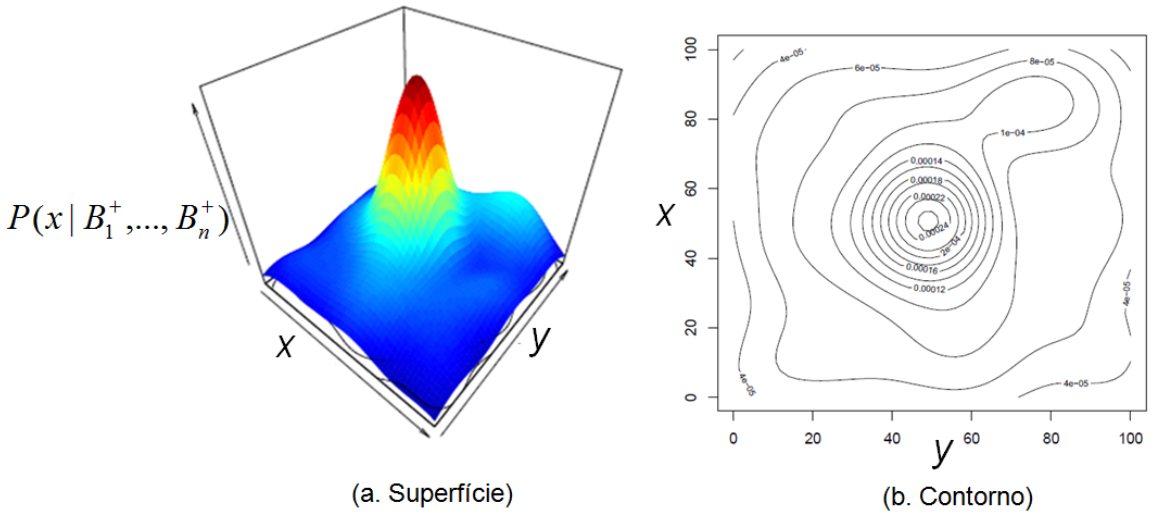


Figura 3.4: Probabilidade de uma amostra, dado todos os *bags* positivos: $P(\mathbf{x}_i|B_1^+, B_2^+, \cdots, B_{N^+}^+)$.

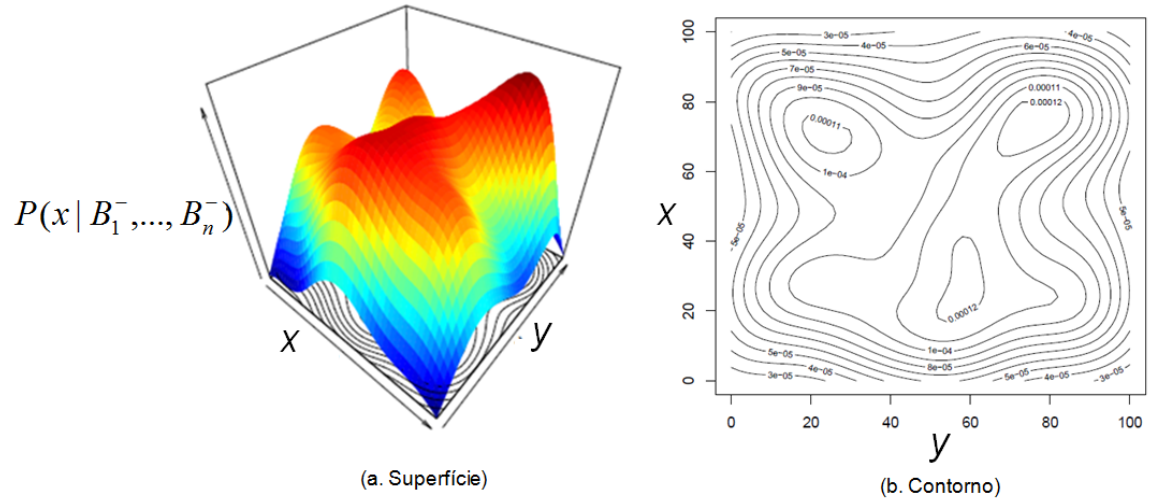


Figura 3.5: Probabilidade de uma amostra, dado todos os *bags* negativos: $P(\mathbf{x}_i|B_1^-, B_2^-, \cdots, B_{N^-}^-)$.

sinteticamente, obtemos uma superfície resultante e seu respectivo gráfico de contorno, representando a probabilidade das instâncias positivas, dados todas os *bags* positivos. Esses gráficos são apresentados na Figura 3.4 a e 3.4 b, respectivamente. A superfície correspondente e gráfico de contorno com relação às instâncias negativas são apresentados nas Figuras 3.5 a e 3.5 b, respectivamente. A suavidade da superfície depende, claramente, do raio da função *Kernel* implementado no KDE.

Como esperado, pode ser observada uma concentração das altas probabilidades na área

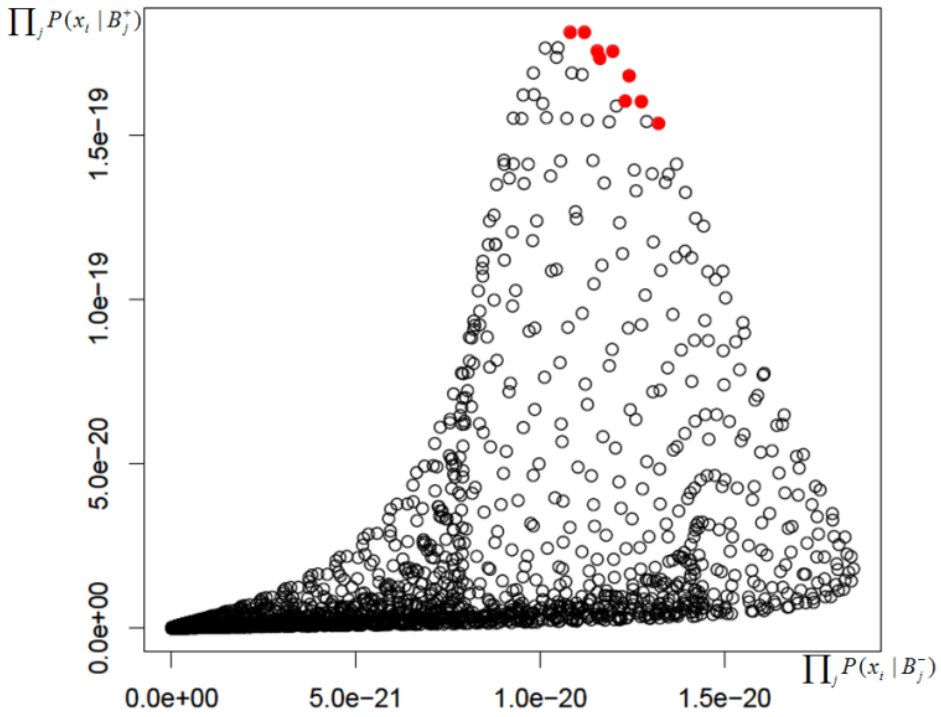


Figura 3.6: Projeção de todos os dados de entrada no espaço de probabilidades $\prod_j P(\mathbf{x}_i|B_j^-) \times \prod_j P(\mathbf{x}_i|B_j^+)$. Pontos em vermelhos são aqueles dentro do quadrado central, como mostrado na Figura 3.3.

central da Figura 3.4, que representa a probabilidade de todas as instâncias positivas (dados todos os *bags* positivos). E já para a Figura 3.5 observa-se uma dispersão das altas probabilidades negativas nos cantos do gráfico.

A Figura 3.6 nos mostra a projeção de todas instâncias positivas e negativas sobre o espaço de probabilidades $\prod_{i=1}^{n} P(\mathbf{x}|B_i^+)$ *vs* $\prod_{j=1}^{m} P(\mathbf{x}|B_j^-)$. Com essa projeção pode-se constatar que as probabilidades, associadas às instâncias positivas localizadas dentro do quadrado central da Figura 3.3, possuem grande proximidade entre si, ficando relativamente perto uma das outras no espaço de probabilidades.

Com base nos estudos realizados com os dados sintéticos (gerados utilizando o mesmo procedimento descrito no artigo de Maron e Lozano-Perez (1998)), o uso do KDE se mostrou uma técnica eficiente para a identificação da instância mais positiva. A instância mais positiva, considerando-se todas as instâncias nos *bags* positivos, será aquela que apresentar o maior valor de diferença entre as probabilidades $P(\mathbf{x}^+)$ e $P(\mathbf{x}^-)$ e será definida a partir desse momento no texto como Instância Mais Positiva (IMP).

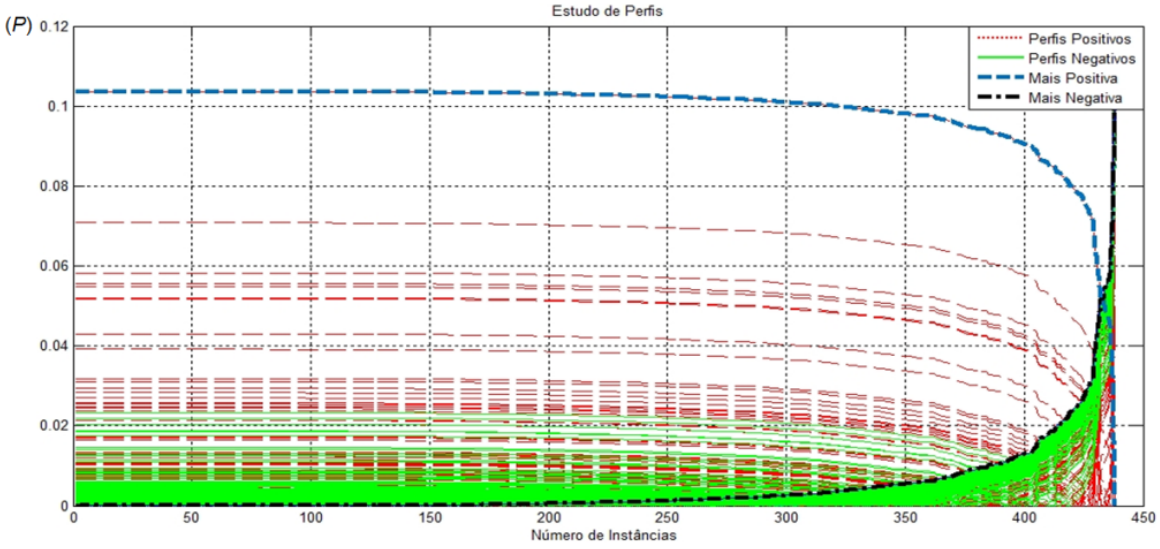


Figura 3.7: Perfil de diferença de probabilidades para a base MUKS1.

3.1.2 Passo 1: Identificação da Instância Mais Positiva

Para identificação da Instância Mais Positiva, inicialmente é calculada para cada instância dos *bags* positivos sua probabilidade de pertencer aos *bags* positivos e aos *bags* negativos ($\prod_{i=1}^{n} P(\mathbf{x}|B_i^+)$) e ($\prod_{j=1}^{m} P(\mathbf{x}|B_j^-)$), como apresentado na Seção anterior.

De posse das probabilidades $P(\mathbf{x}^+)$ e $P(\mathbf{x}^-)$, é calculada a distância, ou diferença, entre todas as probabilidades para todas as instâncias positivas. Assim, é construído uma Matriz de Afinidade, que representa a diferença de probabilidades, entre todas essas instâncias. Se considerarmos cada linha dessa matriz, teremos um "Perfil de diferença de probabilidades" de cada instância em relação a todas as outras. O intuito é encontrar aquela instância que possui a maior probabilidade de pertencer a um *bag* positivo e, ao mesmo tempo, a menor probabilidade de pertencer a um *bag* negativo. Assim, em posse da matriz de afinidade, os dois perfis de distância, mais distante entre si, serão identificados entre os *bags*. Dessa forma, a Instância Mais Positiva será aquela com maior probabilidade de pertencer aos *bags* positivos e menor probabilidade de pertencer aos *bags* negativos.

Uma vez que os dois perfis são encontrados (o mais positivo e o mais negativo), os demais perfis serão ordenados de acordo com a Instância Mais Positiva e utilizamos seus índices de ordenação para ordenar todos os demais perfis dos *bags* positivos e negativos. Esse processo resultará em perfis de distância como o apresentado na Figura 3.7.

3.1.3 Passo 2: Seleção das instâncias positivas - Qualificação

Na etapa de Qualificação, é realizada uma análise em cada bag positivo, com intuito de eleger uma única instância para ser a representante daquele bag. Espera-se que essa instância seja a

mais positiva daquele *bag*.

Após identificar a Instância Mais Positiva IMP, como apresentado na Seção anterior, é calculada a distância Euclidiana entre todas as instâncias em um dado *bag* positivo, em relação à Instância Mais Positiva. As instâncias são então ordenadas de acordo com sua proximidade a IMP.

Para cada *bag* positivo, a instância que apresentar a menor distância até a IMP, consequentemente a mais positiva daquele *bag*, será a eleita naquele *bag*. Essa etapa é chamada de Qualificação.

A metodologia MILKDE2 utiliza, somente, os passos que foram descritos até esse ponto. Para esse algoritmo, a classe positiva é formada por somente uma instância positiva que melhor representa cada *bag*. Ou seja, o número de instâncias positivas será a quantidade de *bags* positivos que a base de dados possuírem. Essas instâncias positivas escolhidas irão compor o treinamento do classificador juntamente a todas as instâncias negativas. É justo dizer que sempre que utilizarmos esse algoritmo, iremos ter o problema de classes desbalanceadas, com uma representação bem maior da classe negativa. Porém, como os resultados do capítulo 4 apresentam, mesmo com essa característica *i.e.*, classes bem desbalanceadas, a metodologia proposta foi capaz de obter resultados competitivos com as demais abordagens da literatura.

3.1.4 Passo 3: Repescagem

Avaliando os problemas de Aprendizado de Múltiplas Instâncias, é razoável assumir que em um *bag* positivo exista mais de uma instância positiva, sendo assim, um terceiro passo é desenvolvido: Repescagem.

Para execução do processo de Repescagem, todas as instâncias não selecionadas na Qualificação são ordenadas de acordo com sua distância em relação à Instância Mais Positiva. Um aspecto importante dessa etapa é selecionar o número correto de instâncias para Repescagem. Para determinar o número de instâncias a ser selecionado, um percentual de instâncias é testado, variando esse percentual entre 5% até 50% em passos de 5%. Para cada passo, é avaliada a precisão, por meio de validação cruzada utilizando $10 - fold$. Como critério de seleção, é utilizado o percentual de instâncias escolhidas que adicionadas às da Qualificação fornece a maior acurácia para a metodologia.

Sumarizando a metodoligia proposta: inicialmente, é procurada a Instância Mais Positiva dentre todas as instâncias positivas, baseado na estimação de suas probabilidades. Um segundo passo, chamado de Qualificação, identifica a instância mais positiva em cada *bag* positivo, com base na sua proximidade à IMP. Por último, o terceiro passo chamado de Repescagem seleciona um grupo de instâncias, não escolhidas na Qualificação.

O conjunto de instâncias positivas selecionadas na etapa de Qualificação e aquelas selecionadas na Repescagem formam, em conjunto com todas as instâncias negativas, o conjunto de dados utilizados para o treinamento do classificador.

Algoritmo 1 Proposta

procedure MILKDE(B_i^+,B_l^-)
  $n \leftarrow$ *total de* $Bags^+$
  $b_i \leftarrow$ *total de instancias em cada* Bag^+
  $m \leftarrow$ *total de instancias para repescagem*

  $1^{a}Etapa : Qualificacao\ de\ Instancias$
  para $i \leftarrow 1$ **até** Todas instancias de treinamento **faça**
    $P(\mathbf{x}_i^+) = \prod_{j=1}^{n} P\left(\mathbf{x}_i | B_j^+\right)$
    $P(\mathbf{x}_i^-) = \prod_{j=1}^{m} P\left(\mathbf{x}_i | B_j^-\right)$
  fim para
  $MA \leftarrow MatrizAfinidade(P(\mathbf{x}_i^+), P(\mathbf{x}_i^-))$
  $IMP \leftarrow InstanciaMaisPositiva(MA)$
  para $i \leftarrow 1$ **até** n **faça**
    para $j \leftarrow 1$ **até** b_i **faça**
      $D(j) = Dist(B_{ij}^+, IMP)$
    fim para
    $InstQualificada(i) = B_{ij} \rightarrowtail minimo(D(j))$
  fim para

  $2^{a}Etapa : Repescagem\ de\ Instancias$
  $InstCandidataRepescagem \leftarrow B_i^+ - InstQualificada$;
  $InstCandidataRepescagemOrdenada \leftarrow Sort(InstCandidataRepescagem)$;
  $InstRepescadas \leftarrow InstCandidataRepescagemOrdenada(1..m)$;

  $3^{a}Etapa : Classificador$
  $ClassePositiva \leftarrow InstQualificada + InstRepescadas$;
  $ClasseNegativa \leftarrow B_l^-$;
  $Test \leftarrow LOGITBOOST(ClassePositiva, ClasseNegativa)$;
fim procedure

Após realizar testes com 03 classificadores: Redes Neurais Artificiais (Haykin, 1998), Máquina de Vetores de Suporte (Boser et al., 1992) e LogitBoost (Friedman et al., 2000), o algoritmo LogitBoost que se adaptou melhor ao tipo de problema tratado neste trabalho, obtendo melhores resultados. Além dos bons resultados entre os demais, LogitBoost foi escolhido devido aos seus resultados com variadas classes de problemas na literatura e utilizar poucos parâmetros de configuração. Esse algoritmo, da família de "*boosting*", será brevemente abordado a seguir.

3.1.5 LogitBoost

Os algoritmos *Boosting* possuem grande sucesso de aplicação em Aprendizado de Máquina. Originalmente, os algoritmos baseados em *Boosting* foram desenvolvidos com o propósito de

combinar classificadores "fracos", com intuito de melhorar seus resultados. O primeiro método de *Boosting* foi proposto, originalmente, por Freund (1995), chamado de AdaBoost, e, desde então, uma série de outras variações tem sido propostas.

LogitBoost é pertence à família dos algoritmos *Boosting*, desenvolvido por Friedman et al. (2000). O método é baseado na regressão logística e construção de classificadores "fracos". A ideia básica do LogitBoost é empregar a abordagem *Boosting* enquanto constrói um modelo **logit**.

De acordo com A. Dehzangi et al. (2009), LogitBoost realiza uma regressão logística afim de gerar um modelo que maximize as probabilidades. Resumidamente, pode-se dizer que o algoritmo AdaBoost otimiza a perda exponencial enquanto o LogitBoost minimiza a perda logística. Para todos os experimentos neste trabalho, foram utilizados o LogitBoost como classificador. Um pseudocódigo do LogitBoost é apresentado no Algoritmo 2.

Algoritmo 2 LogitBoost

1: Inicialize os pesos $w_i = 1/N, i = 1, 2, ..., N, F(x) = 0$ e estimativas de probabilidades $p(x_i) = 1/2$
2: **para** m $\leftarrow$ 1 **até** M **faça**
3: (a)Calcule a resposta z_i e os pesos w_i
4: $z_i = \frac{y_i^* - p(x_i)}{p(x_i)(1-p(x_i))}$
5: $w_i = p(x_i)(1 - p(x_i))$
6: (b) Ajuste a função do classificador "*fraco*" $f_m(x)$ fazendo a regressão de z_i em x_i por mínimos quadrados ponderando usando os pesos w_i
7: (c) Atualize o classificador final $F(x) \leftarrow F(x) + \frac{1}{2} f_m(x)$ e $p(x) \leftarrow \frac{(e^{F(x)})}{e^{F(x)} + e^{-F(x)}}$
8: **fim para**
9: Saia com o classificador final $sign[F(x)] = sign[\sum_{m=1}^{M} f_m(X)]$

3.2 Conclusões do capítulo

Neste capítulo foram apresentados dois novos algoritmos, baseados em seleção de instâncias, a fim de lidar com o problema de Aprendizado de Múltiplas Instâncias. Ambos os algoritmos são fundamentados no cálculo das probabilidades, empregando o Estimador de Densidade por Kernel (KDE), de uma instância pertencer a um *bag* positivo ou negativo. As abordagens baseadas em seleção de instâncias têm por objetivo selecionar as instâncias mais positivas dos *bag* positivos, eliminando as possíveis instâncias negativas contidas nestes *bag*, dessa forma eliminando os ruídos em cada *bag*.

Estudos sobre uma base gerada sinteticamente demonstraram a escolha do KDE como uma boa alternativa para o cálculo das probabilidades. As etapas principais das metodologias podem ser resumidas da seguinte forma: a) Seleção da Instância Mais Positiva, b) Qualificação e c) Repescagem. Para o segundo algoritmo proposto (MILKDE2), a etapa de Repescagem não é utilizada.

Capítulo 4

Experimentos, Resultados e Discussões

Neste capítulo, são apresentados e discutidos os experimentos realizados para validar as metodologias propostas para Aprendizado de Múltiplas Instâncias, apresentadas no capítulo anterior. A Seção 4.1 descreve as bases utilizadas para validar e comparar a metodologia proposta com aquelas da literatura. Na Seção 4.2, são abordadas as análises estatísticas que serão empregadas para avaliar os resultados obtidos em relação aos algoritmos da literatura. A Seção 4.3 apresenta os resultados encontrados para as metodologias propostos e algumas discussões sobre esses resultados. Por fim, na Seção 4.4 são realizadas discussões finais sobre os resultados alcançados.

4.1 Base de dados

Nesta seção são descritas as principais bases[1] de dados utilizadas para validar os algoritmos propostos para Aprendizado de Múltiplas Instâncias. As primeiras bases apresentadas são as bases mais utilizadas na literatura para comparar as diversas metodologias propostas. As demais bases apresentadas são bases multiclasses adaptadas para o problema de Aprendizado de Múltiplas Instâncias. Estas, também, têm sido adotadas em alguns trabalhos para testes comparativos. Um sumário com as principais informações dessas bases estão apresentados nas Tabelas 4.1 e 4.2.

4.1.1 Base de dados MUSK, Elefante, Tigre e Raposa

As primeiras bases de dados utilizadas para AMI foram a MUSK1 e MUSK2. Essas duas bases de dados foram fornecidas por Dietterich et al. (1997), e as duas estão disponíveis na UCI *Machine Learning Repository* (Lichman, 2013).

[1] É de extrema importância ressaltar que os rótulos das instâncias contidas nos *bags* positivos são totalmente desconhecidas. Não existe local algum na literatura ou outra fonte em que esses rótulos sejam apresentados.

Essas bases representam descrições geométricas de moléculas, e o objetivo é prever se uma molécula cheira Almíscar (*MUSK*) ou não (Dundar et al., 2008). Cada molécula é um *bag* e cada *bag* possui um conjunto de instâncias, que representam diferentes formas geométricas da molécula. Cada instância é um vetor de característica com 166 dimensões (Andrews et al., 2003), representando 162 características geométricas sobre a forma da molécula, como exemplificado na Figura 4.1, na qual a molécula e centralizada em um sistema de cordenadas e vários eixos são traçados para realizar as medições. Além dessas medições, são determinadas mais 4 características descrevendo a posição do átomo de oxigênio na molécula. Como apresentado na Tabela 4.1, as bases MUSK1 e MUSK2 possuem 476 e 6598 instâncias, respectivamente.

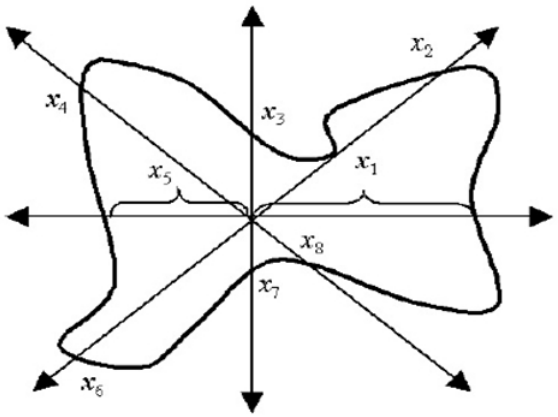


Figura 4.1: Exemplos de caracteristicas geométricas, extraída para cada molécula - *bag* (Figura extraída de Dietterich et al. (1997).)

Com relação à área de recuperação/identificação de imagem, existem três bases de dados que são as mais utilizadas para comparação entre os algoritmos, são elas: Elefante, Raposa e Tigre. Essas bases foram, inicialmente, utilizadas por Andrews et al. (2003) e, posteriormente, em diversos trabalhos para avaliar performances entre diferentes abordagens para AMI (Xiao et al., 2014; Chen et al., 2006a; Chen e Wang, 2004b; Krummenacher e Ong, 2013; Kim e la Torre, 2010; Erdem e Erdem, 2011).

(Elefante) (Tigre) (Raposa)

Figura 4.2: Exemplos de imagens da base: Elefante, Tigre e Raposa.

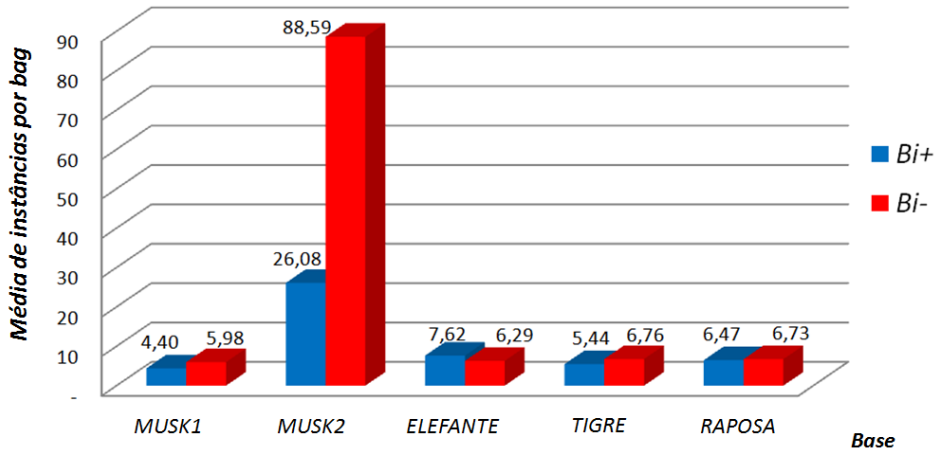


Figura 4.3: Distribuição média das instâncias para cada *bag*, em cada classe.

Essas bases de dados representam características de imagens, como histogramas de cores, momentos de cores, textura, forma de histogramas *etc.*, retiradas de diferentes fotos. Cada foto representa um *bag* e cada região da foto, uma instância. Cada base possui cem *bags* positivos (que contêm o animal do nome da base) e cem *bags* negativos (que contêm fotos sem o animal). O objetivo, então, implica em identificar se uma nova imagem, ou *bag*, contém ou não o animal que nomeia a base.

As fotos foram segmentadas em diferentes regiões (instâncias) usando o algoritmo Blobworld (Carson et al., 1999). Em cada região, ou instância, foram extraídas as características. Como apresentado na Tabela 4.1, as fotos ou *bags* foram segmentadas em 1391 (tigre), 1220 (raposa) e 1320 (elefante) regiões ou instâncias e cada instância com 230 características, ou dimensões.

Tabela 4.1: Base de dados: MUSK, Elefante, Raposa e Tigre.

Dataset	Bags Positivos	Bags Negativos	Total	Número de Características	Número de Instancias	Média de instâncias/Bag
MUSK 1	47	45	92	166	476	5.17
MUSK 2	39	63	102	166	6598	64.69
Elefante	100	100	200	230	1391	6.96
Raposa	100	100	200	230	1220	6.60
Tigre	100	100	200	230	1320	6.10

O gráfico apresentado na Figura 4.3 revela a média de instâncias para cada *bag* em cada classe. Embora a maioria das bases possuam um equilíbrio, com relação à média de instâncias por *bag*, a base MUSK2 possui uma média de quase 90 instâncias por *bag* na classe negativa, enquanto a classe positiva, para a mesma base, possui, aproximadamente, 26 instâncias por *bag*, como média.

Com intuito de entender melhor a distribuição das instâncias em cada *bag*, para cada base de dados, foram obtidos os histogramas apresentados na Figura 4.4. Por meio da análise dessas distribuições, pode-se observar que, para as bases Elefante, Tigre e Raposa, a quantidade de instâncias por *bags* é relativamente equilibrada entre os *bags* positivos e negativos. Em relação à base MUSK1, observa-se que 2 *bags* negativos possuem 40 instâncias enquanto, nos *bags* positivos, os *bags* que possuem mais instâncias estão com 8 instâncias.

Esses dois gráficos (Figuras 4.3 e 4.4), apresentam algumas das características que tornam as bases para Aprendizado de Múltiplas Instâncias tão desafiadoras, sendo necessário um método robusto para tratar estes problemas.

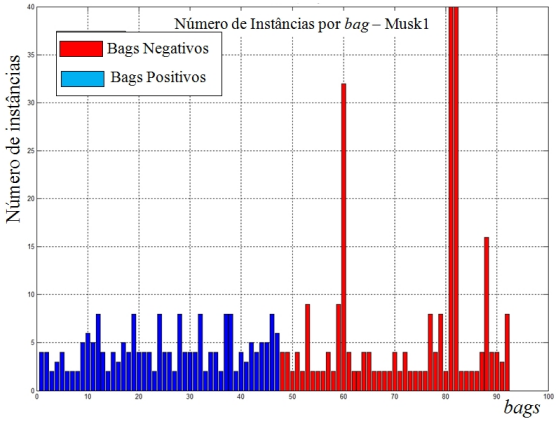
4.1.2 Base de dados COREL

A base de dados da COREL foi publicada pela COREL *Corporation* e possui 2000 imagens naturais. Essas 2000 imagens são divididas em 20 categorias, com 100 imagens cada uma. As categorias são: Africanos e vilas, praias, prédios, ônibus, dinossauros, elefantes, flores, cavalos, montanhas, comida, cachorros, lagartos, modelos, por do sol, carros, cachoeiras, móveis antigos, navios de batalha, ski e desertos. Duas imagens exemplo de cada categoria são apresentadas em cada linha da Figura 4.5.

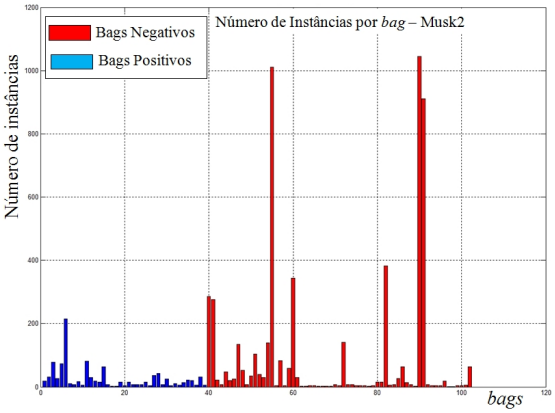
Cada imagem é considerada um *bag* e diferentes regiões dentro das imagens são suas instâncias. Cada instância é representada por um vetor de seis características: as três primeiras representam a média de cada componente do espaço de cor LUV, onde L representa a luminância, e as componentes UV representam as informações sobre a cor. As demais características representam coeficientes de energia, extraída em três bandas de alta frequência, utilizando transformada de Wavelet, com objetivo de extrair informações de textura (Chen et al., 2006a). A Tabela 4.2 apresenta as categorias, com o número de imagens, número de características e a média de instância por categoria.

Como, basicamente, essa base implica um problema de multiclasses e não está projetada para problemas de múltiplas instâncias, foi aplicado para esse experimento a estratégia "um contra todos". Assim, cada base é composta por 100 imagens de uma categoria (classe positiva) e 100 outras imagens uniformemente divididas entre as demais categorias remanescentes (classe negativa).

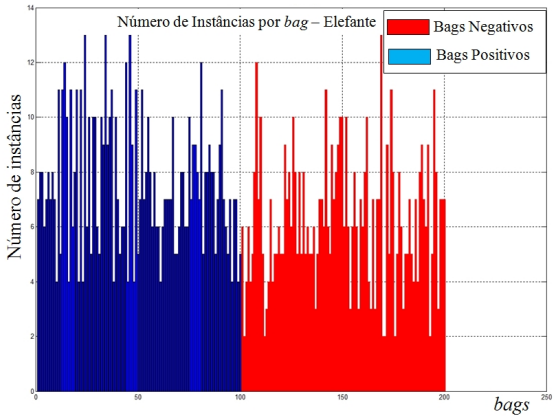
Para utilizar essa base de dados, foi elaborado o seguinte experimento: com a base COREL com 2000 imagens, neste grupo todas as 20 categorias são utilizadas. Para cada experimento, as imagens são divididas em 50% treinamento e 50% teste. Esse procedimento é repetido por cinco vezes e então a média é apresentada. Este foi o mesmo protocolo de teste utilizado para todos os demais algoritmos que utilizaram essa base (Li et al., 2014; Xiao et al., 2014; Erdem e Erdem, 2011; Fu et al., 2011; Chen e Wang, 2004b).



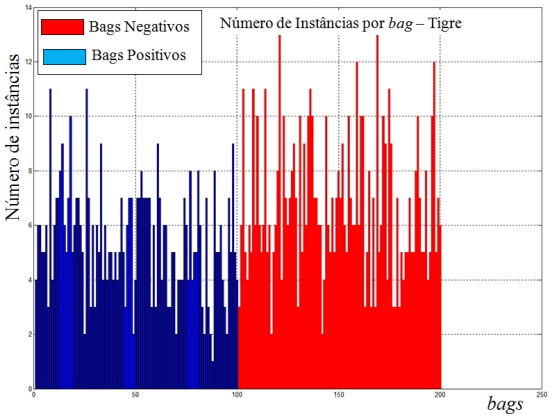
(MUSK1)



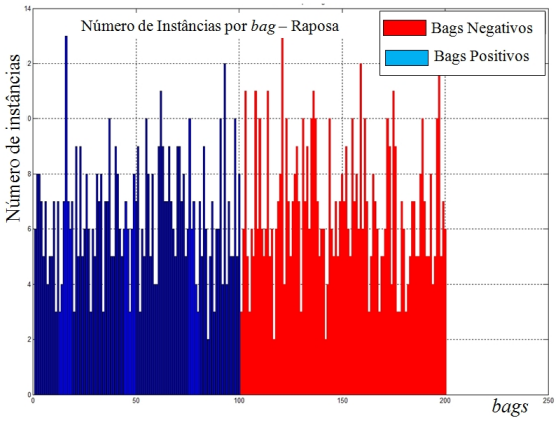
(MUSK2)



(Elefante)



(Tigre)



(Raposa)

Figura 4.4: Histogramas da quantidade de instâncias por *bags* para as bases MUSK1, MUSK2, Elefante, Tigre e Raposa.

Tabela 4.2: Características da base de dados COREL.

Base	Bags Positivo	Bags Negativo	Bags Total	Número de Características	Média de Instancias/Bag
Africanos e vilas	100	100	200	6	4,84
Praias	100	100	200	6	3,54
Prédios históricos	100	100	200	6	3,10
Ônibus	100	100	200	9	7,59
Dinossauros	100	100	200	6	2,00
Elefantes	100	100	200	6	3,02
Flores	100	100	200	6	4,46
Cavalos	100	100	200	6	3,89
Montanhas	100	100	200	6	3,38
Comida	100	100	200	6	7,24
Cachorro	100	100	200	6	3,80
Largatos	100	100	200	6	2,80
Modelos	100	100	200	6	5,19
Pôr do sol	100	100	200	6	3,52
Carros	100	100	200	6	4,93
Cachoeiras	100	100	200	6	2,56
Móveis antigos	100	100	200	6	2,30
Navios de batalha	100	100	200	6	4,32
Ski	100	100	200	6	3,34
Desertos	100	100	200	6	3,65

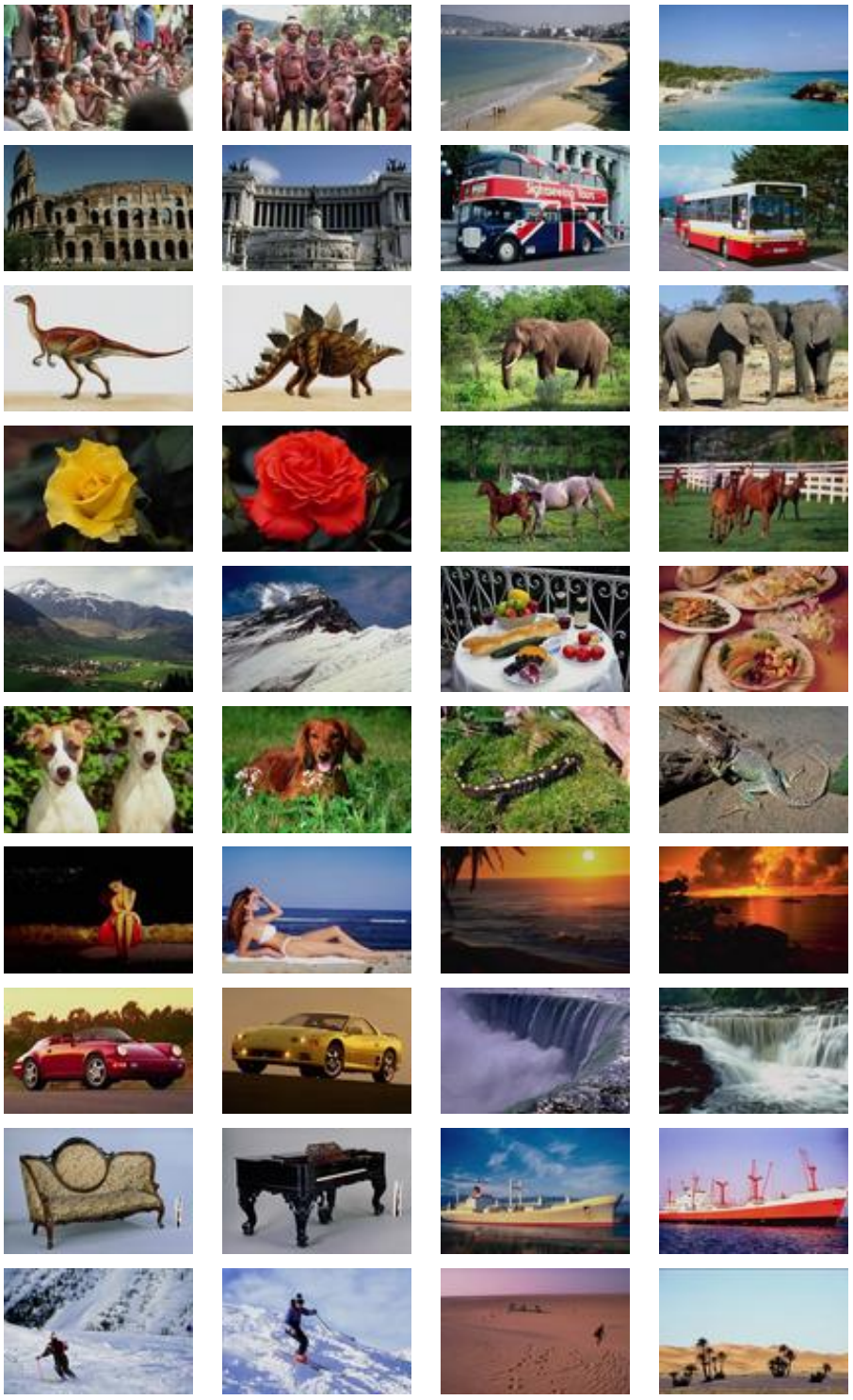


Figura 4.5: Exemplos da base COREL. Cada linha contém dois exemplos de imagens de duas categorias distintas, são elas: Africa, Praias, Prédios, Ônibus, Dinossauros, Elefantes, Flores, Cavalos, Montanhas, Comidas, Cachorros, Largatos, Modelos, Por do sol, Carros, Cascatas, Móveis antigos, Navios de batalhas, Ski e Desertos.

4.2 Análises Estatísticas

Nesta seção são descritos os métodos estatísticos utilizados neste trabalho, a fim de validar os resultados alcançados e as comparações realizadas com os métodos encontrados na literatura.

As análises estatísticas são de extrema importância quando se está comparando resultados de métodos diferentes em várias bases de dados. Em muitos casos, confrontar, simplesmente, a acurácia média entre métodos, em relação aos resultados, pode fornecer análises tendenciosas ou até mesmo não confiáveis.

Infelizmente, apenas alguns trabalhos na literatura reportam os desvios padrões de seus métodos e nenhuma avaliação estatística é empregada para avaliar e comparar seus algoritmos. Essa mesma percepção é, também, relatada em Zh Zhou et al. (2009) e Zhang e Zhou (2006).

Os desvios padrões e intervalo de confiança descritos na Tabela 4.4, para a base de dados MUSK1 e MUSK2, foram encontrados em diferentes trabalhos na literatura e para as bases de dados relativas as imagens (Elefante, Tigre e Raposa), não foram localizados na literatura trabalhos que apresentassemm os desvios padrões ou intervalo de confiança para essas três bases.

Nesse caso, em virtude de a grande maioria dos métodos não apresentarem seus desvios padrões e eles não estarem disponíveis, os resultados apresentados na Tabela 4.3 são utilizados mais para referência em vez de uma rigorosa comparação entre metodologias.

Com o objetivo de melhor avaliar os resultados das metodologias propostas com aquelas da literatura, foram realizadas duas análises estatísticas: a primeira, utilizando-se Intervalo de Confiança (IC) com aqueles que apresentam o IC ou o desvio padrão na literatura. Nesse caso comparando-se somente as bases MUSK1 e MUSK2, uma vez que não são apresentados na literatura desvios padrões ou IC para as bases Elefante, Tigre e Raposa. Na segunda análise, foi realizado o teste de Friedman, levando-se em conta todas as cinco bases.

4.2.1 Intervalo de Confiança

Intervalo de confiança, em estatística, é definido como o intervalo em que a média de um dado parâmetro de um espaço amostral, que tenha a probabilidade de $1 - \alpha$ (α=nível de significância) de conter o valor verdadeiro. Na maioria dos trabalhos, o valor de α é definido como 5%, ou seja, ao afirmarmos que os resultados estão em um intervalo de confiança com nível de confiança igual a 95%, estamos dizendo que nosso erro seria de 5%.

Em alguns trabalhos na literatura (Xiao et al., 2014; Li e Yeung, 2010; Chen et al., 2006a; Fu et al., 2011; Dietterich et al., 1997), são encontrados a acurácia média seguida pelo seu respectivo intervalo de confiança de 95 % (apenas para as bases de dados MUSK1 e MUSK2).

Nesses trabalhos são apresentados seus próprios Desvios Padrões e/ou Intervalos de Confiança bem como os de outros métodos na literatura. Assim, para melhor avaliar os resultados apresentados na Tabela 4.3, foi, também, calculado o Intervalo de Confiança e o Desvio pa-

drão com base em valores de Desvio Padrão e Intervalo de Confiança disponíveis em alguns trabalhos. O cálculo foi realizado de acordo com a Equação 4.1, em que $\bar{X}$ representa a media, Z representa o coeficiente de confiança, α constitui o nível de confiança, σ refere-se ao desvio padrão e n representa o tamanho a amostra.

$$CI = \overline{X} \pm Z_{\alpha/2} \times \frac{\sigma}{\sqrt{(n)}} \tag{4.1}$$

Quando o Intervalo de Confiança de duas variáveis não se sobrepõe, podemos dizer, então, com 95 % de confiança, que existe uma diferença estatisticamente relevante entre estas variáveis. Porém, se existe uma sobreposição entre os Intervalos de Confiança, uma análise mais aprofundada deve ser conduzida. Então, na existência de duas variáveis com seus Intervalos de Confiança se sobrepondo, para afirmarmos, com 95 % de confiança, que elas são diferentes a seguinte inequação deve ser respeitada: $|\overline{x}_1 - \overline{x}_2| - 1.96\sqrt{SE_1^2 + SE_2^2} > 0$. Em que 1.96 representa o nível de confiança de acordo com a tabela de valores de Z, e SE o *Standard Error*, ou Erro padrão.

4.2.2 O teste de Friedman

A fim de comparar os resultados alcançados pela metodologia proposta e as demais da literatura, usando as cinco bases de dados, foi proposto o teste de Friedman, como descrito em Demsar (2006). Para o teste de Friedman, adotou-se a hipótese nula, H_0. Esta atesta que todos os algoritmos em análise são equivalentes, então, os *Ranks* médios são iguais: $R(\pounds) = \frac{1}{N}\sum_N^i R(\pounds_i)$. Assim dado K algoritmos, avaliados em M base de dados, com seu respectivos Ranks médios $R(\pounds_i)$ [2], o teste estatístico de Friedman é dado por:

$$F_F = \frac{(M-1)X_F}{M(K-1) - X_F} \tag{4.2}$$

onde

$$X_F = \frac{12.M}{K(K+1)}\left[\sum_N^i R(\pounds_i) - \frac{K(K+1)^2}{4}\right] \tag{4.3}$$

na qual a distribuição é de acordo com a distribuição F, usando $K-1$ e $(K-1)(M-1)$ graus de liberdade.

Nos casos em que a hipótese nula for rejeitada, de acordo com Demsar (2006), pode-se seguir a avaliação com o teste *post-hoc* de Nemenyi (Nemenyi, 1963). Nesse teste, a diferença entre dois algoritmos distintos são consideradas significativas, com um nível de $1 - \alpha$. Se a diferença entre os dois ranks médios for de, pelo menos, o valor encontrado na Equação 4.4,

[2] No teste de Friedmam os algoritmos são ranqueados para cada base de dados separadamente. O melhor algoritmo recebe *Rank* 1, o segundo *Rank* 2 e assim por diante. Em caso de empate, um *Rank* médio é atribuído, para cada classificador $\pounds_i \in \pounds$

onde CD é a Diferença Crítica ou *Critical Difference* e $q\alpha$ representa os valores críticos de acordo com a Tabela em Demsar (2006).

$$CD = q\alpha\sqrt{\frac{L(L+1)}{6M}} \tag{4.4}$$

4.3 Resultados

Nesta seção são apresentados os resultados alcançados pelas metodologia propostas e os resultados da literatura. São apresentadas, também, análises estatísticas sobre os resultados, a fim de se extrair conclusões mais precisas sobre a eficiência da proposta.

Todos os valores reportados dos algoritmos da literatura (acurácia média, desvio padrão e/ou intervalo de confiança), apresentados nas tabelas deste capítulo, foram extraídos, diretamente, dos artigos originais de seus respectivos autores. Alguns valores de intervalo de confiança foram calculados com base em desvios padrões reportados.

Durante os experimentos, foram conduzidos diversos testes utilizando o pacote MILL *Toolkit for Multiple Instance Learning* (Yang, 2008) e WEKA (*Waikato Environment for Knowledge Analysis*) (Hall et al., 2009), os quais possuem diversos algoritmos implementados e, em alguns casos, o próprio algoritmo disponível de alguns algoritmos, porém, os resultados encontrados foram significativamente inferiores àqueles reportados por seus autores em seus artigos.

Foi decidido, então, para efeitos de comparação, reportar os valores apresentados pelos autores em seus respectivos artigos originais. Este mesmo procedimento é utilizado em todos os outros trabalhos que realizaram comparações entre seus métodos (Xiao et al., 2014; Chen et al., 2006a; Krummenacher e Ong, 2013; Carbonneau et al., 2016; Erdem e Erdem, 2011; Li e Yeung, 2010; Maron e Lozano-Perez, 1998) e outros.

4.3.1 Resultados para as Bases MUSK, Elefante, Raposa e Tigre.

A Tabela 4.3 apresenta o resultado alcançado pelas metodologias propostas e de outros quarenta e sete algoritmos encontrados na literatura. Alguns desses algoritmos foram testados somente com as bases MUSK1 e MUSK2, e outros somente com as bases Elefante, Raposa e Tigre. Com exceção dos algoritmos Citação KNN (Wang e Zucker, 2000) e Bayesian-kNN(Wang e Zucker, 2000), que usam o protocolo de validação *one-leave-out*, em todos os demais experimentos, incluindo as metodologias propostas, foram utilizados 10 $\times$ 10 *fold* de validação cruzada (Haykin, 1994).

De acordo com os resultados reportados na Tabela 4.3, as metodologias propostas foram capazes de fornecer resultados competitivos entre os melhores artigos da literatura. De acordo com os resultados, avaliando-se somente a acurácia média, o algoritmo MILKDE obteve o melhor resultado da literatura para as bases Elefante e Raposa. Com relação à base

MUSK1 e MUSK2, a proposta está entre as melhores dentre os 42 algoritmos avaliados. Outro bom aspecto da metodologia proposta é o fato de ser o único algoritmo que obteve o melhor desempenho em mais de uma base avaliada. Com relação ao MILKDE2, sem repescagem, utilizando-se somente uma instância de cada *bag*, conseguiu-se obter bons resultados comparando com outros métodos. Esse último aspecto, apresentado para o MILKDE2, constitui uma forte evidência da boa qualidade da seleção de instâncias da metodologia proposta.

Os algoritmos que apresentam seus desvios padrões e/ou intervalos de confiança são apresentados na Tabela 4.4. A Figura 4.6 fornece um resumo visual da Tabela 4.4.

Expandindo as análises sobre os resultados, foram feitas duas análises estatísticas: Intervalo de Confiança e Teste de Friedman.

Análise por Intervalo de Confiança

A primeira análise foi realizando utilizando o Intervalo de Confiança com os principais algoritmos da literatura [3], para as bases MUSK1 e MUSK2.

Para a base MUSK1, o MILKDE obteve performance equivalente aos melhores algoritmos da literatura, como apresentado na Tabela 4.4. Observa-se que em alguns casos ocorreram sobreposição com os melhores algoritmos da literatura para a base MUSK1: SMILE, CK-MIL, MILD I e MILD B, como pode ser visualizado na Figura 4.6, que ilustra a Tabela 4.4. Entretanto, realizando uma análise mais crítica sobre essas sobreposições, de acordo com o procedimento descrito na Seção passada, observou-se que os algoritmos SMILE, MILD I e CK MIL são ligeiramente melhores que a metodologia proposta, embora haja sobreposição dos resultados.

Para a base MUSK2, a metodologia proposta conseguiu obter resultados competitivos entre os melhores algoritmos. No entanto, estatisticamente inferior aos algoritmos SMILE, ck-MIL, MILIS, DD-SVM e MILD I.

É válido destacar que os algoritmos MILD I e DD-SVM, embora tenham resultados expressivos para a base MUSK, estes não possuem bons resultados para a base Elefante, Tigre e Raposa, como pode ser observado na Tabela 4.3. E, para os algoritmos ck-MIL e MILIS, não foram encontrados seus respectivos resultados para as outras bases.

Análise por teste de Friedman

Ao contrário da análise anterior, realizada somente sobre a base MUSK, foi realizado também o teste de Friedman sobre todas as cinco bases de dados.

Aplicando o teste de Friedman sobre o MILKDE e outros cinco melhores algoritmos (de

[3] Para as bases Elefante, Tigre e Raposa, não foram realizadas essas análises, pois não foi encontrado em nenhum trabalho seus desvios padrões ou intervalos de Confiança. Assim, a análise do Intervalo de Confiança foi realizada somente para a base MUSK.

Tabela 4.3: Resultados para o MILKDE, MILKDE2 e outros algoritmos disponíveis na literatura: acurácia média (%), para 10 × 10 *fold* de validação cruzada.

Dataset	MUSK1	MUSK2	Elefante	Tigre	Raposa	Média
MILKDE	89,20	87,50	**87,40**	81,10	**66,80**	82,40
MILKDE 2	85,00	86,00	86,00	80,00	64,00	80,20
SMILE (Xiao et al., 2014)	91,30	**91,60**	84,30	86,50	65,90	**83,92**
MILES (Chen et al., 2006a)	86,30	87,70	84,10	80,70	63,00	80,54
eMIL (Krummenacher e Ong, 2013)	84,50	86,00	84,00	**88,80**	58,30	80,32
EoSVM(RISIS) (Carbonneau et al., 2016)	88,80	89,50	84,60	82,50	61,10	81,30
EoSVM(RS) (Carbonneau et al., 2016)	82,80	83,60	82,50	73,70	57,90	76,10
ELM-MIL (Wang et al., 2015)	86,50	85,80	76,70	74,60	59,50	76,62
MILDS (Erdem e Erdem, 2011)	90,90	86,10	84,80	81,50	64,30	81,52
MILD I (Li e Yeung, 2010)	91.30	89,60	83,20	73,40	49,10	77,28
MILD B (Li e Yeung, 2010)	88,30	86,80	82,90	75,80	55,00	77,76
DD (Maron e Lozano-Perez, 1998)	88,90	89,20	81,45	72,20	59,40	78,23
EM DD (Zhang e Goldman, 2001)	84,80	84,90	78,30	72,10	56,10	75,24
mi SVM (linear) (Andrews et al., 2003)	87,40	83,60	82,20	78,40	58,20	77,96
MI SVM (linear) (Andrews et al., 2003)	77,90	84,30	81,40	84,00	57,80	77,08
DD SVM (Chen e Wang, 2004b)	85,80	91,30	81,60	75,80	55,70	78,04
MIForest (Leistner et al., 2010)	85,00	82,00	84,00	82,00	64,00	79,40
Ins-KI-SVM (Li et al., 2009)	84,00	84,40	83,50	82,90	63,40	79,64
BAG-KI-SVM (Li et al., 2009)	88,00	82,00	84,50	85,00	60,50	80,00
MICA (Mangasarian e Wild, 2008)	84,40	90,50	82,50	82,00	62,00	80,28
AL SVM (Gehler e Chapelle, 2007)	85,70	86,20	79,50	78,50	63,50	78,68
AW SVM (Gehler e Chapelle, 2007)	85,70	83,80	82,00	83,00	63,50	79,60
SDP (Guo, 2009)	69,50	61,30	74,80	73,60	56,80	67,20
GPMIL (Kim e la Torre, 2010)	89,50	87,30	83,80	87,40	65,80	82,76
MIMEL WS (Xu et al., 2011)	84,90	82,60	85,00	82,50	61,00	79,20
Citação KNNC (Xu et al., 2011)	90,30	83,30	83,00	58,50	58,50	74,72
1NNC (Woznica e Kalousis, 2010)	83,30	78,00	79,50	77,50	60,00	75,66
Metric Learner (Woznica e Kalousis, 2010)	83,50	79,50	85,50	83,50	65,00	79,40
MILR (Dong, 2006)	73,51	78,51	79,90	75,30	57,35	72,91
MILR ARITH (Dong, 2006)	85,03	81,26	79,85	76,35	55,20	75,54
Adaboost M1 - Optmall ball (Dong, 2006)	87,50	83,18	79,55	66,25	48,65	73,03
Citação kNN (Wang e Zucker, 2000)	**92,40**	89,20	−−	−−	−−	−−
Baysian kNN (Wang e Zucker, 2000)	90,20	82,40	−−	−−	−−	−−
RIPPER MI (Zucker e Chevaleyre, 2001)	88,00	77,00	−−	−−	−−	−−
BP MIP (hua Zhou e ling Zhang, 2002)	83,70	80,40	−−	−−	−−	−−
MULTIINST (Auer, 1997)	76,70	84,00	−−	−−	−−	−−
RELIC (Ruffo, 2000)	83,70	87,30	−−	−−	−−	−−
CK MIL (Li et al., 2014)	88,70	88,20	−−	−−	−−	−−
MissSVM (Zhou e Xu, 2007)	87,60	80,00	−−	−−	−−	−−
MI KERNEL (Thomas et al., 2002)	88,00	89,30	−−	−−	−−	−−
MILIS (Fu et al., 2011)	88,60	91,10	−−	−−	−−	−−
APR (Dietterich et al., 1997)	**92,40**	89,20	−−	−−	−−	−−
mi SVM (poly) (Andrews et al., 2003)	−−	−−	78,10	78,10	55,20	−−
mi SVM (rbf) (Andrews et al., 2003)	−−	−−	80,00	78,00	57,90	−−
MI SVM (poly) (Andrews et al., 2003)	−−	−−	79,00	58,80	59,40	−−
MI SVM (rbf) (Andrews et al., 2003)	−−	−−	73,10	66,66	58,80	−−
MILEAGE (Zhang et al., 2013)	−−	−−	84,50	84,00	64,50	−−

Tabela 4.4: Resultados: acurácia média (%), desvio padrão (%) para 10 × 10 *fold* de validação cruzada (em parênteses) e o intervalo de confiança de 95 % (em colchete).

Dataset	MUSK1 (em %)	MUSK2 (em %)
MILKDE	89,2 (± 2,7) [87,5 90,9]	87,5 (± 2,9) [85,5 89,5]
MILKDE2	85,0 (± 2,8) [83,3 86,7]	86,0 (± 2,9) [84,4 87,6]
SMILE (Xiao et al., 2014)	91,3 (± 1,3) [90,5 92,1]	91,6 (± 1,1) [90,9 92,3]
DD (Maron e Lozano-Perez, 1998)	88,0 (± 3,5) [85,8 91,2]	84,0 (± 2,8) [82,3 85,7]
EM DD (Zhang e Goldman, 2001)	84,8 (± 2,8) [83,1 86,5]	85,2 (± 2,2) [83,8 86,6]
mi SVM (linear) (Andrews et al., 2003)	87,4 (± 3,3) [85,4 89,5]	83,6 (± 2,8) [81,7 85,3]
MI SVM (linear) (Andrews et al., 2003)	77,9 (± 3,6) [75,7 80,1]	84,3 (± 2,1) [83,0 85,6]
DD SVM (Chen e Wang, 2004b)	85,8 (± 1,7) [84,8 86,9]	91,3 (± 1,3) [90,5 92,1]
MULTIINST (Auer, 1997)	76,7 (± 4,3) [74,0 79,4]	84,0 (± 3,7) [81,7 86,3]
CK MIL (Li et al., 2014)	88,7 (± 3,2) [86,7 90,7]	88,2 (± 3,6) [86,0 90,4]
ck MIL (Li et al., 2014)	91,8 (± 2,4) [90,3 93,3]	89,8 (± 1,1) [89,1 90,5]
MissSVM (Zhou e Xu, 2007)	87,6 (± 2,8) [85,9 89,3]	80,0 (± 2,3) [78,6 81,4]
MILES (Chen et al., 2006a)	86,3 (± 2,3) [84,9 87,7]	87,7 (± 2,3) [86,3 89,1]
MILD I0 (Li e Yeung, 2010)	87,9 (± 0,9) [87,3 88,5]	84.4 (± 1,5) [83,5 85,2]
MILD I (Li e Yeung, 2010)	91,1 (± 0,9) [90,5 91,7]	89.6 (± 1,5) [88,7 90,4]
MILD B (Li e Yeung, 2010)	90,2 (± 1,2) [89,5 90,9]	86,8 (± 1,1) [85,8 87,1]
MILIS (Fu et al., 2011)	88,3 (± 4,5) [85,8 91,5]	91,1 (± 2,7) [89,4 92,8]
APR (Dietterich et al., 1997)	92,4 (± 8,7) [87,0 97,8]	89,2 (± 9,7) [83,2 95,2]

acordo com o ranking do Teste de Friedman), por meio da Equação 4.2, foi encontrado o valor de F_F=1.5821, distribuído de acordo com a distribuição F de 5 (numerador) e 20 (denominador) graus de liberdade. Com esses valores, foi encontrado o valor crítico F(5,20)=2.710, utilizando α=0.05, então $F_F < F(5, 20)$.

Dessa forma, foi observado que a hipótese nula H_0, a qual atesta que todos os algoritmos em análise são equivalentes, não pode ser rejeitada. Assim, é possível declarar que a abordagem proposta possui desempenho estatisticamente equivalente aos cinco melhores algoritmos da literatura, de acordo com o teste de Friedman, quando avaliadas as cinco bases de dados.

4.3.2 Resultados para a base COREL

Avaliando os resultados sobre a base COREL, com os testes realizados para a base de 2000 imagens, a Tabela 4.5 apresentam os resultados para cada base, juntamente a seus respectivos desvios padrões. Os resultados apresentados nessa tabela são os resultados publicados no trabalho de Li et al. (2014) para a base com 2000 imagens. Não foram encontrados outros trabalhos na literatura que publicaram seus resultados para cada subclasse das respectivas bases. Para os testes reportados para a base COREL, somente o algoritmo MILKDE foi apresentado. Os resultados encontrados para a segunda metodologia proposta, MILKDE2, foram essencialmente os mesmos encontrados em MILKDE. Essa evidência indica que as instâncias utilizadas para a etapa de Repescagem não afetaram o resultado final.

Resultados de outros algoritmos, além dos apresentados na Tabela 4.5, são encontrados na literatura, porém é apresentada somente a média final 20 categorias. A Tabela 4.6 apresenta

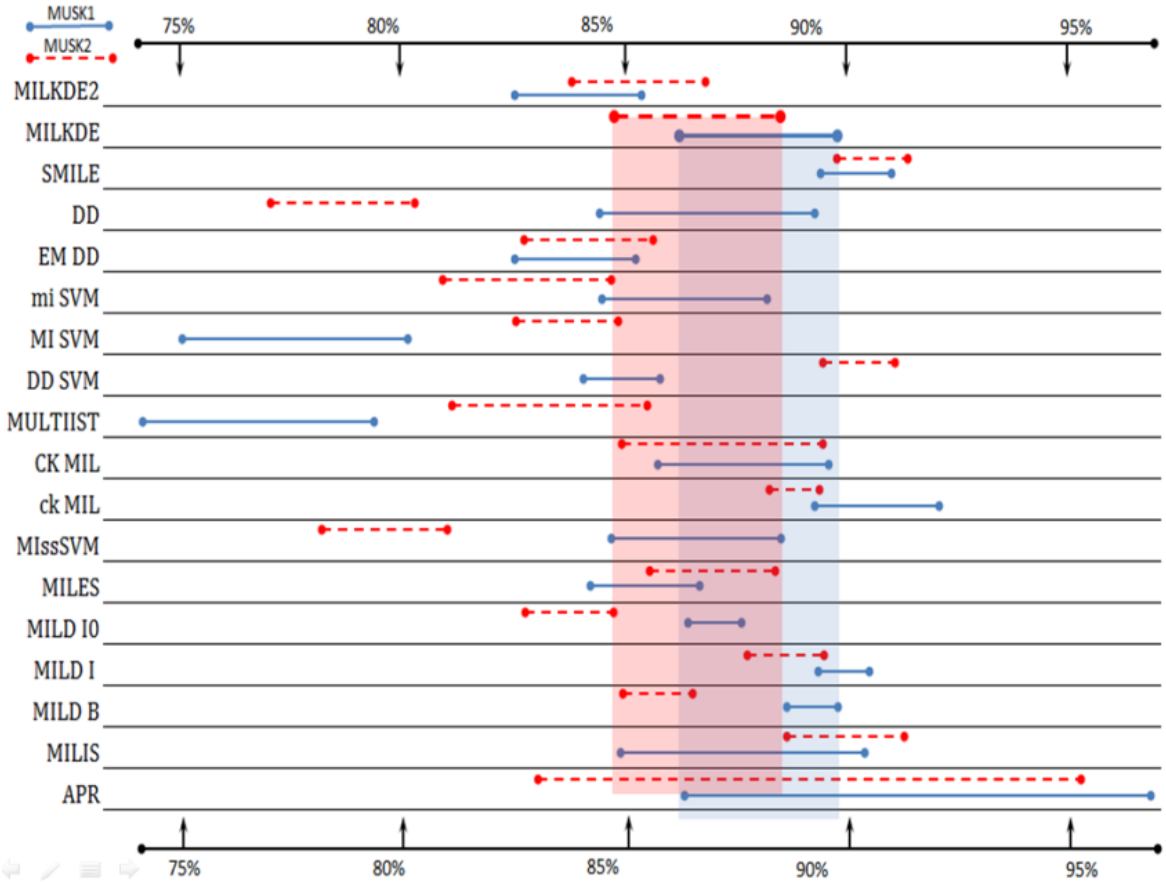


Figura 4.6: Intervalo de Confiança - Gráfico resumo da Tabela 4.4. As áreas hachuradas representam os resultados para as metodologias MILKDE e MILKDE2, para as bases de dados MUSK1 e MUSK2.

esses outros resultados.

Análise por Intervalo de Confiança

Realizando uma análise utilizando o Intervalo de Confiança, com os algoritmos que fornecem seus resultados para todas as sub-bases, se obtém o gráfico apresentado na Figura 4.7. Neste gráfico é possível observar que, com exceção do algoritmo MI SVM, todos obtiveram sobreposição dos resultados. Este fato se deve aos resultados possuírem um alto desvio padrão, característica de uma base tão heterogênea. Entretanto, essa sobreposição de resultados necessita de uma análise mais crítica, como descrito na Seção 4.2.1. Após o cálculo das sobreposições fica constatado que somente o método CK MIL é compatível com MILKDE. Para os demais métodos, pode-se dizer, com 95% de confiança, que são estatisticamente inferiores a metodologia proposta.

Tabela 4.5: Acurácia Média (%) e Desvio Padrão (%) para a Base COREL 2000 Imagens.

	Algoritmos						
Base	MILKDE	ck MIL	MI-Kernel	MI-SVM	DD-SVM	MILES	MissSVM
Africanos e vilas	**74,9 ± 1,6**	63,4 ± 0,9	61,8 ± 1,0	51,6 ± 1,3	58,4 ± 1,2	58,3 ± 1,6	55,6 ± 1,2
Praias	**75,8 ± 1,3**	62,6 ± 1,3	61,1 ± 1,1	51,3 ± 2,1	59,5 ± 1,6	59,1 ± 1,9	56,3 ± 1,5
Prédios históricos	**72,0 ± 0,8**	67,5 ± 2,1	65,9 ± 1,6	51,7 ± 1,9	61,5 ± 1,1	61,6 ± 1,2	57,6 ± 1,7
Ônibus	81,7 ± 2,1	**89,8 ± 1,1**	85,6 ± 1,2	61,3 ± 1,5	80,1 ± 1,8	82,1 ± 1,3	73,6 ± 1,5
Dinossauros	**95,3 ± 2,2**	92,7 ± 1,2	90,8 ± 1,2	61,8 ± 1,8	83,1 ± 1,3	88,2 ± 1,7	80,8 ± 1,1
Elefantes	**73,7 ± 1,5**	73,5 ± 1,5	70,3 ± 1,4	51,2 ± 1,1	67,3 ± 1,8	67,6 ± 1,8	63,7 ± 1,2
Flores	**90,5 ± 2,0**	89,2 ± 1,3	86,8 ± 2,0	60,7 ± 1,8	80,4 ± 1,6	82,2 ± 1,9	79,6 ± 2,2
Cavalos	**88,6 ± 1,1**	83,6 ± 1,2	81,9 ± 1,6	54,5 ± 1,5	75,1 ± 1,2	79,1 ± 1,2	71,8 ± 1,8
Montanhas	**72,7 ± 1,8**	69,2 ± 1,1	66,8 ± 1,9	54,2 ± 1,2	62,7 ± 2,1	62,7 ± 1,7	60,3 ± 1,5
Comida	76,9 ± 1,7	**78,5 ± 1,6**	76,7 ± 2,1	58,2 ± 1,9	72,1 ± 1,7	71,3 ± 1,6	67,3 ± 1,9
Cachorro	**73,5 ± 1,9**	70,5 ± 2,0	68,9 ± 2,7	51,8 ± 1,1	65,3 ± 2,3	65,3 ± 2,2	62,5 ± 2,5
Largatos	**68,9 ± 0,7**	68,8 ± 1,2	66,9 ± 1,6	51,6 ± 2,2	61,4 ± 1,5	64,1 ± 1,2	61,8 ± 1,3
Modelos	**79,6 ± 0,8**	64,1 ± 1,5	61,9 ± 1,9	53,2 ± 1,8	57,9 ± 1,2	59,2 ± 1,6	59,3 ± 1,2
Pôr do sol	**82,6 ± 2,1**	82,3 ± 1,3	81,5 ± 1,1	60,2 ± 1,5	76,3 ± 1,3	76,5 ± 1,7	72,9 ± 1,3
Carros	75,3 ± 2,4	**76,7 ± 2,1**	74,9 ± 1,6	55,2 ± 1,8	71,3 ± 1,7	71,5 ± 1,5	70,2 ± 1,8
Cachoeiras	**75,6 ± 1,4**	69,8 ± 1,9	65,8 ± 1,9	50,9 ± 1,5	61,3 ± 1,3	62,9 ± 1,7	61,5 ± 2,1
Móveis antigos	**84,1 ± 1,4**	63,3 ± 1,1	61,8 ± 1,2	51,1 ± 1,6	59,5 ± 1,8	59,3 ± 2,3	58,9 ± 1,8
Navios de batalha	80,1 ± 1,0	**86,8 ± 1,6**	82,9 ± 2,0	56,8 ± 1,9	75,1 ± 1,5	79,8 ± 2,2	71,7 ± 1,7
Ski	**76,8 ± 2,2**	67,5 ± 1,1	65,6 ± 1,5	53,8 ± 1,6	61,2 ± 1,1	62,5 ± 1,7	62,2 ± 1,3
Desertos	56,1 ± 2,3	**64,5 ± 1,7**	61,9 ± 1,8	51,4 ± 1,2	60,5 ± 1,6	58,7 ± 1,9	57,9 ± 2,1
Média	**77,8 ± 8,5**	74,2 ± 9,9	72,0 ± 9,8	54,6 ± 3,8	67,5 ± 8,2	68,6 ± 9,5	65,2 ± 7,7

Tabela 4.6: Resultados para a base COREL: Acurácia média (%) para base com 2000 Imagens.

Algoritmos	Base COREL 2000 (%)
MILKDE	77,8
MILES (Chen et al., 2006a)	68,7
K-means-SVM	52,3
MILIS (Fu et al., 2011)	70,1
mi-SVM (Andrews et al., 2003)	53,7
MILDS (Erdem e Erdem, 2011)	69,4
MILD B (Li e Yeung, 2010)	67,7
MIForest (Leistner et al., 2010)	66,0

Análise por teste de Friedman

Aplicando-se o teste de Friedman, Equação 4.2, sobre os resultados encontrados e mais os outros 6 algoritmos avaliados para a base COREL com 2000 imagens, obteve-se o valor de F_F=143,13, de acordo com a distribuição F para 6(numerador) e 114(denominador) graus de liberdade. Com esses valores, foi encontrado o valor de F crítico de F(6,114)= 2.1799 utilizando α=0.05, então, $F_F > F(4, 114)$.

Dado esse resultado, em que a hipótese nula H_0 foi rejeitada, pode-se, nesse caso, seguir com o teste *post-hoc*. Nessa análise, como descrito na Seção 4.2.2, foi calculado o valor crítico,

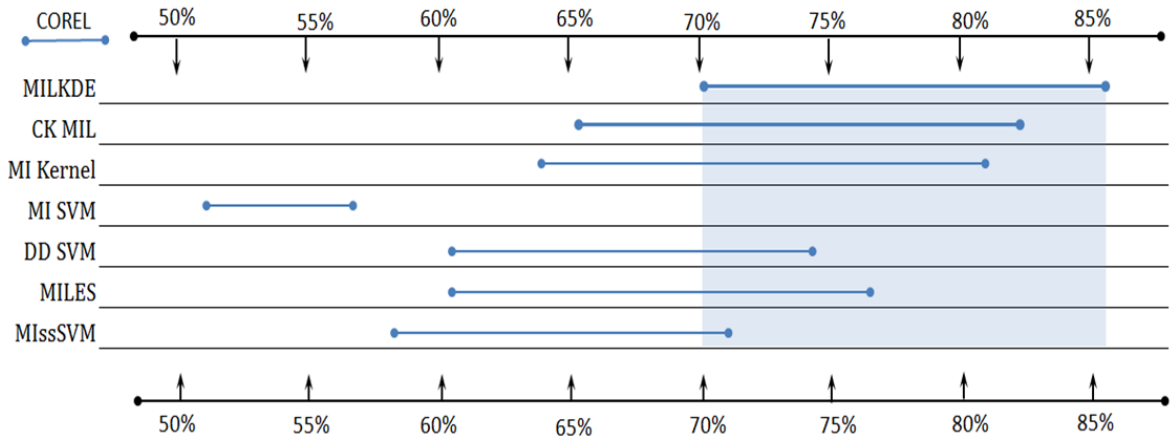


Figura 4.7: Intervalo de Confiança - Gráfico resumo dos cálculos do Intervalo de Confiança para os resultados da Tabela 4.5. As áreas hachuradas representam os resultados para a metodologia MILKDE aplicada à base COREL.

conforme Equação 4.4, utilizando o teste Bonferroni-Dunn para 7 classificadores. Para esse experimento, de acordo com a tabela encontrada em Demsar (2006), o valor é igual a $q_{0,05} = 2,638$, e o valor crítico encontrado foi de $CD = 1,8021$.

Dessa forma, um classificador $\pounds_i$, para ser equivalente a metodologia proposta $\pounds_j$, tem que satisfazer a seguinte inequação: $|R(\pounds_j) - R(\pounds_i)| < CD$. A Tabela 4.7 apresenta os valores das diferenças entre os classificadores. Como pode ser observado na Tabela 4.7, a metodologia proposta ficou com performance estatisticamente equivalente aos métodos ck MIL (Li et al., 2014) e MI-Kernel (Thomas et al., 2002) e superior aos demais.

Tabela 4.7: Análise das diferenças de *Rank*, de acordo com teste de Friedmam para a base Corel 2000 imagens.

Algoritmos	$\|R(\pounds_j) - R(\pounds_i)\|$	Diferença
ck MIL (Li et al., 2014)	| 1.6500-1.7000|	0.0500
MI-Kernel (Thomas et al., 2002)	| 1.6500-2.9000|	1.2500
MI-SVM (Andrews et al., 2003)	| 1.6500-7.0000|	**5.3500**
DD-SVM (Chen e Wang, 2004b)	| 1.6500-4.6750|	**3.0250**
MILES (Chen et al., 2006a)	| 1.6500-4.3000|	**2.6500**
MissSVM (Zhou e Xu, 2007)	| 1.6500-5.7750|	**4.1250**

4.4 Discussões

Avaliando-se as diferentes abordagens da literatura, é notável que a maioria dos melhores algoritmos utilizam abordagens baseadas em seleção de instâncias (Xiao et al., 2014; Li et al., 2014; Xiao et al., 2010b; Chen et al., 2006a; Erdem e Erdem, 2011; Chen e Wang, 2004b; Fu et al., 2011), e essa abordagem vem sendo utilizada na maioria dos trabalhos mais recentes. Esses algoritmos permitem identificar instâncias realmente positivas, nos *bags* positivos, para, assim, transformar o problema de Aprendizado de Múltiplas Instâncias em Problemas tradicionais de aprendizado supervisionado.

Analisando-se os trabalhos da literatura, observa-se que em relação a todos os algoritmos, o algoritmo SMILE é o que mais se destaca em termos de desempenho. Seus resultados indicam que sua estratégia, em usar todas as instâncias associando pesos diferentes a elas, funciona bem para as bases de dados avaliadas. O algoritmo SMILE possui a melhor acurácia média entre todos os algoritmos avaliados e é uma referência na literatura.

Alguns métodos como MILIS, MILDS, e outros, se baseiam na distribuição das instâncias negativas (as quais se sabem seguramente que todas são negativas) para selecionar suas instâncias. Esse tipo de abordagem supõe que as instâncias dos *bags* negativos tenham uma coerência entre si e, para as bases testadas, realmente obtiveram bons resultados. Porém, em Aprendizado de Múltiplas Instâncias, as instâncias negativas podem ser qualquer coisa e não necessariamente ter coerência entre si, por exemplo: digamos que, na base de dados Elefante, tenhamos uma foto com um elefante em uma selva. Assim, teremos um *bag* positivo com instâncias positivas (regiões da imagem onde o elefante esteja) e instâncias negativas (regiões sem o elefante: vegetação, árvore, rio, plantas, *etc*...). Porém, nessa mesma base, podemos ter um *bag* negativo que seja a imagem de uma praia. Certamente, neste caso, teremos um *bag* negativo, em que regiões dessa imagem nada têm em comum com as regiões do *bag* positivo ou com outras próprias de *bags* negativos. E, quando esses métodos assumem que existem uma distribuição modelável das instâncias negativas, e se baseiam somente na classe negativa, eles estão subjugando esse aspecto essencial do AMI.

Dentre todos os algoritmos, o APR e o Citation kNN apresentam os melhores resultados para a base MUSK1. É válido mencionar que, de acordo com Xiao et al. (2010b) o e outros autores, o motivo de a APR possuir tão bons resultados para a base de dados MUSK1 e MUSK2 ocorre em virtude de o algoritmo ter sido desenvolvido especialmente para essas duas bases, não trabalhando bem para outras bases de dados. Ainda sobre Citation kNN, além do argumento anterior, os autores conduziram e publicaram seus resultados utilizando validação do tipo *one-leave-out*, o que é diferente de todos os outros trabalhos publicados que utilizam $10 \times$ *fold* de validação cruzada.

A metodologia proposta MILKDE foi capaz de alcançar bons resultados em quase todas as bases. Quando avaliado nas cinco principais bases de dados: MUSK1, MUSK2, Elefante, Tigre e Raposa, a metodologia somente não obteve bons resultados para a base do Tigre. Realizando-

se análises mais profundas sobre essa base, foi observado que, dentre todas as avaliadas, a base Tigre apresenta o maior desvio padrão para a classe positiva. Considerando-se que o MILKDE é baseado em distância e possui seu melhor desempenho sobre os acertos das classes positivas, esse desvio padrão alto tem interferência direta sobre os resultados alcançados para esta base. A Imagem 4.8 apresenta um gráfico radar que resume os resultados alcançados pelos melhores algoritmos. Observa-se que a metodologia proposta consegue obter resultados bem próximos e algumas bases superando o método SMILE que é um dos melhores da Literatura, quando se é avaliado a média geral para as cinco principais bases.

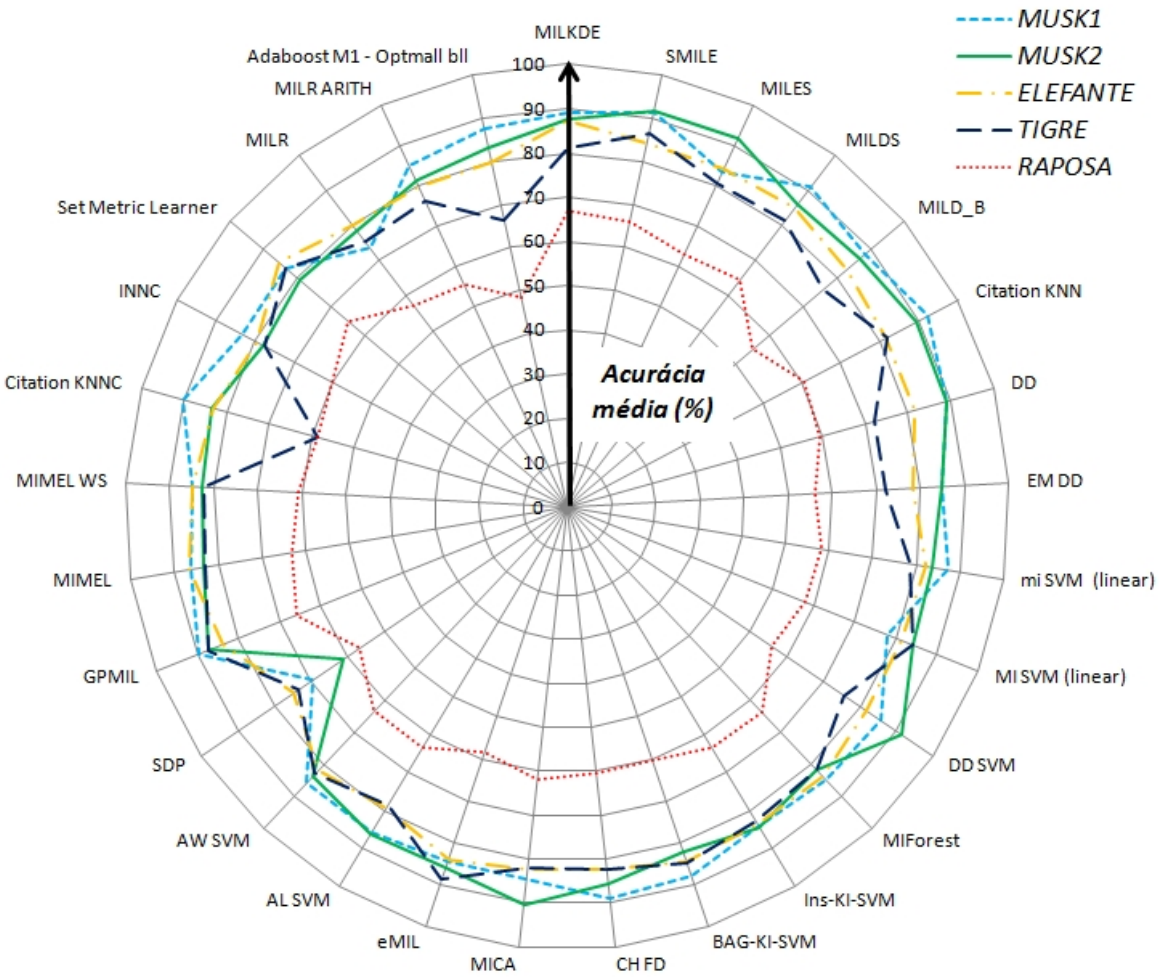


Figura 4.8: Gráfico radar comparativo entre a metodologia proposta MILKDE e alguns algoritmos da literatura.

Os resultados para a base COREL pela metodologia proposta também foram bem significativos quando comparado aos demais. Devido à Base COREL ser composta de várias outras bases, tendo uma característica bem heterogênea, as análises estatísticas são conservadoras e indicam igualdade entre a metodologia proposta e outros dois algoritmos. Porém, realizando

uma análise mais próxima e individual para cada sub-base, observamos que os resultados da metodologia proposta para a base COREL são bem expressivos. Para a base COREL 2000, dentre as 20 categorias de imagens, o MILKDE apresentou o melhor resultado em 15 (75%) e obteve a melhor média dentre todos os algoritmos comparados. Observa-se, também, que, para as sub-bases: Africanos, Praias, Cavalos, Modelos, Cachoeiras, Móveis antigos e Ski, a diferença entre a metodologia proposta e os demais é bem considerável.

4.5 Conclusões do capítulo

Neste capítulo foram detalhados os experimentos conduzidos a fim de avaliar as metodologias propostas nessa tese. Inicialmente, foram apresentadas e discutidas as principais bases de testes realizadas como *benchmark* na literatura e os testes estatísticas realizadas para validar os resultados obtidos. Os resultados alcançados mostram que as metodologias propostas são competitivas com os melhores algoritmos na literatura, obtendo, em algumas bases, os melhores resultados dentre todos.

Capítulo 5

Seleção de Instâncias em Expressões Gênicas

Um dos principais desafios para o tratamento de câncer é encontrar o tratamento específico para cada tipo distinto de câncer, a fim de maximizar a eficiência do método e minimizar o excesso de medicação indevida ou desnecessária ao paciente. Sendo assim, a classificação do tipo de câncer é essencial para o sucesso do tratamento. Neste capítulo será abordado o problema de classificação de tipos de Câncer: Leucemia Linfoblástica Aguda e Leucemia Mieloblástica Aguda, por meio de expressões gênicas obtidas por técnicas de *Microarray*. A técnica de análise por expressão gênica fornece um grande número de variáveis, o que torna sua análise complexa e lenta. Com intuito de contornar esse inconveniente, serão aplicadas técnicas de seleção de características, visando diminuir, substancialmente, o número de genes, aliadas ao método apresentado no capítulo 3, a fim de selecionar as instâncias ou pacientes mais significativos por classe.

5.1 O problema de classificação do tipo de leucemia

De acordo com o Instituto Nacional do Câncer [1], leucemia é um tipo de câncer que se inicia na formação das células sanguíneas (células imaturas), como o da medula óssea, e com grande impacto nas células saudáveis a serem produzidas. Os tipos diferentes de leucemia são agrupados de acordo com a velocidade a qual a doença se desenvolve (crônica ou aguda) e, também, categorizada de acordo com o tipo de célula que são afetadas: Linfoide ou Mieloide. A leucemia se origina nas células blastócitos da medula óssea (células B), células timo (células T) e gânglios linfáticos. Leucemia Linfoblástica Aguda ocorre, predominantemente, em crianças até 4 anos, porém pode afetar, também, adultos. ALL é categorizada em três grupos

[1] http://www.cancer.gov/cancertopics/types/leukemia

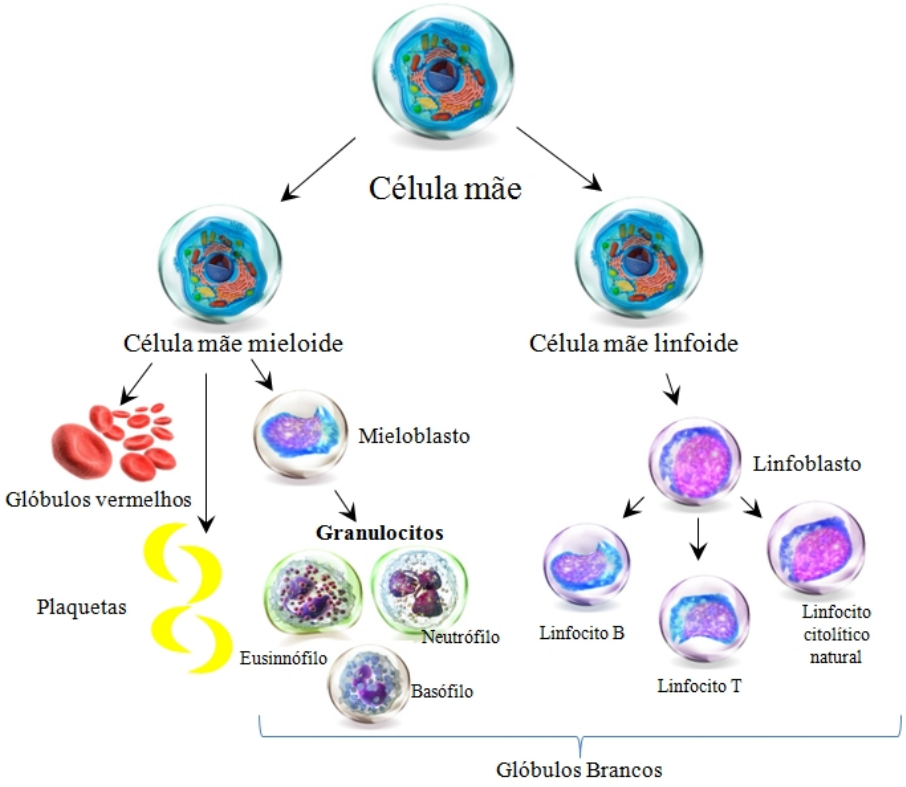


Figura 5.1: Evolução de uma célula sanguínea. Figura inspirada na original disponível no Instituto Nacional do Câncer.

principais que será abordado posteriormente.

De acordo com o Instituto Nacional do Câncer, em uma pessoa saudável, a medula óssea desenvolve células sanguíneas mães, conhecidas como células imaturas e, posteriormente, se tornam células maduras. Essas células, posteriormente, se transformam em células mieloblásticas (mieloides) ou linfoblásticas (linfoides). As células mieloides se transformam em uma dos três tipos de glóbulos sanguíneos maduros: Glóbulos vermelhos que transportam oxigênio, Plaquetas que formam coágulos e Glóbulos brancos que defendem nosso sistema contra infecções. Por outro lado, uma célula linfoide se transforma em um dos três tipos de linfócitos: Linfócitos B, que produzem anticorpos, Linfócitos T que auxiliam os Linfócitos B a produzir anticorpos e Linfócitos citolítico naturais que lutam contra as células cancerosas e vírus. A Figura 5.1 apresenta a evolução de uma célula sanguínea.

O tipo aguda é a doença na qual os leucócitos apresentam uma aparência anormal das células na medula óssea, sangue periférico e, frequentemente, no fígado, baço, nódulos linfáticos e outros órgãos parenquimatosos (Mughal et al., 2006). Na leucemia aguda, as células sanguíneas não carregam suas funções normais e um número anormal de células aumentam rapidamente. Existem dois tipos comuns de leucemia aguda: Leucemia Linfoide Aguda ou Leucemia Linfoblástica Aguda (LLA) e Leucemia Mieloide Aguda ou Leucemia Mieloblástica

Aguda (LMA).

Leucemia Linfoblástica Aguda é uma doença maligna causada pelo crescimento anormal das células brancas na medula óssea. De acordo com associação Francês-Americano-Britânico de classificação morfológica de Leucemia, baseados nas análises morfológicas e citoquímicas das células maligas (Vardiman et al., 2009): apresenta, na sua maioria, linfoblastos pequenos com núcleo regular, representa cerca de 25% a 30% dos casos (L1), em sua maioria são lifoblastos médios e grandes, com núcleo irregular, representa cerca de 70% dos casos (L2) e (L3) com presença de linfoblastos de tamanho médio a grande e núcleos bem irregulares, representa cerca de 1% a 3% dos casos.

A Leucemia Mieloblástica Aguda é um tipo de câncer presente no sangue e na medula óssea, e é mais comum sua formação em adultos, homens e com idade a partir de 65 anos. Devido aos seus avanços no tratamento, LMA tem tido resposta de retrocesso da doença em aproximadamente 60% até 70% dos adultos que recebem o tratamento correto. LMA é caracterizada em sete grupos principais, de acordo com Associação Francês-Americano-Britânico de classificação morfológica de Leucemia, baseados em análises morfológicas e citoquímicas das células maligas: Leucemia mieloblástica indiferenciada (M0), Leucemia mieloblástica pouco diferenciada (M1), Leucemia mieloblástica diferenciada (M2), Leucemia promielocítica (M3), Leucemia mielomonocítica (M4), Leucemia monocítica (M5), Leucemia eritroleucemia (M6) e Leucemia megacarioblástica (M7).

Ambos os tipos de leucemia aguda (LLA e LMA) parecem ser idênticas sobre uma avaliação microscópica e, devido a isso, por muitos anos, foram consideradas como uma única doença. No entanto, elas são, geneticamente, muito distintas entre si e podem causar diferenças clínicas, significativamente, diferentes e respostas diferentes a tipos de tratamento. Nesse sentido, LLA e LMA devem ser distinguidas o quanto antes para definição do correto tratamento. Embora a distinção entre LLA e LMA possam ser, geneticamente, bem distintas, nenhum teste único é atualmente suficiente para estabelecer a diagnose (Look, 1997; Rowley., 2000).

O uso de análise de expressão gênica por *microarray* representa uma abordagem alternativa eficiente para diferenciar entre os tipos de Leucemia LLA e LMA, e tornam possível a assinatura da expressão gênica associada com os subtipos de leucemia (Wouters et al., 2009). Devido aos testes para análise de expressão gênica que podem ser realizadas em paralelo, elas têm sido propostas como um teste robusto para diagnose. A análise de expressão gênica por meio de DNA *microarray* é, geralmente, caracterizada por possuir um grande número de variáveis (genes) e poucas amostras (indivíduos ou experimentos). Esse aspecto é um grande desafio para modelar os dados. Esse problema conduz a uma procura de métodos eficientes para a seleção de genes para o desenvolvimento de métodos de baixo custo e rápidos para testes de diagnósticos.

5.2 Aplicação de Aprendizado de Múltiplas Instâncias para o problema de classificação de leucemia

Como apresentado no capítulo 3, a metodologia proposta MILKDE mostrou-se eficiente para selecionar as instâncias mais representativas em um *bag* (conjunto de instâncias). Expandindo os conceitos de MILKDE, apresentados anteriormente, para problemas relacionados à bioinformática, pode-se adaptar a metodologia, por exemplo, com o objetivo de identificar em um grupo de pacientes aqueles que melhores representam cada classe patológica. Com a escolha de pacientes mais representativos, teremos redução do número de pacientes de treinamento, diminuindo o tempo de modelamento e os possíveis ruídos entre pacientes.

Para realizar este estudo, sobre a ótica do paradigma de Aprendizado de Múltiplas Instâncias, uma nova formulação deve ser feita: um *bag* B_i será definido como um conjunto de pacientes, a instância B_{ij} definida como um paciente pertencente ao *bag* B_i e, finalmente, a característica B_{ijk} será definida como os genes pertencentes ao pacienteB_{ij}. Dessa forma, um *bag* formado por pacientes com Mieloide será representado por B_i^M e um *bag* formado por pacientes com Linfoide representado por B_i^L.

O grande desafio para essa adaptação é como será feita a criação de cada *bag* (ou grupo de pacientes) B_i^M ou B_i^L. Para avaliação da metodologia, foi escolhida a base de dados disponibilizada por Golub et al. (1999), a qual contém pacientes com 02 tipos de Leucemia: Linfoblástica e Mieloblástica.

Para a montagem dos *bags*, foram criados grupos formados com sete[2] pacientes. A formação de cada *bag*, seguiu o procedimento a seguir: Foi tomado o primeiro paciente e mais 6 outros pacientes aleatórios do mesmo tipo de Leucemia, sem repetição para cada *bag*. Esse mesmo processo foi realizado para os demais pacientes de toda base. Dessa forma, para os dados de treinamento coposto por 27 casos de ALL e 11 LMA, foram criados 27 grupos (*bags*) de ALL, contendo 7 pacientes distintos (com ALL) em cada *bag* e 11 grupos de LMA de 7 pacientes distintos (com LMA) em cada *bag*.

Após a formação dos *bags*, foi encontrado o paciente que melhor representa a classe em cada *bag*. O cálculo foi realizado por meio das probabilidades via Estimador de Densidade por *kernel*, como descrito no capítulo 3. Após a eleição do melhor paciente em cada *bag*, foram verificados que alguns pacientes não foram eleitos em nenhum *bag* e outros foram eleitos em mais de um *bag*. Sendo assim, após a exclusão dos não eleitos e dos repetidos, houve uma redução do número de pacientes em relação ao número original.

Observemos o seguinte exemplo apresentado na Figura 5.2: originalmente, cada classe (ALL e LMA) possui 4 pacientes. Após a adaptação, foram criados 4 *bag* e cada um ficou formado por 3 pacientes aleatórios da mesma classe e distintos entre si. Após o cálculo das probabilidades, os pacientes que possuem a maior probabilidade de pertencer àquela classe

[2]Para a base em análise, foram testadas diversas formações de *bags*, sendo 7 (1+6) a formação que apresentou o melhor resultado, por meio de validação cruzada com o conjunto de treinamento.

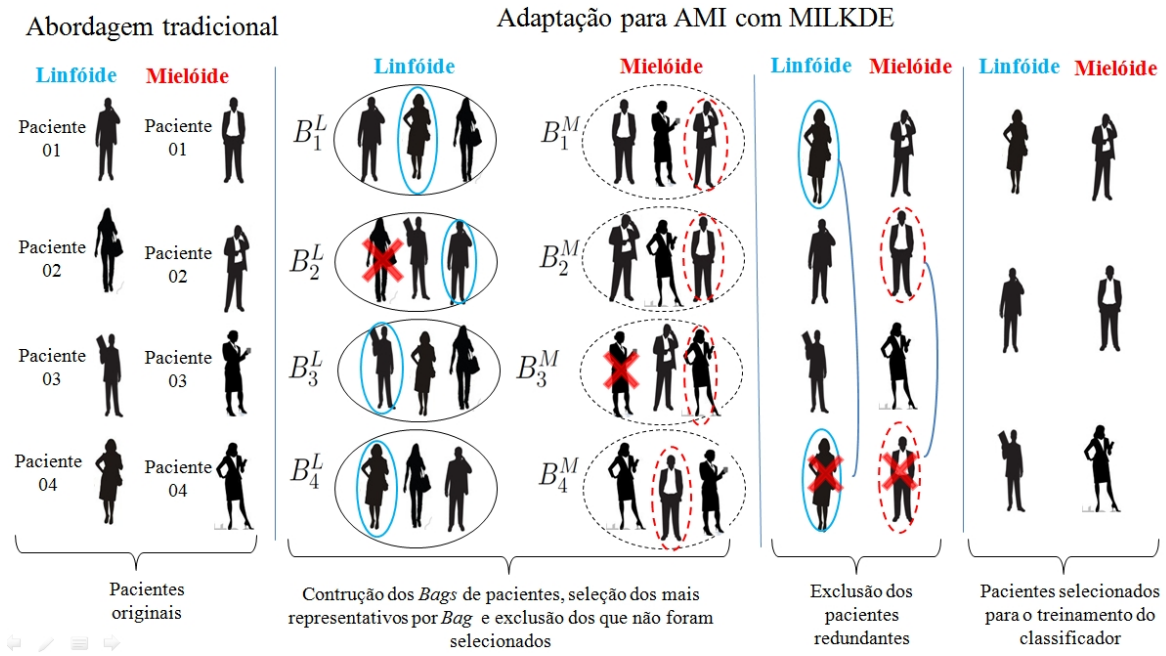


Figura 5.2: Adaptação do problema de Leucemia para abordagem MILKDE.

foram escolhidos como representante desta (pacientes circulados em azul). Nessa mesma fase é observado que um paciente de cada classe não foi escolhido em nenhum *bag* (pacientes excluídos com um × em vermelho), aparentemente por não ser um paciente forte para discriminar aquela classe de Leucemia. Observa-se, ainda, que o primeiro e quarto *bag*, para Linfoide, e segundo e quarto *bag*, para Mieloide, tiveram o mesmo paciente escolhido, sendo assim um deles foi excluído reduzindo o número de pacientes originais. Esse grupo final de pacientes, separados por classe, é, então, destinado para o treinamento do classificador.

5.3 Experimento

Um dos primeiros trabalhos para realizar análise de expressões gênicas, visando identificação de leucemia, foi o trabalho publicado por Golub (Golub et al., 1999). Em seu trabalho ele propôs uma técnica chamada de *Neighborhood Analysis*, a qual foi possível identificar uma lista de 50 genes principais, fortemente relacionado com a diferenciação entre ALL e AML. Diversos outros trabalhos (Smith et al., 2000; Lemos e Baranaukas, 2006; Nguyen e Rocke, 2001; Slonim et al., 2000; Gamberger et al., 2004; Zararsiz et al., 2012; Mishra e Sahu, 2011) após o de Golub et al. (1999) trabalharam com análise de expressão gênica, utilizando o mesmo conjunto de dados.

Para realizar o experimento, foi utilizado a base de dados de Leucemia disponibilizada por (Golub et al., 1999)[3], a qual contém casos de Leucemia Linfoboblástica Aguda e Leucemia

[3]http://www.broadinstitute.org

Mieloblástica Aguda. A base de dados contém um total de amostras de 72 casos (pacientes) e 7.129 variáveis (nível de expressões gênicas). A base é dividida em dois subconjuntos: Treinamento, composta por 38 casos (27 casos de ALL e 11 de AML) e a base de Teste composta por 34 casos (20 casos de ALL e 14 casos de AML).

Devido à grande dimensionalidade de características para a base em estudo, 7.129 genes, foi decidido utilizar, inicialmente, um método para seleção de características. Esses métodos buscam empregar ferramentas estatísticas, a fim de extrair os dados mais relevantes em um conjunto de dados, dessa forma encontrando as características que melhor representam uma classe, eliminando dados equivalentes e outros que não acrescentam na separabilidade de classe. Para este trabalho, foi utilizado o método de Gini (Gini, 1912; Shlomo e Edna, 2013) para redução da dimensionalidade. Este método foi escolhido devido sua boa performance para a mesma base testada, quando comparado com outros métodos Faria et al. (2015).

O coeficiente de Gini (Gini, 1912; Shlomo e Edna, 2013) é uma métrica utilizada para medir uma dispersão estatística de uma distribuição, desenvolvida por Corrado Gini. Esse coeficiente é utilizado como uma medida de desigualdade de distribuição, originalmente para medir distribuição de renda. Entretanto, essa medida pode ser aplicada para qualquer distribuição de dados. Quando encontramos um baixo coeficiente de Gini, este indica uma distribuição mais coesa. Sendo assim, um coeficiente de valor 0 indica uma completa igualdade da distribuição, em contrapartida um coeficiente se aproximando de 1 indica uma distribuição completamente desigual.

Para os experimentos, inicialmente foi aplicado do coeficiente de Gini para redução de dimensionalidade, e selecionados os 50 primeiros genes para seguir os trabalhos. A quantidade de genes escolhidos (50) foi o mesmo utilizado por Golub et al. (1999) para efeitos de comparação. Após a redução da dimensionalidade foram realizados 02 experimentos:

1. Empregando os 50 primeiros genes escolhidos pelo método de Gini. Para o treinamento, foram utilizados os 38 pacientes (27 ALL e 11 AML) e entregues ao algoritmo Logitboost como classificador;

2. Empregando os 50 genes escolhidos por Gini, porém utilizando a metodologia MILKDE para seleção de instâncias, foram utilizados, somente, 19 pacientes (10 ALL e 9 AML). Novamente, foi aplicado o algoritmo LogitBoost como classificador.

5.3.1 Resultados

A Tabela 5.3 apresenta os 50 genes selecionados utilizando o método Gini com suas respectivas descrições funcionais. Entre eles, 21 genes estão presentes, também, no trabalho de Golub et al. (1999) (genes em negrito na primeira coluna da Tabela 5.3). Além dos 21 genes escolhidos que também estão no trabalho de Golub et al. (1999), quatro genes merecem destaque: ADM

Adrenomedullin, CYSTATIN A, Ahnak-related Sequence e SNRPN. Esses genes, selecionados pelo método Gini, possuem relação direta com o processo da geração de tumores.

Na Tabela 5.1 são apresentados os resultados para os dois experimentos conduzidos: no primeiro utilizando os 38 pacientes para treinamento e o segundo experimento, utilizando 19 pacientes para treinamento, escolhidos pela metodologia MILKDE para seleção de instâncias. Ambos com os 50 genes escolhidos pelo método Gini e utilizando o algoritmo LogitBoost como classificador. Para a classe ALL, em todos os testes realizados os métodos propostos foram capazes de prever corretamente os 20 casos.

Para o segundo experimento (utilizando MILKDE) é apresentado na Tabela 5.1, a acurácia média e o desvio padrão. O desvio padrão é apresentado para este caso devido à característica do método em selecionar de forma aleatória os pacientes para a formação dos *bags*. Assim, foram apresentados a média e o desvio padrão para 10 diferentes avaliações.

Tabela 5.1: Resultados para a metodologia proposta. Base de teste - 34 pacientes (20 ALL e 14 AML). Para o primeiro teste é apresentado a Acurácia (%) e para o segundo a Acurácia média (%) e o desvio padrão (%).

Configuração do treinamento	Total (%)	ALL (%)	AML (%)
38 pacientes / 50 genes	94,1	100,0	85,7
19 pacientes / 50 genes	95,9 $\pm$ 1,55	100,0 $\pm$ 0,0	91,5 $\pm$ 3,0

Os resultados obtidos com o emprego de Aprendizado de Múltiplas Instâncias, para redução do número de pacientes de treinamento (50%), se mostrou eficiente obtendo resultados ligeiramente melhores que quando treinado com os 38 pacientes originais. Esse indício corrobora a habilidade da metodologia proposta (MILKDE) em identificar instâncias que melhor representaram cada classe. Uma hipótese válida, para a melhora dos resultados ou resultados equivalentes, utilizando a metade dos pacientes do treinamento inicial, pode ocorrer devido à exclusão de pacientes, que, embora classificados como ALL ou AML, não representam bem cada classe. Os resultados encontrados são equiparados com alguns dos melhores resultados encontrados na literatura, conforme apresentado na Tabela 5.2.

Tabela 5.2: Resultados da literatura. Base de teste - 34 pacientes (20 ALL e 14 AML)

Método	Configuração do treinamento	Acurácia (%)
(Golub et al., 1999)	38 pacientes / 50 genes	93,94
(Nguyen e Rocke, 2002)	38 pacientes / 50 genes	91,18
(Tong et al., 2009)	38 pacientes / 48 genes	97,06
(Zararsiz et al., 2012)B.SVM	38 pacientes / 37 genes	92,50
(Zararsiz et al., 2012)S.SVM	38 pacientes / 37 genes	87,50
(Olshen e Jain, 2002)	38 pacientes / 100 genes	94,00

Tabela 5.3: 50 Genes mais relevantes, selecionados pelo método de *Gini*. Os genes em negritos estão também presentes no trabalho de Golub et al. (1999).

Genes Selecionados por *Gini*	Descrição
D14874	*ADM Adrenomedullin*
D26308	*NADPH-flavin reductase*
D49950	*Liver mRNA for interferon-gamma inducing factor(IGIF)*
D88422	*CYSTATIN A*
HG4321	*Ahnak-Related Sequence*
J04615	*SNRPN Small nuclear ribonucleoprotein polypeptide N*
J04990	*CATHEPSIN G PRECURSOR*
J05243	*SPTAN1 Spectrin, alpha, non-erythrocytic 1 (alpha-fodrin)*
L08177	*CMKBR7 Chemokine (C-C) receptor 7*
L47738	*Inducible protein mRNA*
M11722	*Terminal transferase mRNA*
M16038	*LYN V-yes-1 Yamaguchi sarcoma viral related oncogene homolog*
M21551	*Neuromedin B mRNA*
M23197	*CD33 antigen (differentiation antigen)*
M27891	*CST3 Cystatin C (amyloid angiopathy and cerebral hemorrhage)*
M31166	*PTX3 Pentaxin-related gene, rapidly induced by IL-1 beta*
M31303	*Oncoprotein 18 (Op18) gene*
M54995	*PPBP Connective tissue activation peptide III*
M55150	*FAH Fumarylacetoacetate*
M77142	*NUCLEOLYSIN TIA-1*
M80254	*PEPTIDYL-PROLYL CIS-TRANS ISOMERASE, MITOCHONDRIAL PRECURSOR*
M81933	*CDC25A Cell division cycle 25A*
M84526	*DF D component of complement (adipsin)*
M91432	*ACADM Acyl-Coenzyme A dehydrogenase, C-4 to C-12 straight chain*
M92287	*CCND3 Cyclin D3*
M96326	*Azurocidin gene*
U02020	*Pre-B cell enhancing factor (PBEF) mRNA*
U12471	*Thrombospondin-p50 gene extracted from Human thrombospondin-1 gene, partial cds*
U46499	*GLUTATHIONE S-TRANSFERASE, MICROSOMAL*
U46751	*Phosphotyrosine independent ligand p62 for the Lck SH2 domain mRNA*
U50136	*Leukotriene C4 synthase (LTC4S) gene*
X04085	*Catalase (EC 1.11.1.6) 5'flank and exon 1 mapping to chromosome 11*
X59417	*PROTEASOME IOTA CHAIN*
X62320	*GRN Granulin*
X62654	*ME491 gene extracted from H.sapiens gene for Me491/CD63 antigen*
X70297	*CHRNA7 Cholinergic receptor, nicotinic, alpha polypeptide 7*
X74262	*RETINOBLASTOMA BINDING PROTEIN P48*
X87613	*Skeletal muscle abundant protein*
X90858	*Uridine phosphorylase*
X95735	*Zyxin*
Y12670	*LEPR Leptin receptor*
U22376	*C-myb gene extracted from Human (c-myb) gene, complete primary cds*
L09209	*APLP2 Amyloid beta (A4) precursor-like protein 2*
U37055	*Hepatocyte growth factor-like protein gene*
M12959	*TCRA T cell receptor alpha-chain*
M27783	*ELA2 Elastatse 2, neutrophil*
M83652	*PFC Properdin P factor, complement*
M98399	*CD36 antigen (collagen type I receptor, thrombospondin receptor)*
X14008	*Lysozyme gene (EC 3.2.1.17)*
M31523	*TCF3 Transcription factor 3 (E2A immunoglobulin enhancer binding factors E12/E47)*

5.4 Conclusões do capítulo

Neste capítulo foi introduzido o problema de classificação de tipo de leucemia entre Mieloblástica e Linfoblástica. Inicialmente, foi utilizado o método de Gini para selecionar os 50 genes mais importantes dentre os 7.129 Genes disponíveis. Após a seleção dos Genes, foi utilizada a metodologia MILKDE proposta para selecionar as instâncias (pacientes) que melhor representavam cada classe (AML e ALL). Em relação aos 38 pacientes originais utilizados para treinamento, foram selecionados 19 pacientes pela metodologia proposta. Os resultados foram, ligeiramente, melhor que os encontrados quando realizado o treinamento com todos os pacientes. Esses resultados indicam que a metodologia foi capaz de reduzir a quantidade de instâncias de treinamento, sem impactar na classificação final da base de teste. Parte do trabalho descrito neste capítulo foi publicado em Faria et al. (2015).

Capítulo 6

Conclusões e Trabalhos Futuros

6.1 Conclusões

Aprendizado de Múltiplas Instâncias é uma generalização do aprendizado supervisionado tradicional, em que a rotulação na fase de treinamento é atribuída a conjuntos de instâncias, chamados de *bags*, em vez de cada instância individualmente. A ideia principal desse aprendizado consiste na forma como os rótulos são atribuídos aos *bags*: Um *bag* é rotulado como negativo, se, seguramente, é de conhecimento que todas suas instâncias são negativas e rotuladas como positivas se sabe-se ao certo que pelo menos uma instância, contida naquele *bag* é positiva.

Devido a essa característica do problema de AMI, em não se saber quais e quantas são as instâncias positivas e negativas contidas nos *bags* positivos, esse tipo de aprendizado se apresenta como um grande desafio na área de inteligência computacional.

Existem diversos algoritmos, com diferentes abordagens, na literatura visando trabalhar com problemas de Aprendizado de Múltiplas Instâncias. Entre as diferentes abordagens existentes, a Seleção de Instância tem sido a abordagem mais utilizada nos últimos anos e têm apresentados os resultados mais promissores. Nesse tipo de abordagem, o algoritmo busca identificar, entre as instâncias dos *bags* positivos, aquelas que são realmente positivas e, assim, transformar o problema de Aprendizado de Múltiplas Instâncias em uma abordagem de aprendizado supervisionado tradicional.

Nesta tese foram apresentados os conceitos de Aprendizado de Múltiplas Instâncias, suas principais abordagens, algoritmos e algumas das principais aplicações práticas para esse tipo de aprendizado.

Neste trabalho foram propostos duas novas metodologias para trabalhar com problemas de Aprendizado de Múltiplas Instâncias. As metodologias propostas foram baseados em seleção de instâncias por meio do cálculo da probabilidade, utilizando Estimador de Densidade por *Kernel* e o algoritmo LogitBoost como classificador. As abordagens propostas tiveram

como objetivo reduzir o ruído causado pelas instâncias negativas (dentro dos *bags* positivos) no momento do treinamento do classificador. Dessa forma, a metodologia seleciona, de acordo com os critérios apresentados no capítulo 3, as instâncias consideradas mais positivas dentre todas contidas nos *bags* positivos de treinamento. Assim, a metodologia proposta objetivou realizar um treinamento mais representativo para as instâncias positivas. As duas propostas apresentadas se diferenciam por conter, em uma delas, uma etapa chamada: Repescagem. Nessa etapa, as instâncias que não foram selecionadas durante a primeira etapa de Qualificação, na qual é escolhida uma única instância de cada *bag*, são ordenadas de acordo com sua proximidade da Instância mais Positiva e um percentual dessas instâncias são escolhidas para compor a classe positiva de treinamento.

A metodologia proposta foi capaz de alcançar resultados competitivos em todas as principais bases testadas, superando todos os métodos em algumas dessas bases. Quando avaliado nas cinco principais bases de dados: MUSK1, MUSK2, Elefante, Tigre e Raposa, a metodologia somente não obteve bons resultados para a base do Tigre, que apresenta um grande desvio padrão para a classe positiva.

Para a base de dados COREL, a proposta obteve resultados expressivos em quase todas as 20 bases testadas, obtendo o melhor resultado em 15, das 20 bases possíveis, além da melhor média geral. Esses resultados revelam a versatilidade e poder de generalização da metodologia proposta, não obtendo bons resultados em somente um tipo de base de dados.

Foi realizado, também, um estudo estatístico entre as metodologias propostas e aquelas da literatura que divulgaram seus desvios padrões e/ou intervalos de confiança. Para o estudo estatístico, foram utilizados os métodos: Teste de Friedman e Intervalo de Confiança. É válido ressaltar que na literatura não foi encontrada nenhuma avaliação estatística entre métodos para trabalhar com aprendizado de múltiplas instâncias.

No capítulo 5, foi apresentada uma nova aplicação para AMI e o algoritmo proposto. A metodologia foi utilizada em uma base conhecida de Leucemia, em que o objetivo é distinguir entre dois tipos de Leucemia: Linfoblástica e Mieloblástica. MILKDE foi utilizado para encontrar os pacientes que melhor representava cada classe de Leucemia. Após experimentos, foi possível reduzir pela metade os pacientes de treinamento, e encontrar resultados ligeiramente melhores dos que aqueles quando utilizado todos os pacientes da base. Esse tipo de aplicação para Aprendizado de Múltiplas Instâncias não foi encontrado na literatura e pode ser um passo importante para novas aplicações, as quais sejam interessantes filtrar as instâncias de treinamento, com intuito de remover possíveis ruídos dentre elas.

Como foi enunciado na Introdução desse trabalho, o objetivo principal dessa Tese foi desenvolver uma nova metodologia para trabalhar com os problemas de Aprendizado de Múltiplas Instâncias, que respeitasse as principais premissas de AMI e que obtivesse resultados competitivos na maioria das bases de dados avaliadas. Avaliando-se os resultados dos experimentos conduzidos no capítulo 4, constata-se que esse objetivo foi alcançado e, em algumas das bases

testadas, a metodologia proposta obteve resultados melhores que todos da literatura.

6.2 Trabalhos futuros

Durante a pesquisa e o desenvolvimento das abordagens propostas neste trabalho, foram identificados novos caminhos para uma série de estudos que podem ser desenvolvidos, com intuito de avançar os resultados das metodologias propostas e gerar novas aplicações. A seguir são listados algumas propostas de continuidade aos estudos apresentados nesta Tese:

- A metodologia proposta nesta tese focou, primariamente, em identificar instâncias positivas que melhor representassem cada *bag* positivo, nenhum trabalho foi realizado sobre os *bags* negativos. Uma extensão deste trabalho pode ser realizada estudando melhor os *bags* negativos. Dessa forma, por meio de métodos adaptativos, buscar encontrar, em determinadas bases, coerência entre as instâncias desses *bags* e, assim, selecionar as "melhores" instâncias negativas para compor o classificador;
- Nesta tese utilizou-se o algoritmo LogitBoost como classificador, novos estudos podem ser conduzidos com a utilização de novos classificadores e/ou um estudo de adaptações para estes;
- Investigar novos problemas na área de bioinformática, aos quais sejam difíceis de serem modelados por métodos de aprendizado supervisionado tradicional. Procurar identificar em bases na área médica, aplicações às quais a redução das instâncias (pacientes) possa ser vantajosa e buscar aumentar a acurácia dos métodos já aplicados até hoje;
- Pesquisar problemas práticos, na qual a aplicação do paradigma de Aprendizado de Múltiplas Instâncias, pode ser mais adequado que o Aprendizado tradicional;
- Extensão da metodologia proposta para trabalhar com detecção de falhas ou novidades. O trabalho inicial para esse problema será a formulação dos problemas de detecção de eventos como problemas de Aprendizado de Múltiplas Instâncias, em que as séries temporais que ocorram um evento seja um *bag* positivo e negativo, caso contrário.

6.3 Produção ciêntifica

Durante o curso do Doutorado foram elaborados os seguintes artigos ciêntificos:

- Faria, A.W.C. ; Ceolho, F.G.F. ; Silva, A. M.; de Almeida, G.M., Lemos, A. P. ; Braga, A. P. . "MILKDE: A New Approach for Multiple Instance Learning based on Positive Instance Selection and Kernel Density Estimation". In: Engineering Applications of Artificial Intelligence, 2016, JCR=2,207: Artigo submetido, pendente aprovação;

- Faria, A.W.C. ; Menotti, D. ; Lemos, A. P. ; Braga, A. P. . "A New Approach for Multiple Instance Learning based on a Homogeneity Bag Operator". In: ESANN - The 22th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning, 2014, Bruges. ESANN 2014., V. 1. p. 65-70, Louvain Belgique;

- Faria, A.W.C. ; Silva, A. M. ; Rodrigues, T. S. ; Costa, M. A. ; Braga, A. P. "A Ranking Approach for Probe Selection and Classification of Microarray Data With Artificial Neural Networks", In: Journal of Computational Biology, 2015, Vol 22, pp 953-961, JCR=1,737;

- Faria, A.W.C.; Silva, A. M., Bambirra, L.C.; Costa, F.J.; Leite, C.C; Sady A. S. F.; Souza A.P.; Tierra-Criollo, C.J. and Braga, A. P., "Selection and Classification Methods For Detecting Imagination Spontaneous EEG Signals: A Study of Methodologies", no XIV Congresso Brasileiro de Engenharia Biomédica - CBEB 2014, Uberlândia MG;

- Faria, A.W.C. ; Castro, C. L. ; Braga, A. P. "A New Oversampling-Based Approach for Class Imbalance Problem", Proceedings of the 1st BRICS Countries Congress (BRICS-CCI) and 11th Brazilian Congress on Computational Intelligence - BRICS-CCI & CBIC, 2013., Porto de Galinhas PE;

- Silva, A. M. ; Faria, A.W.C. ; Rodrigues, T. S. ; Costa, M. A. ; Braga, A. P. "Artificial Neural Networks and Ranking Approach for Probe Selection and Classification of Microarray Data". Proceedings of the 1st BRICS Countries Congress (BRICS-CCI) and 11th Brazilian Congress on Computational Intelligence - BRICS-CCI & CBIC, 2013., Porto de Galinhas PE;

- Lara, D. S. D. ; Faria, A.W.C. ; Araujo, A. A. ; Menotti, D. "A Novel Hybrid Method for the Segmentation of the Coronary Artery Tree in 2d Angiograms". In International Journal of Computer Science and Information Technology (Print), v. 5, p. 45-65, 2013;

- Marinho, R.Q. ; Faria, A.W.C. ; Carvalho, T.C. " Detection of knocking anomalies in an internal combustion engine with fuel flex technology through wavelet transform analysis". Congresso SAE Brasil 2016, São Paulo SP. - Artigo sumetido, pendente aprovação.

Referências Bibliográficas

A. Dehzangi, S. P.; Amnuaisuk, K. e Mohadesi, E. (2009). Protein fold prediction problem using ensemble of classifiers. In *16 th International Conference Neural Information Processing ICONIP*, pp. 503–511.

Andrews, S.; Tsochantaridis, I. e Hofmann, T. (2003). Support vector machines for multiple-instance learning. *Advances in Neural Information Processing Systems (NIPS)*, Vol. 15:p. 561–568.

Auer, P. (1997). On learning from multi-instance examples: Empirical evaluation of a theoretical approach. In *Proc. 14th International Conference on Machine Learning*, pp. 21–29. Morgan Kaufmann.

Babenko, B. (2008). Multiple instance learning: Algorithms and applications. Technical report, University of California San Diego, San Diego.

Babenko, B.; Verma, N.; Dollr, P. e Belongie, S. (2011). Multiple instance learning with manifold bags. In *International Conference on Machine Learning (ICML)*, pp. 81–88.

Bernhard, S. e J., S. A. (2001). *Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond*. MIT Press, Cambridge, MA, USA.

Blumberg, H. (1920). Hausdorff's grundzüge der mengenlehre. *Bulletin of the American Mathematical Society*, 27(3):116–129.

Boser, B. E.; Guyon, I. L. e Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. In *5th Annual Workshop on Computational Learning Theory*, volume 8, pp. 144–152.

Braga, A. P.; Carvalho, A. P. L. e Ludemir, T. B. (2011). *Redes Neurais Artificiais:Teorias e Aplicacoes*. LTC - Livros Tecnicos e Cientificos Editora Ltda, Rio de Janeiro, RJ, BR, 2nd edição.

Carbonneau, M. A.; Granger, E.; Raymond, A. J. e Gagnon, G. (2016). Robust multiple-instance learning ensembles using random subspace instance selectionl. *Pattern Recognition*, 58(1):83–99.

Carson, C.; Thomas, M.; Belongie, S.; Hellerstein, J. e Malik, J. (1999). Blobworld: A system for region-based image indexing and retrieval. In Huijsmans, D. P. e Smeulders, A. W. M., editores, *V*isual Information and Information Systems, volume 1614 of *L*ecture Notes in Computer Science, pp. 509–517. Springer.

Chen, Y.; Bi, J. e Wang, J. (2006a). Miles: Multiple-instance learning via embedded instance selection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1931 – 1947.

Chen, Y.; Bi, J. e Wang, J. (2006b). Miles: Multiple-instance learning via embedded instance selection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 28(12):p. 1931–1947.

Chen, Y. e Wang, J. Z. (2004a). Image categorization by learning and reasoning with regions. In *Journal of Machine Learning Research*, pp. 913–939.

Chen, Y. e Wang, J. Z. (2004b). Image categorization by learning and reasoning with regions. *Journal of Machine Learning Research*, Vol. 5:p. 913–939.

Cover, T. M. e Hart, P. E. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

Das, S.; Amoedo, B.; la Torre, F. D. e Hodgins, J. (2012). Detecting parkinsons symptoms in uncontrolled home environments: A multiple instance learning approach. In *Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pp. 3688–3691.

Demsar, J. (2006). Statistical comparisons of classifiers over multiple data sets. *Journal of Machine Learning Research*, Vol. 7:p. 1–30.

Dietterich, T. G.; Lathrop, R. H. e Lozano-Perez, T. (1997). Solving the multiple instance problem with axis-parallel retangles. *Artificial Intelligence*, Vol. 89(1-2):p. 31–71.

Dong, L. (2006). Phd thesis: A comparison of multi-instance learning algorithms. Technical report, Department of Computer Science, Waikato University, Hamilton, New Zealand. Thesis of Master of Science.

Duda, R.; Hart, P. e Stork, D. (2000). *Pattern Classification*. Wiley-Interscience, 2 nd. edição.

Dundar, M.; Fung, G.; Krishnapuram, B. e Rao, R. B. (2008). Multiple-instance learning algorithms for computer-aided detection. *IEEE Transactions on Biomedical Engineering*, 55(3):1015–1021.

Edgar, G. (1995). *Measure, topology, and fractal geometry*. Springer-Verlag, 3 edição.

Erdem, A. e Erdem, E. (2011). Multiple-instance learning with instance selection via dominant sets. In *Lecture Notes in Computer Science - SIMBAD 2011*, pp. 177–191.

Faria, A. W. C.; da Silva, A. M.; de Souza, R. T.; Costa, M. A. e Braga, A. P. (2015). A ranking approach for probe selection and classification of microarray data with artificial neural networks. *Journal of Computational Biology*, 22(1):953–961.

Faria, A. W. C.; Menotti, D.; Lemos, A. P. e Braga, A. P. (2014). A new approach for multiple instance learning based on a homogeneity bag operator. In *22th European Symposium on Artificial Neural Networks, Computational Intelligence and Machine Learning*, pp. 65–70.

Freund, Y. (1995). Boosting a weak learning algorithm by majority. *Information and Computation*, 121:256–285.

Friedman, J.; Hastie, T. e Tibshirani, R. (2000). Additive logistic regression: A statistical view of boosting. In Stanford University, D. o. B., editor, *A*nnals of Statistics, pp. 337–407.

Fu, Z. e A., R.-K. (2009). An instance selection approach to multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2009*, pp. 911– 918.

Fu, Z. e Kelly, A. R. (2009). An instance selection approach to multiple instance learning. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 911–918.

Fu, Z.; Robles-Kelly, A. e Zhou, J. (2011). Milis: Multiple instance learning with instance selection. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 958 – 977.

Gamberger, D.; Lavrac, N.; Zelezny, F. e Tolar, J. (2004). Induction of comprehensible models for gene expression datasets by subgroup discovery methodology. *Journal of Biomedical Informatics*, 37:269–284.

Garfield, E. (1979). *Citation Indexing – Its Theory and Application in Science, Technology, and Humanities*. John Wiley & Sons, Inc.

Gehler, P. V. e Chapelle, O. (2007). Deterministic annealing for multiple-instance learning. In Meila; M. e Shen, X., editores, *1*1 th AISTATS International Conference on Artificial Intelligence and Statistics, pp. 123–130.

Gini, C. (1912). Variability e mutabilita. *Memorie di metodologia statistica*.

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

Guo, Y. (2009). Max-margin multiple-instance learning via semidefinite programming. *Lecture Notes in Computer Science, Advances in Machine Learning*, Vol. 5828:p. 98–108.

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P. e Witten, I. H. (2009). The weka data mining software: An update. *SIGKDD Explor. Newsl.*, 11(1):10–18.

Haykin, S. (1994). *Neural Networks: A Comprehensive Foundation*. Macmillan, 2nd edição.

Haykin, S. (1998). *Neural Networks: A Comprehensive Foundation*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edição.

hua Zhou, Z. e ling Zhang, M. (2002). Neural networks for multi-instance learning. In *International Conference on Intelligent Information Technology*, pp. 1–14.

Huang, X.; Chen, S.-C. e Shyu, M.-L. (2003). An open multiple instance learning framework and its application in drug activity prediction problems. In *IEEE Symposium on Bioinformatics and Bioengineering (BIBE)*, pp. 53–59. IEEE Computer Society.

Keshtkar, F. e Inkpen, D. (2011). A pattern-based model for generating text to express emotion. In Mello, S. D.; Graesser, A.; Schuller, B. e Martin, J.-C., editores, *A*ffective Computing and Intelligent Interaction, volume 6975, pp. 11–21. Lecture Notes in Computer Science.

Kim, M. e la Torre, F. D. (2010). Gaussian processes multiple-instance learning. In *ICML International Conference on Machine Learning*, pp. 535–542. Omnipress.

Kira, K. e Rendell, L. (1992). A practical approach to feature selection. In *9th International Conference in Machine Learning*, pp. 249–256.

Krummenacher, G. e Ong, C.-S. (2013). Ellipsoidal multiple instance learning. In *International Conference on Machine Learning (ICML) 2013*, pp. 1–8.

Leistner, C.; Saffari, A. e Bischof, H. (2010). Miforests: Multiple instance learning with randomized trees. In *ECCV 10 European conference on Computer vision: Part VI*, pp. 29–42. Springer-Verlag.

Lemos, R. N. e Baranaukas, J. A. (2006). Uma avaliacao de dados de expressao genica em leucemias agudas utilizando arvores de decisao. *X Congresso Brasileiro de Informatica em Saude - CBIS - Florianopolis - SC*.

Li, C.; Lam, K. M.; Zhang, L.; Huj, C. e Zhang, S. (2012a). Mammogram microcalcification cluster detection by locating key instances in a mult-instance learning framework. In *IEEE International Conference on Signal Processing Communication and Computing (ICSPCC)*, pp. 175–179.

Li, W.; Duan, L.; Tsang, I. e Xu, D. (2012b). Batch mode adaptive multiple instance learning for computer vision tasks. In *IEEE Conference on Digital Computer Vision and Pattern Recognition (CVPR)*, pp. 2368–2375.

Li, W.-J. e Yeung, D.-Y. (2010). Mild: Multiple-instance learning via disambiguation. *IEEE Transactions on Knowledge and Data Engineerin*, Vol. 22(1):p. 76–89.

Li, Y.-F.; Kwok, J. T.; Tsang, I. W. e Zhou, Z.-H. (2009). Convex method for locating regions of interest with multi-instance learning. In Buntine, W.; Grobelnik, M.; Mladenic, D. e Shawe-Taylor, J., editores, *E*CML PKDD 2009 - Proceedings of the European Conference on Machine Learning and Knowledge Discovery in Databases, pp. 15–30. Springer Berlin Heidelberg.

Li, Z.; Geng, G.-H.; Feng, J.; y. Cheng, J.; Wen, C. e li Liang, J. (2014). Multiple instance learning based on positive instance selection and bag structure construction. *Pattern Recognition Letters*, Vol. 40:p. 19–26.

Lichman, M. (2013). Uci machine learning repository.

Liu; Zhou, C.; Shen, J.; Li, P. e Zhang, S. (2010). Video caption detection algorithm based on multiple instance learning. In *Fifth International Conference on Internet Computing For Science and Engineering.*

Locke, J. (1690). *Essay concerning Human Understanding.* The Baffet, London.

Look, A. T. (1997). Oncogenic transcription factors in the human acute leukemias. *Science*, 278:1059–1064.

Mahnot, A. e Popescu, M. (2012). Fumil fuzzy multiple instance learning for early illness recognition in older adults. In *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*, pp. 1–5. IEEE Computer Society.

Mangasarian, O. L. e Wild, E. W. (2008). Multiple instance classification via successive linear programming. *Journal of Optimization Theory and Applications*, Vol. 137:p. 555–568.

Maron, O. (1998). *Learning from Ambiguity.* PhD thesis, Dept. of Electrical and Computer Science, Massachusetts Inst. of Technology, Cambridge, USA.

Maron, O. e Lozano-Perez, T. (1998). A framework for multiple-instance learning. In Jordan, M.; Keams, M. e Solla, S., editores, *A*dvances In Neural Information Processing Systems (NIPS), pp. 570–576. MIT Press Cambridge, MA, USA.

Maron, O. e Ratan, A. (1998). Multiple-instance learning for natural scene classification. In *International Conference on Machine Learning (ICML)*, pp. 341–349.

Mishra, D. e Sahu, B. (2011). Feature selection for cancer classification: A signal-to-noise ratio approachl. In *International Journal of Scientific and Engineering Research IJSER*, volume 2, pp. 1–7.

Mughal, T.; Goldman, J. e Mughal, S. (2006). Understanding leukemias, lymphomas, and myelomas. Technical Report 1, Leukaemia Foundation.

Nguyen, D. V. e Rocke, D. M. (2001). Classification of acute leukemia based on dna microarray gene expressions using partial least squares. *Kluwer Academic*.

Nguyen, D. V. e Rocke, D. M. (2002). Tumor classification by partial least squares using microarray gene expression datal. In *Bioinformatics*, volume 18, pp. 39–50.

Olshen, A. B. e Jain, A. N. (2002). Deriving quantitative conclusions from microarray expression data. In *Bioinformatics*, volume 18, pp. 961–970.

Parsons, S. e Bigham, J. (1996). Possibility theory and the generalised noisy or model. In *in Proceedings of the Sixth International Conference on Information Processing and the Management of Uncertainty*, pp. 853–858.

Parzen, E. (1962). On estimation of a probability density function and mode. *JSTOR: The Annals of Mathematical Statistics*, pp. 1065–1076.

Pavan, M. e Pelillo, M. (2007). Dominant sets and pairwise clustering. *IEEE Transactions on Pattern Analysis and and Machine Inteligence*, 29(1):167–172.

Price, K.; Storn, R. M. e Lampinen, J. A. (2005). *Differential Evolution: A Practical Approach to Global Optimization (Natural Computing Series)*, volume 1. Springer-Verlag New York, Inc, 1 edição.

Rowley., J. D. (2000). Molecular genetics in acute leukemia. *Leukemia*, 14:513–517.

Ruffo, G. (2000). Phd thesis: Learning single and multiple instance decision tree for computer security applications. Technical report, Department of Computer Science, University of Turin, Torino, Italy. PhD Thesis.

S., R. e M., C. (2005). Supervised learning versus multiple instance learning: and empirical comparison. In *International Conference on Machine Learning*, pp. 697–704.

Shlomo, Y. e Edna, S. (2013). More than a dozen alternative ways of spelling gini. *Series in Statistics*, 272:11–31.

Slonim, D. K.; Tamayo, P.; Mesirov, J. P.; Golub, T. R. e Lander, E. S. (2000). Class prediction and discovery using gene expression data. In *RECOMB '00: Proceedings of the fourth annual international conference on Computational molecular biology*, pp. 263–272, New York, NY, USA. ACM.

Smith, A.; Satagopan, J.; Gonen, M. e Begg, C. B. (2000). Exploring class prediction for leukemia gene expression data. *CAMDA*.

Sun, L.; Lu, Y.; Yang, K. e Li, S. (2012). Ecg analysis using multiple instance learning for myocardial infarction detection. *IEEE Transactions on Biomedical Engineering*, Vol. 59(12):p. 3348–3356.

Thomas, G.; Peter, A. e Kowalczyk, A. (2002). Multi instance kernels. In *International Conference on Machine Learning, 2002*, pp. 179–186.

Tong, D.; Phalp, K.; A, A. S. e Mintram, R. (2009). Innovative hybridisation of genetic algorithms and neural networks in detecting marker genes for leukaemia cancer. In *4th IAPR International Conference on Pattern Recognition in Bioinformatics (PRIB)*, pp. 1–7.

Vardiman, J. W.; Thiele, J.; Arber, D. A.; Brunning, R. D.; Borowitz, M. J.; Porwit, A.; Harris, N. L.; Le Beau, M. M.; Hellstrom-Lindberg, E.; Teffer, A. e Bloomfield, C. D. (2009). The 2008 revision of the world health organization (who) classification of myeloid neoplasms and acute leukemia: rationale and important changes. *Blood*, 114:937–951.

Wang, J.; Cai, L.; Peng, J. e Jia, Y. (2015). A novel multiple instance learning method based on extreme learning machine. In *Computational Intelligence and Neuroscience*, volume 2015, pp. 1–6.

Wang, J. e Zucker, J.-D. (2000). Solving the multiple instance problem: A lazy learning approach. In Langley, P., editor, *1*7th International Conference on Machine Learning (ICML), pp. 1119–1126. Morgan Kaufmann.

Wang, S.; McKenna, M.; Nguyen, T.; Burns, J. E.; Petrick, N. e Sahiner, B. (2012). Seeing is believing: Video classification for computed tomographic colonography using multiple-instance learning. *IEEE Transactions on Medical Imaging*, 31(5):1141–1153.

Wei, H. e Yu, W. (2009). Text representation and classification based on multi-instance learning. In Hua, L. e Yu-Hong, Y., editores, *I*nternational Conference on Management Science and Engineering (ICMSE), pp. 34–39. IEEE Computer Society.

Wouters, B. J.; Lowenberg, B. e Delwel, R. (2009). A decade of genomewide gene expression profiling in acute myeloid leukemia: flashback and prospects. *Blood*, 113:291–298.

Woznica, A. e Kalousis, A. (2010). Adaptive distances on sets of vectors. In *ICDM -IEEE International Conference on Data Mining series*, pp. 579–588. IEEE Computer Society.

Xiao, Y.; Liu, B.; Cao, L.; Yin, J. e Wu, X. (2010a). Smile: A similarity-based approach for multiple instance learning. In *IEEE 10th International Conference on Publication Data Mining (ICDM)*, pp. 589–598. IEEE Computer Society.

Xiao, Y.; Liu, B.; Cao, L.; Yin, J. e Wu, X. (2010b). Smile: A similarity-based approach for multiple instance learning. In *2010 IEEE 10th International Conference on Publication Data Mining (ICDM)*, pp. 589 – 598.

Xiao, Y.; Liu, B.; Hao, Z. e Cao, L. (2014). Smile: A similarity-based approach for multiple instance learning. *IEEE Transactions on Cybernetics*, Vol. 44:p. 500–515.

Xu, Y.; Ping, W. e Campbell, A. T. (2011). Multi-instance metric learning. In *2011 11th IEEE International Conference on Data Mining*, pp. 874–883. IEEE Computer Society.

Xu, Y.; Zhu, J.-Y.; Chang, E. e Tu, Z. (2012). Multiple clustered instance learning for histopathology cancer image classification, segmentation and clustering. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 964–971. IEEE Press.

Yang, J. (2005). Review of multi-instance learning and its applications. Technical report, School of Computer Science, Carnegie Mellon University, Pittsburgh, USA.

Yang, J. (2008). MILL: A multiple instance learning library. `http://www.cs.cmu.edu/~juny/MILL`. last visit on March 2016.

Yao, M.; Yi, W.; Zhu, R. e Cheng, R. (2010). Semantic image retrieval based on multiple instance learning. In *IEEE International Conference on Data Mining Workshops*.

Zafra, A.; Pechenizkiy, M. e Ventura, S. (2010). Feature selection is the relief for multiple instance learning. In *10th International Conference on Intelligent Systems Design and Applications (ISDA)*, pp. 1–8.

Zararsiz, G.; Elmalia, F. e Ozturk, A. (2012). Bagging support vector machines for leukemia classificationl. In *International Journal of Computer Science Issues IJCSI*, volume 9, pp. 355–358.

Zh Zhou, Z.-H.; Sun; Yu-Yin; Li e Yu-Feng (2009). Multi-instance learning by treating instances as non-i.i.d. samples. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada*, pp. 1249– 1256.

Zhang, C.; Chen, X.; ching Cheng, M. C. S. e Shyu, M. L. (2005). A multiple instance learning approach for content based image retrieval using one class suport vector machine. In *IEEE International Conference on Multimedia and Expo*, pp. 1142–1145.

Zhang, D.; He, J.; Si, L. e Lawrence, R. D. (2013). Mileage: Multiple instance learning with global embedding. In Dasgupta, S. e Mcallester, D., editores, *P*roceedings of the 30th International Conference on Machine Learning (ICML), pp. 82–90. JMLR.

Zhang, M. L. e Zhou, Z. H. (2006). Adpting rbf neural networks to multi instance learning. In *Neural Processing Letters*, pp. 1– 26.

Zhang, Q. e Goldman, S. A. (2001). Em-dd: An improved multiple-instance learning technique. *Advances in Neural Information Processing Systems (NIPS)*, Vol. 28:p. 1073–1080.

Zhao, H.; Cheng, J.; Jiang, J. e Tao, D. (2013). Multiple instance learning via distance metric optimization. In *20th IEEE International Conference on Digital Image Processing*, pp. 2617–2621.

Zhou, Z. e Xu, j. (2007). On the relation between multi instance learning and semi supervised learning. In *International Conference on Machine Learning, 2007*, pp. 1167–1174.

Zucker, J.-D. e Chevaleyre, Y. (2001). Solving multiple-instance and multiple-part learning problems with decision trees and decision rules. application to the mutagenesis problem. *Lecture Notes in Artificial Intelligence*, Vol. 2056:p. 204–214.