Detecção de anomalias utilizando métodos paramétricos e múltiplos classificadores

Gabriel de Barros Paranhos da Costa

Detecção de anomalias utilizando métodos paramétricos e múltiplos classificadores

Gabriel de Barros Paranhos da Costa

Orientador: **Prof. Dr. Moacir Pereira Ponti Junior**

Dissertação apresentada ao Instituto de Ciências Matemáticas e de Computação - ICMC-USP, como parte dos requisitos para obtenção do título de Mestre em Ciências - Ciências de Computação e Matemática Computacional. *VERSÃO REVISADA*

USP – São Carlos
Outubro de 2014

Ficha catalográfica elaborada pela Biblioteca Prof. Achille Bassi
e Seção Técnica de Informática, ICMC/USP,
com os dados fornecidos pelo(a) autor(a)

Costa, Gabriel B. P.
C837d Detecção de anomalias utilizando métodos paramétricos e múltiplos classificadores / Gabriel B. P. Costa; orientador Moacir P. Ponti-Jr. -- São Carlos, 2014.
60 p.

Dissertação (Mestrado - Programa de Pós-Graduação em Ciências de Computação e Matemática Computacional) -- Instituto de Ciências Matemáticas e de Computação, Universidade de São Paulo, 2014.

1. Detecção de anomalias. 2. Reconhecimento de padrões. 3. Aprendizado de máquina. I. Ponti-Jr, Moacir P., orient. II. Título.

Abstract

Anomalies or outliers are examples or group of examples that have a behaviour different from the expected. These examples may represent diseases in individuals or populations, as well as other events such as fraud and failures in banking systems. Several existing techniques seek to identify these anomalies, including adaptations of classification methods, statistical methods and methods based on information theory. The main challenges are that the number of samples of each class is unbalanced, the cases when anomalies are disguised among normal samples and the definition of normal behaviour associated with the formalization of a model for this behaviour. In this dissertation, we propose the use of a new space to help with the detection task, this space is called *parameter space*. We also present a new framework to perform anomaly detection by using the fusion of convex hulls in multiple parameter spaces to perform the detection. The method is considered a framework because it is possible to choose which parameter spaces will be used by the method according to the behaviour of the target dataset. For the experiments, two parameter spaces were used (mean and standard deviation; mean, variance, skewness and kurtosis) and the results were compared to some commonly used anomaly detection methods. The results achieved were comparable or better than those obtained by the other methods. Furthermore, we believe that a parameter space created great flexibility for the proposed method, since it allowed the user to choose a parameter space that best models the application. Both the flexibility and extensibility provided by the use of parameter spaces, together with the good performance achieved by the proposed method in the experiments, make parameter spaces and, more specifically, the proposed methods appealing when solving anomaly detection problems.

Resumo

Anomalias ou *outliers* são exemplos ou grupo de exemplos que apresentam comportamento diferente do esperado. Na prática, esses exemplos podem representar doenças em um indivíduo ou em uma população, além de outros eventos como fraudes em operações bancárias e falhas em sistemas. Diversas técnicas existentes buscam identificar essas anomalias, incluindo adaptações de métodos de classificação e métodos estatísticos. Os principais desafios são o desbalanceamento do número de exemplos em cada uma das classes e a definição do comportamento normal associada à formalização de um modelo para esse comportamento. Nesta dissertação propõe-se a utilização de um novo espaço para realizar a detecção, esse espaço é chamado espaço de parâmetros. Um espaço de parâmetros é criado utilizando parâmetros estimados a partir da concatenação (encadeamento) de dois exemplos. Apresenta-se, então, um novo *framework* para realizar a detecção de anomalias através da fusão de detectores que utilizam fechos convexos em múltiplos espaços de parâmetros para realizar a detecção. O método é considerado um *framework* pois é possível escolher quais os espaços de parâmetros que serão utilizados pelo método de acordo com o comportamento da base de dados alvo. Nesse trabalho utilizou-se, para experimentos, dois conjuntos de parâmetros (média e desvio padrão; média, variância, obliquidade e curtose) e os resultados obtidos foram comparados com alguns métodos comumente utilizados para detecção de anomalias. Os resultados atingidos foram comparáveis ou melhores aos obtidos pelos demais métodos. Além disso, acredita-se que a utilização de espaços de parâmetros cria uma grande flexibilidade do método proposto, já que o usuário pode escolher um espaço de parâmetros que se adeque a sua aplicação. Tanto a flexibilidade quanto a extensibilidade disponibilizada pelo espaço de parâmetros, em conjunto com o bom desempenho do método proposto nos experimentos realizados, tornam atrativa a utilização de espaços de parâmetros e, mais especificamente, dos métodos apresentados na solução de problemas de detecção de anomalias.

Agradecimentos

À Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP) e à Coordenação de Aperfeiçoamento de pessoal de Nível Superior (CAPES) pela concessão de bolsa de mestrado e pelo apoio financeiro que possibilitou a realização deste trabalho.

Ao Instituto de Ciências Matemáticas e de Computação (ICMC), aos seus professores e funcionários pela formação acadêmica, apoio e colaboração. Em especial ao Prof. Dr. Moacir Pereira Ponti Junior, pelo incentivo e orientação no desenvolvimento desta pesquisa.

Aos meus pais, familiares e amigos por todo apoio e carinho.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Siglas

AUC	*Area under the curve*
CH-AD	*Convex Hull Anomaly Detection*
CHF-AD	*Convex Hull Fusion Anomaly Detection*
FN	Falso negativo
FP	Falso positivo
OC-SVM	*One-Class SVM*
ROC	*Receiver operating characteristics*
SIMP	*Signal and Image Measurements and Processing*
SVDD	*Support vector data description*
SVM	*Support vector machine*
TVF	Taxa de falsos positivos
TVP	Taxa de verdadeiros positivos
VN	Verdadeiro negativo
VP	Verdadeiro positivo

Capítulo

1

Introdução

1.1 Contexto e motivação

Detecção de anomalia é a ação de encontrar objetos, observações ou exemplos cujo comportamento não esteja de acordo com o esperado. Segundo Barnett e Louis [3], uma anomalia ou *outlier* é um exemplo ou grupo de exemplos inconsistentes ao conjunto de dados do qual fazem parte. Esses exemplos podem também ser chamados de discordantes, exceções, aberrações, peculiaridades ou contaminantes. É comum, em detecção de anomalias, haver um grande desbalanceamento no número de exemplos rotulados de cada uma das classes, principalmente devido a escassez ou ausência de exemplos anômalos. Técnicas de detecção de anomalias podem ser empregadas em diversas aplicações, como para auxiliar na detecção de falhas, detecção de intrusões em redes, em diagnósticos médicos e no monitoramento do estado de saúde de um paciente, entre outras aplicações.

A detecção de anomalias é crucial em diversos sistemas, uma vez que anomalias podem alertar sobre comportamentos inesperados, como defeitos ou ações ilícitas. Exemplos de eventos que podem ser detectados como anomalias são apresentados por Hodge e Austin [21]: defeitos na rotação de um motor, intrusos com intenções criminosas em um sistema, falhas na linha de produção de uma fábrica e uma doença ou condição médica perigosa, entre outros. Nesse contexto, chama-se o comportamento *não* anômalo de "normal". Esse termo é utilizado sem qualquer relação à distribuição Gaussiana. Desta forma, o comportamento de uma anomalia também pode ser chamado de "anormal".

Um detector de anomalias clássico recebe como entrada um exemplo ou um conjunto de exemplos e identifica se esses são normais ou anormais, de acordo com o comportamento esperado. Define-se como um detector de anomalias, um classificador binário que busca reconhecer quando um exemplo pertence ou não a uma classe de interesse. Na maior parte das aplicações, exemplos que apresentam o comportamento normal estão disponíveis em grande quantidade, enquanto anomalias são escassas [7]. Essa é uma característica intrínseca ao problema de detecção de anomalias e resulta, em geral, em amostras não completas ou não representativas de anomalias.

Os principais desafios envolvidos na detecção de anomalias são: 1) a definição do comportamento normal e a formalização de um modelo para esse comportamento, 2) a tentativa de camuflagem das anomalias originadas a partir de ações maliciosas dentre as normais, 3) a alteração do comportamento normal com o passar do tempo, 4) a escassez ou ausência de dados anormais para realizar o treinamento ou a validação do classificador, 5) a presença de ruídos nos dados normais, que se assemelham a anomalias. Esses desafios fazem com que os métodos de classificação existentes, em geral, não sejam adequados para detectar anomalias, o que estimula a criação de métodos específicos ou adaptações de métodos existentes para tratar esse problema. Dentre os métodos mais utilizados estão os métodos estatísticos e os baseados em teoria da informação [7].

Existem três abordagens principais para realizar a detecção de anomalias: 1) não supervisionada, 2) supervisionada e 3) semi-supervisionada. Na primeira abordagem, quando não há nenhum conhecimento prévio sobre os dados, são utilizados métodos de aprendizado de máquina não supervisionados, como os métodos de agrupamento. A ideia principal dessa abordagem é a de que exemplos que aparentemente não pertençam a nenhum grupo são anomalias. Já as técnicas supervisionadas modelam ambos os comportamentos, normal e anormal, sendo assim necessário que existam exemplos rotulados para ambas as classes.

A abordagem semi-supervisionada ou parcialmente supervisionada busca modelar apenas um dos comportamentos, normalmente o comportamento normal ou, em alguns poucos casos, apenas o comportamento anômalo [10], dessa forma precisando apenas de exemplos rotulados da classe utilizada. Segundo Hodge e Austin [21], as principais vantagens de usar um método semi-supervisionado para detectar anomalias são: 1) necessita somente de exemplos rotulados da classe normal, 2) não assume, a priori, nenhum tipo de distribuição para os dados anômalos, pois modela apenas a classe normal.

Nesta dissertação foca-se principalmente na abordagem parcialmente supervisionada. O método apresentado nela modela apenas o comportamento da classe normal, utilizando

majoritariamente exemplos pertencentes a essa classe. Dessa forma foi possível herdar diversas vantagens da abordagem parcialmente supervisionada.

Além das técnicas criadas especificamente para detectar anomalias, existem também técnicas de classificação adaptadas para detectar fraudes, intrusões em redes de computadores, tentativas de *spoofing* em sistemas biométricos, entre outros [21] [7]. Um exemplo são os métodos *one-class classification* que tentam distinguir conjuntos normais de dados de conjuntos anormais [24]. Alguns métodos para a detecção de anomalias utilizam também múltiplos classificadores, criados sequencialmente pelo método de *Boosting* [12].

Em um típico problema de aprendizado de máquina, incluindo a detecção de anomalias, trabalha-se com dois espaços: o espaço de entrada $\mathcal{X}$ (espaço de características ou espaços de instâncias) e o espaço de saída $\mathcal{Y}$ (espaço de rótulos ou espaço de classes). Nesta dissertação, propõe-se a utilização de um novo espaço, chamado *espaço de parâmetros* com o objetivo de facilitar a detecção de anomalias.

Espaços de parâmetros são criados usando combinações de parâmetros estimados através de pares de exemplos. Define-se o conjunto de todos os espaços de parâmetros como $\widehat{\Theta}^{(k)}$, onde k é um conjunto de parâmetros. A ideia de se utilizar um espaço de parâmetros é motivada pela possibilidade de observar a relação entre os exemplos, enquanto, tradicionalmente, tais exemplos são observados individualmente.

Para mapear os exemplos para um subespaços de parâmetros, primeiramente é realizada a seleção de pares de exemplos e, então, a concatenação desses exemplos. A concatenação é feita como um encadeamento dos vetores de características, semelhante a concatenação de duas *strings*. Após concatenar os dois exemplos, estima-se cada um dos parâmetros escolhidos, gerando um ponto no espaço de parâmetros. Espera-se que, ao estimar os parâmetros a partir da concatenação de dois exemplos, seja possível capturar um modelo que explore o relacionamento entre tais exemplos, ao invés de utilizá-los individualmente.

Nesta dissertação, utiliza-se espaços de parâmetros na tentativa de estimar o comportamento normal da relação entre os exemplos de forma aproximadamente convexa. Sendo assim, o método proposto utiliza exemplos da classe normal para estimar os parâmetros de forma a modelar o comportamento de sua classe. *Subespaços de parâmetros* são criados a partir da combinações entre esses parâmetros e, para cada espaço, são definidas regiões através do cálculo de fechos convexos das estimativas. Tais fechos convexos representam os modelos da classe de interesse. Espera-se que ao concatenar exemplos normais, o fecho convexo das estimativas obtidas será bem comportado ou estável. Como não é possível garantir que a amostra de exemplos normais utilizada para realizar o calculo das

estimativas é totalmente representativa, calcula-se também um limiar para permitir que pequenas variações no fecho convexo não sejam consideradas anomalias.

Sendo o fecho convexo uma representação estável dos dados, anomalias podem ser associadas a perturbações. Portanto, quando realizada a concatenação de um exemplo normal com uma anomalia, espera-se que os parâmetros desviem do comportamento modelado através dos exemplos normais, gerando perturbações no fecho obtido anteriormente e, assim, realizando a detecção. Cada espaço de parâmetros e, consequentemente, cada fecho convexo, pode ser utilizado com um detector. Portanto, a decisão final é obtida através da fusão da saída de todos os detectores.

O método apresentado não tenta definir os limites das classes no espaço de características, como feito por *One Class Support Vector Machine* e *Support Vector Data Description* [55]. Ao invés disso, ele busca explorar a relação entre os exemplos através do cálculo de estimativas dos parâmetros, para cada par de exemplos. A detecção ocorre através da observação de perturbações nos fechos convexos.

Ele também difere do método *RANdom SAmple Consensus (RANSAC)*, o qual tenta separar os exemplos que pertencem a diferentes distribuições através do cálculo de parâmetros lineares. Enquanto o RANSAC precisa receber uma amostra completa (todos os pontos ao mesmo tempo) para realizar a separação dos exemplos, o método apresentado nesta dissertação permite testar exemplos de forma individual, classificando-os como normais ou anomalias. Além disso, não existem diretrizes claras para utilizar o RANSAC em espaços de alta dimensionalidade, e grande parte das aplicações apresentadas nesta dissertação lidam com esse problema.

Neste trabalho tem-se como classe de interesse a classe normal. Nesse cenário, anomalias podem ser geradas através de diferentes distribuições, enquanto os dados normais podem ser modelados através de formas fixas capturadas em diferentes subespaços de parâmetros.

1.2 Organização

Além deste capítulo, esta dissertação é composta por outros quatro capítulos:

- No Capítulo 2 são descritas as principais técnicas concorrentes às apresentadas nesta dissertação. Além disso, também são apresentados os conhecimentos necessários para a melhor compreensão desses métodos e dos experimentos realizados.

- No Capítulo 3, os métodos propostos são explicados em detalhes. Além disso, as bases de dados utilizadas durante os experimentos realizados também são detalhadas.

- No Capítulo 4 estão descritos os experimentos realizados e os resultados obtidos. Para esta dissertação, foram realizados três experimentos que incluem um experimento para avaliar o desempenho do método proposto adaptado para realizar o treinamento com complexidade linear e dois experimentos com o objeto de comparar o desempenho dos métodos propostos com o de métodos concorrentes.

- No Capítulo 5 são apresentadas as conclusões tiradas a partir dos resultados obtidos e as principais contribuições deste trabalho. Além disso, também são indicadas possíveis continuações para o trabalho desenvolvido.

Capítulo

2

Fundamentos

Este capítulo apresenta os conhecimentos necessários para melhor compreensão das técnicas apresentadas e dos experimentos realizados. Nele são apresentados alguns conceitos utilizados para realizar a detecção de anomalias. Em seguida, estão as descrições de técnicas similares ou concorrentes que foram utilizadas para fins de comparação. Além disso, descreve-se o funcionamento de sistemas de múltiplos classificadores e são apresentadas algumas medidas usadas para avaliação de resultados obtidos por métodos de detecção de anomalias.

2.1 Detecção de anomalias

Segundo Hawkins [19], uma anomalia é uma observação que difere tanto das demais ao ponto de levantar suspeitas de que ela foi gerada por um mecanismo diferente. Como pode ser observado na Figura 2.1, as anomalias, representadas pelos pontos a_1, a_2, a_3 e pelo grupo A_4, podem aparecer isoladas dos demais objetos (cujo comportamento segue o esperado, como as amostras dos grupos N_1 e N_2) ou como um conjunto de pontos pequeno e/ou esparso, distantes da grande maioria.

Objetos anômalos podem aparecer nos dados por diversos motivos, como atividades maliciosas, fraudes ou falhas no sistema. Apesar da variedade de motivos possíveis para o surgimento de anomalias nos dados, é interessante identificar e analisar esses objetos,

uma vez que por eles é possível obter informações importantes sobre o evento pelo qual foram gerados.

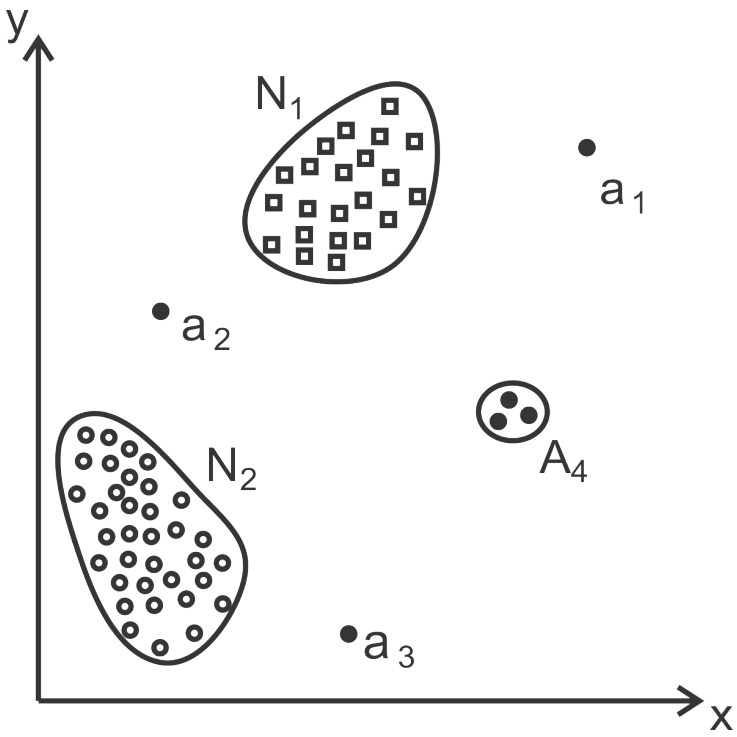


Figura 2.1: Exemplos de *outliers* em um espaço de duas dimensões, onde: N_1 e N_2 são grupos de observações normais; a_1, a_2 e a_3 são anomalias que aparecem isoladas e A_4 um grupo de anomalias. (Adaptada de Chandola e colaboradores [7])

A detecção de anomalias está fortemente relacionada à remoção de ruídos [56] e à detecção de novidades [36] [35]. Entretanto, o objetivo da redução de ruídos é remover dados que representam fenômenos não interessantes ao usuário, removendo estes objetos antes mesmo de realizar qualquer análise. Já na detecção de novidades, o objetivo é detectar objetos que representam um novo padrão nos dados, como um novo tópico de discussão em um grupo de notícias, e incluí-los como parte do conjunto considerado normal depois de realizar uma análise. As técnicas utilizadas para a remoção de ruídos e a detecção de novidades podem ser adaptadas para realizar a detecção de anomalias.

Os métodos de detecção de anomalias em geral recebem uma observação ou conjunto de observações e separam as saídas em duas classes: normal e anormal. Na maior parte das aplicações é comum haver predominância de amostras da classe normal e poucos (ou nenhum) exemplares disponíveis da classe anormal [7]. Como descrito no Capítulo 1, muitos métodos comuns de classificação não funcionam bem para a detecção de anomalias devido às suas limitações em enfrentar os desafios existentes.

Devido a isso, foram desenvolvidos métodos específicos para tratar esse problema. Baseando-se na definição de que uma anomalia é uma observação cujo comportamento não é o esperado para dados normais, uma abordagem direta para a detecção de anomalias seria a definição de uma região do espaço que represente o comportamento considerado normal, assumindo que qualquer observação que não estiver contida nessa região, é uma anomalia. Contudo, diversos fatores, como definir a região que engloba todos os dados normais e a variação do desvio do comportamento normal aceitável conforme a área de aplicação fazem desta, uma abordagem complexa.

Todos os elementos citados fazem da detecção de anomalias um problema de difícil solução. Por isso, a maioria das técnicas existentes foca em uma aplicação específica do problema, utilizando várias informações, como a natureza dos dados, o tipo das anomalias a serem detectadas e a disponibilidade de dados rotulados, para facilitar a tarefa. Alguns conceitos de diversas áreas como estatística, aprendizado de máquinas, teoria da informação e mineração de dados também foram adaptados com o objetivo de realizar a detecção de anomalias em aplicações específicas [7].

2.1.1 Aspectos importantes da detecção de anomalias

Um dos principais aspectos que devem ser considerados ao se escolher uma técnica de detecção de anomalias é a natureza dos dados. As técnicas de detecção de anomalias são comumente aplicadas a conjuntos de observações (bases de dados) e, portanto, são sensíveis aos formatos utilizados pelas amostras ou objetos que compõem esses conjuntos. Cada objeto é composto por um conjunto de um ou mais atributos, onde cada atributo pode ser de um tipo diferente: binários, categóricos ou contínuos. Esses tipos caracterizam a natureza dos dados.

A natureza dos dados define a aplicabilidade das técnicas de detecção de anomalias. Por exemplo, enquanto uma técnica baseada em proximidade dependerá da natureza dos dados para definir o método utilizado para o cálculo das distâncias entre os objetos, uma técnica baseada em modelos estatísticos permite apenas a utilização de dados com atributos categóricos ou contínuos.

Além disso, bases de dados podem ser categorizadas conforme a relação entre os objetos que a compõe. A maioria das técnicas de detecção de anomalias existentes lida com dados pontuais, nos quais não se assume qualquer relação entre as amostras. Entretanto, é possível que haja relação entre os objetos, como em dados sequenciais, dados espaciais ou grafos. Em dados sequenciais, as amostras são ordenadas linearmente e cada amostra pode influenciar diretamente os objetos posteriores a ela; já em dados espaciais, a influência pode ocorrer entre os vizinhos próximos.

Esta grande variedade na natureza dos dados cria também diversos tipos de anomalias que podem ser detectados. O tipo mais simples e mais pesquisado é a anomalia pontual, que é definida quando uma amostra diverge de todo o restante dos dados. Exemplos de anomalias pontuais, representadas pelos pontos a_1, a_2, a_3 e pelos pontos pertencentes ao conjunto A_4, podem ser vistas na Figura 2.1. Além de anomalias pontuais, também são vistas na Figura 2.1 duas classes normais (N_1 e N_2).

Existem também anomalias de contexto, onde uma amostra é considerada anômala somente em um contexto específico e nos demais, esta mesma amostra pode ser considerada normal. A noção de contexto pode ser obtida através da estrutura da base de dados e deve ser especificada na formulação do problema. Esse segundo tipo de anomalia é utilizado principalmente para séries temporais.

Há também as anomalias coletivas, que são caracterizadas nas situações em que as amostras podem não ser consideradas anomalias quando analisadas individualmente; entretanto, a ocorrência conjunta ou sequencial de algumas amostras é uma anomalia. Um exemplo de anomalia coletiva é apresentado na Figura 2.2, que ilustra um eletrocardiograma humano [16] onde a região destacada mostra uma anomalia (pouca variação durante um período de tempo anormalmente grande). É importante notar que momentos com pouca variação porém em um curto espaço de tempo não representam anomalias. Algumas anomalias, como a apresentada no exemplo, são especialmente difíceis de se detectar, pois a maior parte das técnicas de detecção de anomalias e aprendizado de máquina consideram apenas exemplos independentes e identicamente distribuídos (iid). Portanto, torna-se necessária uma técnica especializada para detectar anomalias em conjuntos de dados com essa característica, o que não será feito nesse trabalho.

As técnicas de detecção de anomalias existentes também variam conforme a presença ou não de rótulos nos dados (indicando se cada amostra apresenta um comportamento normal ou anormal) e da quantidade de dados rotulados [7]. É com base nesse critério que são categorizados os métodos de detecção de anomalia, em supervisionado, semi-supervisionado e não supervisionado, como descrito previamente.

Nos métodos supervisionados, assume-se a existência de conjuntos de amostras normais e anormais rotuladas para realizar o treinamento do detector. Uma abordagem comum para esse caso é construir um modelo para prever a classe a qual os novos objetos pertencem. Entretanto, existem dois grandes problemas como os métodos supervisionados, a disponibilidade de amostras da classe anormal é drasticamente menor que a da classe normal para a realização do treinamento e a obtenção de rótulos representativos é difícil, especialmente para as anomalias. No geral, esse grupo de métodos de detecção

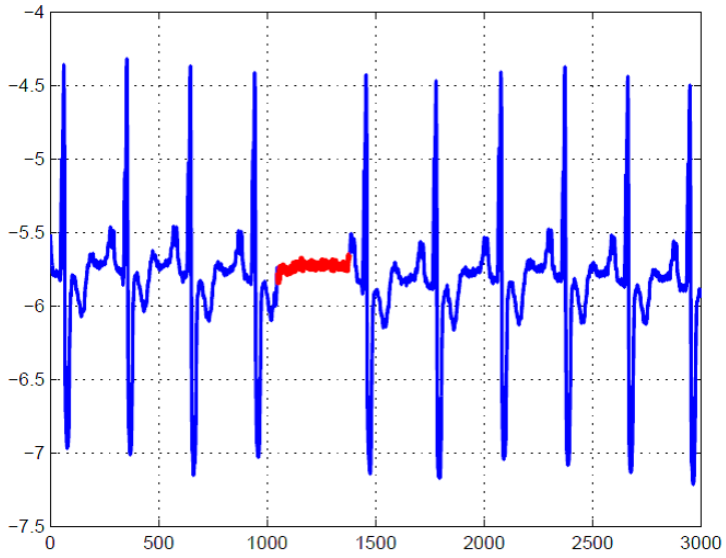


Figura 2.2: Exemplo de uma anomalia coletiva no resultado de um eletrocardiograma, onde a região azul representa a classe normal e a região vermelha, a anormal (Fonte: Goldberger e colaboradores [16]).

de anomalias assemelha-se muito aos problemas enfrentados na construção de modelos preditivos e nos problemas clássicos de classificação [21].

Já as técnicas de detecção de anomalias semi-supervisionadas assumem a existência de um conjunto de amostras rotuladas apenas de uma das classes. É mais comum a utilização de amostras rotuladas da classe normal, mas em alguns poucos casos, apenas o comportamento anormal é utilizado [10]. Devido a não necessidade de amostras rotuladas como anomalias, as técnicas dessa categoria são aplicáveis a um maior número de áreas do que os métodos supervisionados. A abordagem mais comum nessa categoria é a criação de um modelo que corresponde ao comportamento dos objetos normais e a utilização desse modelo para encontrar as anomalias.

Os métodos de detecção de anomalias não supervisionados não necessitam de um conjunto de treinamento rotulado e, portanto, são considerados os mais versáteis dentre as três categorias [7]. Esses métodos assumem implicitamente que as amostras normais são muito mais frequentes do que as anormais. Muitas dessas técnicas podem ser adaptadas para funcionar de modo semi-supervisionado utilizando-se de um conjunto de dados não rotulados como base de treinamento. Tal adaptação assume que um número muito pequeno de anomalias está presente no conjunto de teste e que o modelo aprendido durante o treinamento será robusto a essas anomalias.

Outro aspecto importante das técnicas de detecção de anomalias é a saída gerada para as amostras do conjunto de teste. As saídas mais comuns para essas técnicas são na forma

de rótulos binários, indicando a qual classe (normal ou anormal) pertence cada amostra, ou através de pontuações que indicam quão anômalas são as amostras daquele conjunto de testes, permitindo a produção de um *ranking* baseado nessa pontuação para análise.

2.1.2 Principais aplicações

Existem diversas aplicações que utilizam a detecção de anomalias. Cada aplicação difere das demais quanto à noção de anomalia utilizada para a detecção, a natureza dos dados utilizados, os desafios associados à detecção e às técnicas existentes para realizar a detecção. Serão citadas nesta seção as três principais aplicações: a detecção de intrusos, a detecção de fraudes e a detecção de anomalias em dados médicos.

2.1.2.1 Detecção de intrusos

A detecção de intrusos busca identificar atividades maliciosas em sistemas computacionais. Tais eventos são interessantes do ponto de vista de segurança computacional, uma vez que é possível evitar novas ocorrências baseando-se no comportamento de casos ocorridos previamente [7].

O maior desafio dessa aplicação é que a quantidade de dados a serem analisados é muito grande e, normalmente, são apresentados como um fluxo de dados, sendo necessário realizar a análise em tempo real. Além disso, o grande volume de dados causa muitas ocorrências de falsos alarmes (falsas anomalias), tais ocorrências dificultam bastante a análise dos resultados obtidos. Nessa aplicação é comum existirem somente dados rotulados para a classe normal, possibilitando a utilização de técnicas parcialmente ou não supervisionadas.

2.1.2.2 Detecção de fraudes

A detecção de fraudes visa identificar atividades criminosas que ocorrem em organizações comerciais, como bancos e empresas de cartões de crédito [7]. Comumente, as fraudes ocorrem quando usuários mal-intencionados utilizam recursos providos por essas empresas sem a autorização das mesmas. Essas empresas se interessam pela detecção imediata desses casos para evitar perdas econômicas. Nessa aplicação, é comum monitorar a atividade de cada usuário individualmente de forma a criar um perfil específico para aquele usuário, classificando variações incomuns em seu comportamento como anomalias.

2.1.2.3 Detecção de anomalias em dados médicos

A detecção de anomalias em dados médicos normalmente trabalha com registros de pacientes ou resultados de exames específicos [7]. Podem ser considerados anomalias resultados de exames que fogem do comportamento comum devido à condição do paciente, erros instrumentais ou erros no momento do registro. Técnicas de detecção de anomalias também são utilizadas na tentativa de identificar precocemente surtos de doenças em áreas específicas [59]. A detecção de anomalias em dados médicos é um problema crítico, pois requer um alto nível de acurácia, já que os resultados podem influenciar no tratamento de pacientes.

A maioria das técnicas de detecção de anomalias utilizadas nesse domínio é semi-supervisionada, já que geralmente só existe disponibilidade de amostras rotuladas para a classe normal (pacientes saudáveis), e busca-se encontrar registros anômalos nos dados (anomalias pontuais). Também é comum a análise de fluxos de dados temporais, como eletrocardiogramas e eletroencefalogramas. Nesses casos são aplicadas técnicas para encontrar anomalias coletivas [31]. O maior desafio presente para as técnicas de detecção de anomalias em dados médicos é a grande importância de não classificar erroneamente uma anomalia.

2.1.3 Técnicas para detecção de anomalias

Técnicas de classificação como as baseadas em redes neurais, redes Bayesianas e *support vector machines* (SVM) foram adaptadas para realizar a detecção de anomalias [39] [59]. Essas técnicas buscam aprender um modelo a partir de dados rotulados e utilizar o modelo aprendido para classificar os demais objetos. Sendo assim, na classificação o modelo deverá ser capaz de distinguir entre as classes normal e anormal no espaço utilizado [7]. Essas técnicas podem ser utilizadas como métodos supervisionados ou semi-supervisionados para a detecção de anomalias.

Além das técnicas de classificação, técnicas de agrupamento, técnicas estatísticas e métodos baseados em vizinhos mais próximos também foram utilizados como base para o desenvolvimento de métodos de detecção de anomalias [29] [36] [30]. As técnicas que utilizam a ideia de vizinhos mais próximos, em sua maioria, assumem que os exemplos, cujo comportamento segue o esperado, ocorrem em vizinhanças densas, enquanto anomalias aparecem distantes de seus vizinhos mais próximos. Portanto, para utilizá-las é necessário que haja pelo menos dados normais classificados para realizar o treinamento, fazendo com que esses métodos sejam semi-supervisionados ou supervisionados.

Em geral, nas técnicas baseadas em métodos estatísticos, utiliza-se um modelo estocástico para definir o comportamento da classe que possui disponibilidade de amostras rotuladas e considera-se os objetos que ocorrerem em regiões de baixa probabilidade do modelo como pertencentes a outra classe. Geralmente, devido à maior disponibilidade de amostras rotuladas da classe normal, cria-se o modelo do comportamento normal e as amostras que apresentarem baixas probabilidades são consideradas anomalias.

As técnicas baseadas em métodos estatísticos podem ser divididas em duas categorias diferentes: paramétricas e não-paramétricas. Nos métodos paramétricos, considera-se que os dados normais são gerados a partir de uma distribuição específica, definida pelo usuário, com função de densidade de probabilidade. Os parâmetros dessa distribuição são obtidos através dos dados de treinamento. Já nos métodos não paramétricos, o modelo estrutural dos dados não é definido *a priori*, mas determinado conforme os dados do conjunto de treinamento, fazendo com que sejam necessárias realizar menos suposições sobre os dados. Entretanto, é necessário garantir que o conjunto de dados de treinamento seja grande o suficiente para estimar bem a distribuição das amostras como um todo.

A detecção baseada em agrupamento [23] [54] busca reunir dados similares em um mesmo grupo, assumindo que os dados normais se reúnem em determinadas regiões do espaço, enquanto as anomalias aparecem mais dispersas. A partir dessa ideia, as anomalias podem ser definidas de três modos: objetos que não pertencem a grupo nenhum; amostras que se encontram na periferia dos grupos, distantes de seus centros e dados presentes em grupos pequenos ou esparsos. A maior vantagem das técnicas de detecção de anomalias baseadas em agrupamento é a possibilidade de trabalhar de modo não supervisionado, ou seja, não é necessária a utilização de amostras rotuladas.

Existem também métodos de detecção de anomalias baseados em teoria da informação [28] [20] [2]. Esses métodos são baseados na suposição de que anomalias em dados induzem irregularidades no conteúdo da informação. Por exemplo, seja $\mathcal{C}(D)$ a complexidade de um conjunto de dados D. Um algoritmo básico busca pelo subconjunto mínimo de amostras I de forma que $\mathcal{C}(D) - \mathcal{C}(D - I)$ seja máximo, ou seja, maximiza a redução da complexidade. Todas as amostras no subconjunto I são consideradas anômalas. A complexidade $\mathcal{C}$ pode ser medida pelo tamanho do arquivo do conjunto de dados comprimido (estimação da medida da complexidade de Kolomogorov) [25] ou pela incerteza relativa [2]. As técnicas envolvem otimização dual (maximização da redução da complexidade e minimização do subconjunto I). Apesar da vantagem de não assumirem uma distribuição estatística para os dados, a alta complexidade computacional (o algoritmo básico possui complexidade exponencial), a grande influência da escolha da

medida de complexidade e a necessidade de uma quantidade suficientemente grande de amostras para realizar na detecção de anomalias [7] são desvantagens desses métodos.

A seguir estão descritas algumas técnicas de detecção de anomalias e classificação utilizadas para comparação com os métodos propostos.

2.1.3.1 Método paramétrico baseado na distribuição Gaussiana

Essa técnica de detecção de anomalia é semi-supervisionada, baseada em um método estatístico paramétrico e assume que os dados são gerados a partir de uma distribuição Gaussiana, também conhecida como distribuição Normal. Nela, os parâmetros da distribuição são estimados para o caso normal a partir de um conjunto de treinamento e, por meio de um conjunto de validação, encontra-se um limiar experimental T. Esse limiar deve especificar a probabilidade mínima que uma nova amostra deve ter para que seja considera como tendo a mesma origem das amostras do conjunto de treinamento, ou seja, para que seja classificada como normal. O limiar T a ser utilizado também pode ser definido como a região contendo 99.7% dos dados, ou seja, a região $\mu \pm 3\sigma$ [50], sendo μ a média da distribuição e σ o seu desvio padrão.

Para esse método é possível utilizar os modelos estocásticos Gaussiano uni e multivariados. No caso do modelo univariado, cuja função de densidade de probabilidade pode ser observada na Equação 2.1, assume-se que não existe nenhuma relação entre os atributos de um único objeto, uma vez que, para cada atributo, é utilizada uma distribuição Gaussiana diferente, com média e variância calculadas a partir daquele atributo das amostras presentes na base de treinamento. A probabilidade final de cada amostra é dada a partir da multiplicação das probabilidades calculadas para cada atributo.

$$f(x) = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(x-\mu)^2}{2\sigma^2}} \tag{2.1}$$

Já ao utilizar o modelo Gaussiano multivariado, com função de densidade de probabilidade apresentada na Fórmula 2.2, cria-se a possibilidade de considerar relações entre os atributos de um mesmo objeto. Isso é possível devido à substituição da variância σ pela matriz de covariâncias Σ, também obtida a partir das amostras da base de treinamento, possibilitando assim a existência de covariâncias entre os atributos de cada objeto.

$$f(x) = \frac{1}{\sqrt{(2\pi)^M |\Sigma|}} e^{-\frac{(x-\mu)^T \Sigma^{-1} (x-\mu)}{2}} \tag{2.2}$$

2.1.3.2 *Naive* Bayes

O classificador *Naive* Bayes é probabilístico, baseado na aplicação do teorema de Bayes, assumindo que os atributos de um objeto são totalmente independentes, por isso o algoritmo é considerado ingênuo (*naive*).

O teorema de Bayes define uma relação entre uma probabilidade condicional e sua inversa, ou seja, da probabilidade de uma hipótese dada a observação de uma evidência e da probabilidade da evidência, assumindo que a hipótese ocorreu. O teorema de Bayes é representado pela Equação 2.3, em que: $P(A)$ e $P(B)$ são as probabilidades *a priori* de A e B e $P(A|B)$ e $P(B|A)$ são as probabilidades *a posteriori* de B dado que A ocorreu e de A dado que B ocorreu, respectivamente.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \tag{2.3}$$

Em um problema de classificação, este teorema é utilizado para encontrar a probabilidade de um dado objeto x, com atributos $F_1, F_2, \ldots, F_n$, pertencer à classe C (Equação 2.4).

$$P(C|F_1, F_2, \ldots, F_n) = \frac{P(F_1, F_2, \ldots, F_n|C)P(C)}{P(F_1, F_2, \ldots, F_n)} \tag{2.4}$$

Como o valor do denominador é independente da classe C e os valores dos atributos $F_1, F_2, \ldots, F_n$ são conhecidos, o denominador será constante. Isso permite ocultar temporariamente o denominador para facilitar a visualização. Ao assumir-se de forma ingênua que não há relação entre os atributos pode-se utilizar a Equação 2.5 e, consequentemente, a Equação 2.6.

$$P(F_i|C, F_j) = P(F_i|C) \tag{2.5}$$

$$\begin{aligned} P(C|F_1, F_2, \ldots, F_n) &\approx P(C)P(F_1, F_2, \ldots, F_n|C) \\ &\approx P(C)P(F_1|C)P(F_2, F_3, \ldots, F_n|C, F_1) \\ &\approx P(C)P(F_1|C)P(F_2|C)P(F_3, F_4, \ldots, F_n|C, F_1, F_2) \\ &\approx P(C)P(F_1|C)P(F_2|C)\ldots P(F_n|C, F_1, F_2, \ldots, F_{n-1}) \\ &\approx P(C)P(F_1|C)P(F_2|C)\ldots P(F_n|C) \end{aligned} \tag{2.6}$$

Portanto, pode-se concluir que o algoritmo *Naive* Bayes atribui a distribuição condicional apresentada na Equação 2.7 sobre a variável que define a classe C, dado que

Z é um fator que depende somente de $F_1, F_2, \ldots, F_n$ e que é constante caso os valores dos atributos sejam conhecidos. A classe que apresentar a maior probabilidade dados os atributos do objeto a ser classificado é a classe escolhida para aquele objeto.

$$P(C|F_1, F_2, \ldots, F_n) = \frac{1}{Z} P(C) \prod_{i=1}^{n} P(F_i|C) \tag{2.7}$$

No contexto de detecção de anomalias, o Naive Bayes pode ser utilizado como um método supervisionado.

2.1.3.3 *Support Vector Data Description* e *One-Class Support Vector Machine*

Os métodos *Support Vector Data Description* (SVDD) [55] e *One-Class Support Vector Machine* (OC-SVM) [46] são muito semelhantes e ambos utilizam a abordagem de aprendizado *one-class*. Nessa abordagem utiliza-se somente dados de uma das classes para realizar o treinamento e seus principais objetivos são a distinção entre duas classes quando há um grande desbalanceamento na disponibilidade das mesmas e a detecção de *outliers*.

A estratégia utilizada pelo SVDD é mapear os dados em um espaço de alta dimensionalidade utilizando uma função *kernel* e então usar uma hiper-esfera para descrevê-los, colocando a maior parte dos dados dentro da hiperesfera e considerando como vetores de suporte as amostras que ficarem na parte externa ou na borda da hiperesfera. Encara-se então esse problema como um problema de otimização, no qual se tenta encontrar a menor hiperesfera possível que contenha a maior parte das amostras. Já o OC-SVM busca, também em um espaço de alta dimensionalidade, encontrar um hiperplano que separe as amostras do conjunto de treinamento da origem, sendo os vetores de suporte definido pelas amostras que forem consideradas anomalias e as que pertencerem ao hiperplano encontrado. Nesse caso, assume-se que a origem representa todos os pontos de baixa similaridade com a base de treinamento e busca-se maximizar a margem entre o hiperplano encontrado e um hiperplano paralelo a esse que passa pela origem.

Para formalizar a estrutura utilizada pelo SVDD, deve-se primeiro definir a função $F(R, a, \xi)$ a ser minimizada (Equação 2.8) e suas restrições (Equações 2.9 e 2.10), nas quais a e R são, respectivamente, o centro e o raio da hiperesfera; x_i é o conjunto de treinamento, com $i = 1, \cdots, N$; $\varphi(x_i)$, o espaço de características do conjunto x_i; ξ_i variáveis inseridas para possibilitar que o algoritmo considere a existência de *outliers* no conjunto de treinamento, permitindo com que a distância de x_i até a não precise ser estritamente menor do que R^2, mas penalizando distâncias maiores; e C, o parâmetro que controla o balanceamento entre o volume da hiperesfera e os erros cometidos.

$$F(R, a, \xi) = R^2 + C \sum_{i=1}^{N} \xi_i \tag{2.8}$$

$$\|\varphi(x_i) - a\|^2 \leq R^2 + \xi_i, \forall i \tag{2.9}$$

$$\xi_i \geq 0, \forall i \tag{2.10}$$

A partir da utilização de multiplicadores de Lagrange α_i e γ_i para incorporar as restrições (Equações 2.9 e 2.10) à Equação 2.8, obtêm-se a Equação 2.11 que deve ser minimizada com relação a R, a e ξ_i e maximizada com relação a α_i e γ_i, sendo $\alpha_i \geq 0$ e $\gamma_i \geq 0$.

$$\mathcal{L}(R, a, \xi, \alpha, \gamma) = R^2 + C \sum_{i=1}^{N} \xi_i - \sum_{i=1}^{N} \alpha_i [R^2 + \xi_i - (\varphi(x_i) - a)^T (\varphi(x_i) - a)] - \sum_{i=1}^{N} \gamma_i \xi_i \tag{2.11}$$

Encontrando-se as derivadas parciais e igualando-as a zero, chega-se nas restrições apresentadas nas Equações 2.12, 2.13 e 2.14, sujeitas a $0 \leq \alpha_i \leq C$ e $\sum_{i=1}^{N} \alpha_i = 1$.

$$\frac{\partial \mathcal{L}}{\partial R} = 2R - 2R \sum_{i=1}^{N} \alpha_i = 0 \Rightarrow \sum_{i=1}^{N} \alpha_i = 1 \tag{2.12}$$

$$\frac{\partial \mathcal{L}}{\partial a} = 2 \sum_{i=1}^{N} \alpha_i a - 2 \sum_{i=1}^{N} \alpha_i \varphi(x_i) = 0 \Rightarrow a = \sum_{i=1}^{N} \alpha_i \varphi(x_i) \tag{2.13}$$

$$\frac{\partial \mathcal{L}}{\partial \xi_i} = C - \alpha_i - \gamma_i = 0 \Rightarrow \alpha_i + \gamma_i = C \tag{2.14}$$

Da Equação 2.14 e das restrições $\alpha_i \geq 0$ e $\gamma_i \geq 0$, é possível afirmar que $0 \leq \alpha_i \leq C$, se $\sum_{i=1}^{N} \alpha_i = 1$.

Então, substituindo as restrições encontradas nas Equações 2.12, 2.13 e 2.14 à Equação 2.11 e considerando que o produto entre dois espaços de características $(\varphi(x_i) \cdot \varphi(x_i))$ pode ser substituido por uma função kernel $\mathcal{K}(x_i, x_j)$, chega-se na Equação 2.15, na qual deve-se encontrar o valor para α que maximize a função, considerando que $0 \leq \alpha_i \leq C$ e $\sum_{i=1}^{N} \alpha_i = 1$.

$$\mathcal{L}(R, a, \xi, \alpha, \gamma) = \arg\max_{\alpha} \left(\sum_{i=1}^{N} \alpha_i \mathcal{K}(x_i, x_i) - \sum_{i=1}^{N} \sum_{j=1}^{N} \mathcal{K}(x_i, x_j) \right) \tag{2.15}$$

Para chegar na teoria utilizada pelo OC-SVM a partir do SVDD, considera-se que a função *kernel* é invariante à translação, ou seja, $\mathcal{K}(x_i, x_j)$ é constante sobre x. Assim, simplifica-se o problema para a Equação 2.16, também com as restrições: $0 \leq \alpha_i \leq C$ e $\sum_{i=1}^{N} \alpha_i = 1$. Para o OC-SVM, a origem representa todos os pontos que apresentam baixa similaridade com o conjunto de treinamento, já que $\mathcal{K}(x_i, \overrightarrow{0}) = 0$. Quando a função *kernel* é invariante à translação, todas as observações aparecem no perímetro de uma esfera no novo espaço. Nesse caso, o OC-SVM e o SVDD cortam a mesma parte da esfera e, consequentemente, utilizam os mesmos vetores de suporte. Uma representação pode ser vista na Figura 2.3, em que os vetores de suporte são representados por quadrados, as amostras normais por círculos, o SVM pela cor verde e o SVDD por vermelho.

$$\mathcal{L}(R, a, \xi, \alpha, \gamma) = \arg_{\alpha} \max \left(\sum_{i=1}^{N} \sum_{j=1}^{N} \mathcal{K}(x_i, x_j) \right) \tag{2.16}$$

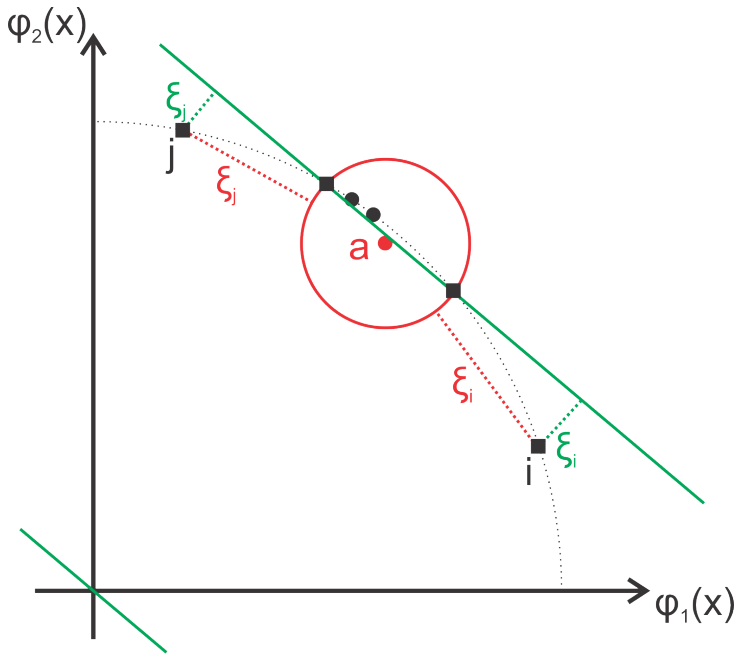


Figura 2.3: Representação da abordagem utilizada pelo SVM (em verde) e pelo SVDD (em vermelho). Os vetores de suporte estão representados por quadrados, a é o centro da hiperesfera escolhida pelo SVDD e ξ_i é o custo de cada anomalia encontrada na base de treinamento.

2.2 Classificadores baseados em cascas convexas

Existem diversos classificadores que buscam identificar uma área no espaço dos atributos que defina a classe dos objetos [53]. Normalmente, esses classificadores determinam

as regiões de forma indireta através de funções ou estimando uma fronteira de decisão. Ainda assim, alguns classificadores estão ligados a determinados tipos de regiões. Por exemplo, uma máquina de vetores de suporte (do inglês *Support Vector Machine*, SVM) linear [57] utiliza cascas convexas para determinar a área correspondente a duas classes. Já um classificador baseado em vizinhos mais próximos utiliza um diagrama de Voronoi das amostras [8].

Os dois exemplos de métodos que mais se destacam na literatura são o classificador baseado em cascas convexas [53] e o SVM de uma classe, que utiliza como base a descrição de dados por vetores de suporte (do inglês *Support Vector Data Description*, SVDD) [55], ambos descritos na Seção 2.1.3.3.

2.3 Sistemas de múltiplos classificadores

Sistemas de múltiplos classificadores buscam aumentar a acurácia da classificação através da combinação do resultado de diversos classificadores. Essa abordagem se tornou muito popular na década de 1990 com diversos artigos publicados que demonstram o porquê combinar classificadores pode ser interessante [27]. Através da combinação de diversos classificadores é possível lidar melhor com dados com alta taxa de ruído, uma vez que a diversidade de classificadores utilizadas no sistema aumenta sua robustez [43]. Além disso, também é possível diminuir a instabilidade de algumas redes neurais artificiais utilizando técnicas de combinação de múltiplos classificadores [5].

De acordo com Dietterich [58], existem três motivações principais para utilizar um sistema de múltiplos classificadores:

- É possível evitar o pior classificador utilizando a média de diversos classificadores [14], apesar de não haver garantia de que o resultado obtido será melhor do que o apresentado pelo melhor classificador.
- Em algumas circunstâncias, a fusão de múltiplos classificadores pode melhorar o desempenho apresentado pelo melhor classificador individualmente.
- A combinação de classificadores cujo desempenho depende de sua inicialização pode obter um resultado final melhor e evitar a necessidade de realizar diversas execuções com inicializações aleatórias, tornando, também, o resultado mais estável [5].

Existem duas abordagens principais para a combinação de diversos classificadores: fusão e seleção. A fusão assume que os classificadores a serem combinados tem conhecimento sobre todo o espaço de características e sua resposta é o resultado de uma opinião

coletiva. Já a seleção utiliza classificadores que são especialistas em apenas parte do espaço de características, sendo que a classificação de um objeto é dada pelo especialista da área onde este objeto esta localizado. Métodos híbridos ponderam a opinião de cada especialista conforme a distância entre sua área de especialidade e a localização do objeto.

Ao criar um sistema de múltiplos classificadores é importante garantir que existe certa diversidade entre os classificadores utilizados. Essa diversidade pode ser criada utilizando-se diferentes algoritmos para cada classificador, diferentes conjuntos de treinamento ou parametrizações distintas.

2.4 Avaliação de métodos de detecção de anomalias

Existem diversas medidas que podem ser utilizadas para avaliar os resultados apresentados pelos algoritmos de detecção de anomalias. Dentre estas, as mais utilizadas fazem uso dos dados presentes na matriz de confusão. A matriz de confusão é composta pelo mesmo número de linhas e colunas, ambos iguais ao número de classes a serem detectadas pelo algoritmo. Cada coluna da matriz representa os casos distribuídos nas classes pelo algoritmo e cada linha representa os casos nas classes das quais eles realmente pertencem.

No caso de detecção de anomalias, tem-se somente duas classes: normal e anômala. Quando se utiliza o algoritmo para detectar as anomalias obtêm-se quatro resultados: 1) verdadeiros positivos (VP), 2) falsos positivos (FP), 3) verdadeiros negativos (VN) e 4) falsos negativos (FN). O primeiro resultado mostra quantos casos foram identificados como anomalias corretamente. O segundo, quantos casos foram identificados como anomalias mas apresentam comportamento normal. O terceiro, o número de casos normais que também foram classificados como tal. Já, o quarto resultado mostra quantas anomalias foram classificadas erroneamente.

	Classe Predita	
Classe Verdadeira	Verdadeiro Positivo (VP)	Falso Negativo (FN)
	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Figura 2.4: Esquemático de uma Matriz de Confusão

Através dos valores presentes na matriz de confusão (ver Figura 2.4) é possível calcular várias medidas de desempenho utilizadas para a avaliação dos algoritmos de classificação

e detecção de anomalias. As mais utilizadas são: a sensibilidade ou taxa de verdadeiros positivos (TVP) (Equação 2.17), taxa que considera apenas os exemplos verdadeiramente positivos para mostrar qual a porcentagem de acerto do algoritmo; a taxa de falsos positivos (TFP) (Equação 2.18), relação entre a quantidade amostras da classe positiva classificadas erroneamente e o número total de amostras classificadas de forma errada; a especificidade (Equação 2.19), que, assim como a sensibilidade, mostra qual a porcentagem de acerto do algoritmo, mas com relação aos casos negativos; a medida-F1 (Equação 2.20), que combina a precisão e a taxa de acerto do algoritmo em uma medida, e a acurácia (Equação 2.21), medida que considera a quantidade de acertos sobre o total de amostras.

$$TVP = \frac{VP}{VP + FN} \tag{2.17}$$

$$TFP = \frac{FP}{FP + VN} \tag{2.18}$$

$$E = \frac{VN}{VN + FP} \tag{2.19}$$

$$F1 = \frac{2 \cdot VP}{2 \cdot VP + FN + FP} \tag{2.20}$$

$$A = \frac{VP + VN}{VP + VN + FP + FN} \tag{2.21}$$

Apesar de muito utilizadas, a maioria dessas medidas assume que os dados utilizados para analisar o desempenho do algoritmo apresentam certo balanceamento no número de exemplos pertencentes a cada classe e isso não ocorre no caso de detecção de anomalias. Por isso, algumas dessas medidas foram adaptadas de forma a ficarem mais robustas ao desbalanceamento dos dados. A principal medida adaptada é a acurácia balanceada.

Nela atribui-se uma fração do valor final que varia entre 0 e 1 para os acertos em cada uma das classes, ou seja, no caso da detecção de anomalias onde se tem duas classes, $0,5$ corresponde ao acerto de todos os objetos da classe normal, enquanto os outros $0,5$ correspondem ao acerto de todos os objetos anômalos, mesmo que o número de anomalias seja muito inferior ao número de objetos normais. Dessa forma, atribui-se um peso maior para acertos nas classificações dos casos anômalos. Uma das dificuldades do uso desses métodos de avaliação é que o cálculo para bases de dados multi-classes não é tão trivial.

Para realizar os cálculos da acurácia balanceada utiliza-se a Equação 2.22, onde c é o número de classes e $E(i)$ é o erro parcial de c, encontrado através das Equações 2.23,

2.24 e 2.25. Também são necessários para realizar o cálculo da acurácia balanceada: o número total de objetos no conjunto de testes N, o número de amostras de cada classe $N(i)$ e os valores de falsos positivos e falsos negativos ocorridos para cada classe i ($FP(i)$ e $FN(i)$).

$$Ac = 1 - \frac{\sum_{i=1}^{c} E(i)}{2c} \tag{2.22}$$

$$E(i) = e_{i,1} + e_{i,2} \tag{2.23}$$

$$e_{i,1} = \frac{FP(i)}{N - N(i)} \tag{2.24}$$

$$e_{i,2} = \frac{FN(i)}{N(i)} \tag{2.25}$$

A acurácia balanceada é uma medida mais simples de interpretar tanto para bases de dados binárias quanto multi-classe e por isso é utilizada para avaliação dos métodos neste trabalho.

2.5 Considerações Finais

Os conceitos apresentados nesse capítulo foram utilizados como base para desenvolver as técnicas apresentadas no Capítulo 3. Além disso, as técnicas já existentes aqui apresentadas foram utilizadas para comparação de resultados, já que representam o atual estado da arte da detecção de anomalias. Essa comparação será feita utilizando as medidas de avaliação mostradas. No próximo capítulo, serão apresentados o métodos propostos por esta dissertação e algumas bases de dados utilizadas para experimentos de detecção de anomalias.

Capítulo

3

Metodologia

Neste capítulo são apresentados os novos métodos propostos e as bases de dados utilizadas durante os experimentos. O primeiro método descrito (Seção 3.1) tem como base a utilização de espaços de parâmetros para facilitar a detecção de anomalias. O espaço de parâmetros possibilita considerar a relação entre as amostras da base para realizar a detecção, diferente dos demais métodos existentes que usam as próprias amostras como base. O segundo método, apresentado na Seção 3.2, é uma adaptação do primeiro que utiliza uma fusão de classificadores utilizados em cada atributo para realizar a detecção. Além disso, também são apresentados alguns conjuntos de dados que foram utilizados para realizar experimentos com os métodos propostos.

3.1 Detecção de anomalias em espaços de parâmetros utilizando fechos convexos

O método proposto, chamado *Convex Hull Anomaly Detector* (CH-AD), utiliza uma forma convexa para modelar a classe de interesse. Um conjunto é considerado convexo se todo segmento de reta ligando dois pontos pertencentes à esse conjunto estiver completamente contido no conjunto [4]. Formas convexas são muito comuns e podem ser encontradas em diversas aplicações do mundo real, desde em salas que buscam criar um ambiente que facilita a difusão acústica, até na geração de poliedros em modelos de

proteínas [33]. Comparar fechos convexos é considerado um método viável para comparar padrões complexos, tais como proteínas e encaixe de moléculas [37], pois geram modelos simplificados para representar tais padrões. Além disso, pode ser usado para descrever dados, sendo robusto à presença de ruídos [17].

A principal ideia de calcular o fecho convexo é aproximar um conjunto não convexo para um conjunto convexo, já que a análise de conjuntos convexos é composta por métodos mais consolidados na literatura, por exemplo, a otimização convexa tem um tratamento matemático mais simples quando comparada à não convexa [4]. Isso é feito selecionando o fecho convexo com menor volume dentre todos os fechos convexos que englobam todos os pontos contidos no conjunto [33]. Apesar de suas vantagens, a utilização de fechos convexos pode ocasionar problemas ao incluir subespaços que não pertencem originalmente ao conceito que se deseja modelar, ou ainda que sejam supérfluos, ou seja, que delimitem um espaço extra de forma desnecessária.

Nos métodos apresentados nesta dissertação, utiliza-se fechos convexos em espaços chamados espaços de parâmetros. Cada um desses espaços é criado através da combinação de parâmetros. Durante a explicação dos métodos foram usados parâmetros arbitrários, já para os experimentos, os conjuntos de parâmetros utilizados foram: média e desvio padrão; média, variância, obliquidade e curtose.

A criação de um espaço de parâmetros pode ser visto como o resultado da aplicação de diversas funções *kernel*. Uma função *kernel* é definida como uma medida de similaridade k onde:

$$k : \mathcal{X} \times \mathcal{X} \to \mathbb{R},$$

$$(x, x') \mapsto k(x, x'),$$

isto é, uma função que, dados dois exemplos x e x', retorna um número real que descreve sua similaridade [47]. Para a criação do espaço de parâmetros, cada parâmetro é calculado como um *kernel*, porém utiliza-se a concatenação dos exemplos antes da realização do cálculo final. Então, utiliza-se cada parâmetro como uma das dimensões de um espaço, gerando o chamado espaço de parâmetros.

A principal motivação para utilizar um espaço de parâmetros está na observação da relação entre amostras, ao invés da observação de amostras isoladas, como comumente é tratado o problema. Espera-se que a utilização de espaços de parâmetros facilite a detecção de exemplos que tenham comportamento diferente do esperado, modelado a partir de um conjunto de treinamento.

3.1.1 Etapa de treinamento

Sendo s_i o i-ésimo exemplo de um dado conjunto de treinamento S_i e n o número de exemplos da classe de interesse presentes no conjunto S_i. Esses exemplos são vetores de características tais que $s_i \in \mathbb{R}^m$ para todo i, sendo m o tamanho do vetor de características.

A fase de treinamento inicia-se com o cálculo de estimativas utilizando apenas amostras rotuladas da classe de interesse. Tais estimativas são calculadas através da seleção aleatória de pares de exemplos. Para cada par i, j, múltiplos parâmetros podem ser calculados. A notação utilizada para denotar os parâmetros foi $\{\widehat{\theta}^{(1)}, \widehat{\theta}^{(2)}, \widehat{\theta}^{(3)}, \ldots\}$. Por exemplo, assumindo-se dois exemplos s_i e s_j, primeiramente os vetores de características correspondentes a esses são concatenados, formando um único vetor $v_{i,j} \in \mathbb{R}^{2m}$. Então, cada estimativa $\widehat{\theta}^{(p)}$, $p \geq 1$, é calculada a partir do vetor v. Podendo, por exemplo, $p = 1$ ser a média, $p = 2$ a variâncias, e assim por diante.

O número de pares a serem selecionados para o cálculo das estimativas é arbitrário, sendo no máximo $\frac{N(N-1)}{2}$. Para os experimentos realizados escolheu-se $2n$ pares. Essa escolha é discutida na Seção 4.1.

Dado um conjunto de parâmetros $\widehat{\theta}$, é possível gerar múltiplos espaços de parâmetros. É importante lembrar que para o cálculo de um fecho convexo é necessário que o espaço contenha, pelo menos, duas dimensões. Chama-se cada espaço de parâmetro $\widehat{\Theta}^{(k)}$. A Figura 3.1 ilustra os passos traçados para a obtenção de um espaço de parâmetros de duas dimensões.

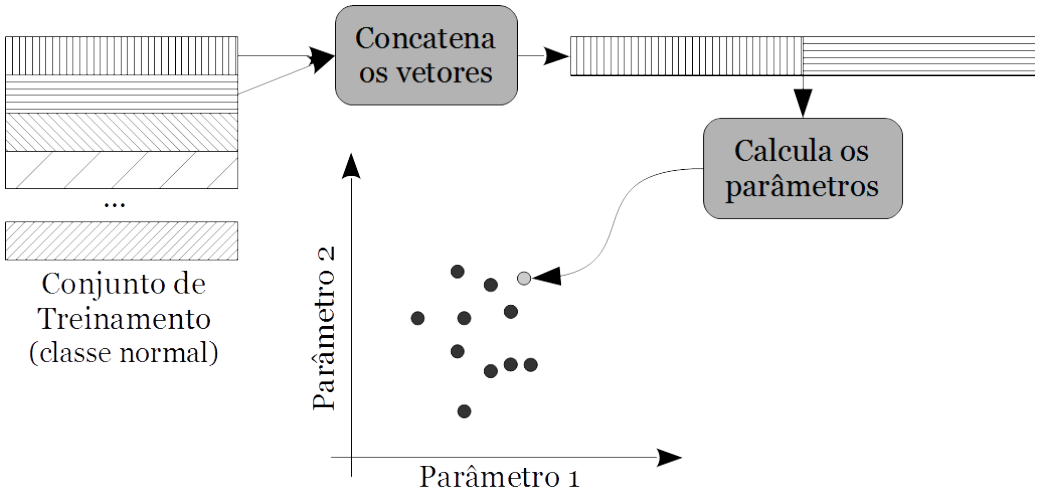


Figura 3.1: Geração de um conjunto de pontos (estimativas dos parâmetros $\widehat{\theta}$) utilizando amostras da classe normal.

O conjunto de parâmetros escolhido determinará quais as características do espaço gerado. Um exemplo seria um espaço composto por média e variância, produzindo um espaço $\widehat{\Theta}^{(1,2)} = \{\widehat{\theta}^{(1)}, \widehat{\theta}^{(2)}\}$ em $\mathbb{R} \times \mathbb{R}_+$.

O fecho convexo encontrado no espaço $\widehat{\Theta}^{(k)}$ é chamado $\mathcal{H}(\widehat{\Theta}^{(k)})$. A escolha dos parâmetros utilizados para a criação dos múltiplos espaços de parâmetros deve ser feita de acordo com o comportamento da classe de interesse. O Algoritmo 1 corresponde à toda etapa de treinamento.

Para cada fecho convexo $\mathcal{H}(\widehat{\Theta}^{(k)})$ é necessário obter um limiar que representa a quantidade de perturbação no fecho convexo que será permitida sem considerar que a nova amostra seja uma anomalia. Esse limiar é obtido a partir de um conjunto de validação composto por alguns poucos exemplos de ambas as classes e será utilizado posteriormente para classificar novos exemplos.

Cada um dos fechos convexos obtidos será utilizado para avaliar novos exemplos. Sendo assim, os únicos exemplos e estimativas que precisam ser armazenados para utilização nas próximas etapas são aqueles que contribuíram para os fechos, os demais podem ser descartados.

Algoritmo 1 Etapa de treinamento do algoritmo CH-AD

Requer: conjunto de treinamento com N amostras da classe de interesse, número de pares $M = (c \cdot N)$, em que M é múltiplo de N e menor do que $\frac{N(N-1)}{2}$, conjunto de validação contendo um pequeno número de amostras de ambas as classes.

1: Estimar $\left\{\widehat{\theta}^{(1)}, \widehat{\theta}^{(2)}, \dots\right\}$ com:
2: **para** cada parâmetro $\widehat{\theta}^{(i)}$ **faça**
3: Selecionar M pares de exemplos (a, b) sendo $a \neq b$
4: **para** cada par i selecionado, $i = 1, \dots, M$ **faça**
5: $v_i \leftarrow$ concatenação de a e b
6: Estimar $\widehat{\theta}_i$ a partir de v_i
7: **fim para**
8: **fim para**
9: **para** cada combinação de parâmetros k, sem repetição **faça**
10: Calcular $\mathcal{H}(\widehat{\Theta}^{(k)})$
11: Usar o conjunto de validação composto por uma pequena quantidade de exemplos de ambas as classes para obter o limiar $T^(k)$
12: **fim para**

A Figura 3.2 ilustra toda a etapa de treinamento. Em (a) e (b), os pares de exemplos são selecionados e, a partir deles, são calculadas as estimativas dos parâmetros. Em (c), as estimativas encontradas são representadas em um gráfico de dispersão, onde, em (d), é também representado o fecho convexo das estimativas. O resultado da etapa

de treinamento é ilustrado em (e), composto pelo fecho convexo e os exemplos que contribuíram para as estimativas que compõem o fecho.

3.1.2 Etapa de detecção

A ideia utilizada pelo método CH-AD é que um exemplo que não pertença a classe de interesse, quando concatenada com os exemplos que contribuíram para o fecho convexo obtido na etapa de treinamento, causará grandes perturbações nesse fecho convexo, possibilitando detectar exemplos que não sigam o comportamento dessa classe. Já um exemplo que pertença a classe de interesse causará pequenas ou nenhuma perturbação no fecho convexo. Tais perturbações são medidas a partir da diferença entre as áreas de ambos os fechos convexos: o fecho obtido durante a fase de treinamento e o obtido através da combinação do novo exemplo com os que contribuíram para o fecho durante a etapa de treinamento.

Para encontrar o segundo fecho convexo, cria-se um novo conjunto de estimativas $\widetilde{\Theta}^{(k)}$ e compara-se esse conjunto com $\widehat{\Theta}^{(k)}$, obtido durante a etapa de treinamento. Para criar o conjunto $\widetilde{\Theta}^{(k)}$, o novo exemplo que deve ser classificado é concatenado com todos os exemplos que contribuíram para o fecho convexo $\mathcal{H}(\widehat{\Theta}^{(k)})$. Através dessas concatenações, calcula-se as estimativas $\{\widetilde{\theta}^{(1)}, \widetilde{\theta}^{(2)}, \widetilde{\theta}^{(3)}, \ldots\}$, obtendo-se então o conjunto $\widetilde{\Theta}^{(k)}$. Finalmente, encontra-se o fecho convexo $\mathcal{H}(\widetilde{\Theta}^{(k)})$, como ilustrado pela Figura 3.3, e compara-se com o fecho convexo original $\mathcal{H}(\widehat{\Theta}^{(k)})$ através dos seus volumes.

Espera-se que, quando somente exemplos pertencentes a classe de interesse são concatenados, ambos os fechos convexos serão similares. Já, quando um exemplo anormal, não pertencente a classe de interesse, estiver presente no conjunto responsável pela criação do segundo fecho convexo, esse será maior do que o obtido durante a etapa de treinamento, como mostrado na Figura 3.4, já que os parâmetros desviarão do comportamento esperado. A etapa de detecção é descrita pelo Algoritmo 2.

3.2 Combinação de detectores usando atributos individualmente

No método apresentado na Seção 3.1, cada estimativa $\{\widetilde{\theta}^{(1)}, \widetilde{\theta}^{(2)}, \widetilde{\theta}^{(3)}, \ldots\}$ é obtida realizando uma concatenação dos vetores de características de dois exemplos. Sendo assim, uma única estimativa é gerada considerando todo o conjunto de atributos. Apesar da vantagem de se projetar todo o conjunto de atributos em único espaço, tornando rápida a detecção de novas amostras, a etapa de treinamento pode ser custosa computacionalmente

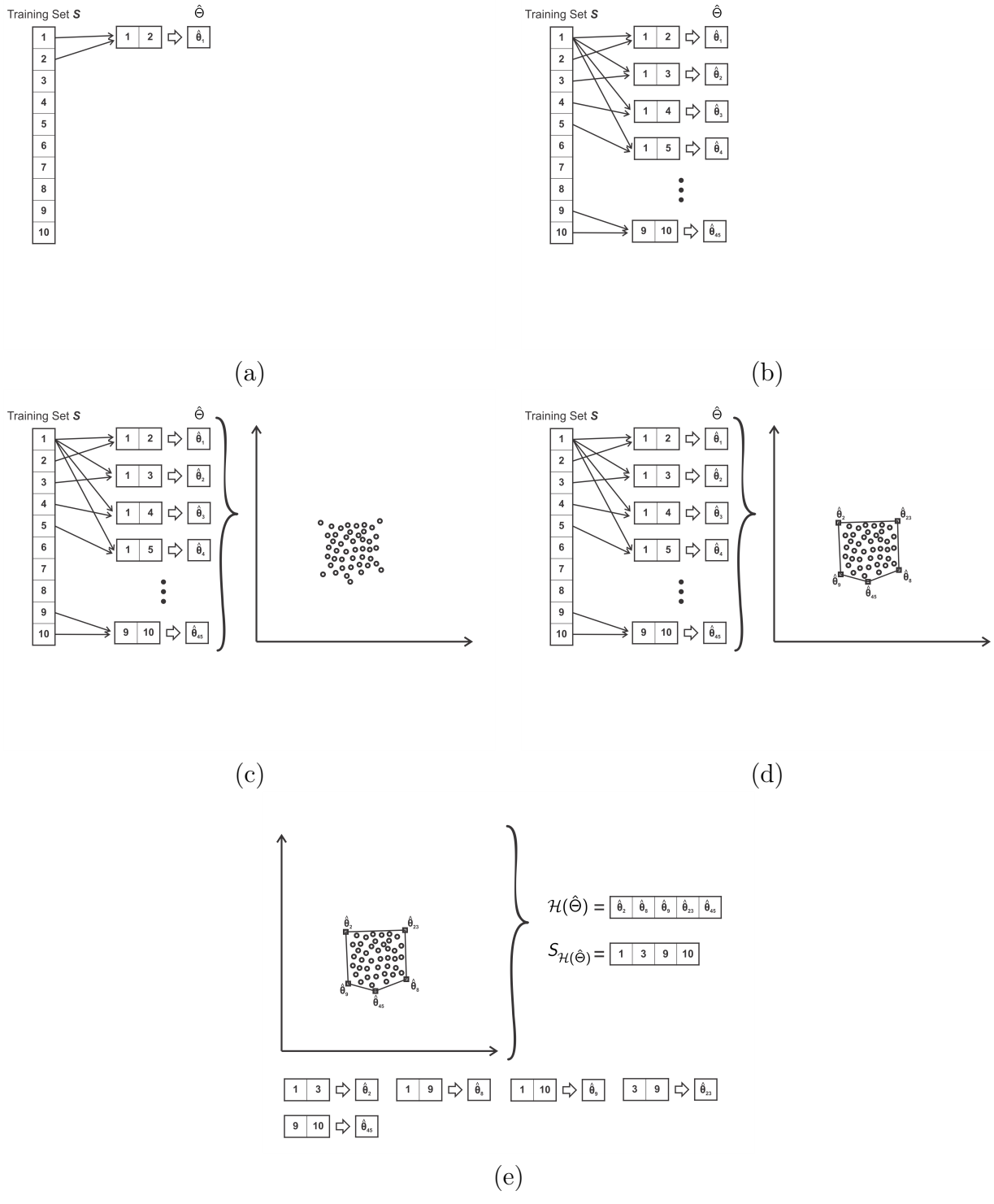


Figura 3.2: Exemplo de execução do algoritmo de treinamento. (a) e (b) ilustram o cálculo das estimativas a partir de pares de exemplos. (c) e (d) ilustram o cálculo do fecho convexo e (e), o resultado final da etapa de treinamento, composto pelo fecho convexo e os exemplos que contribuíram para as estimativas que compõem o fecho.

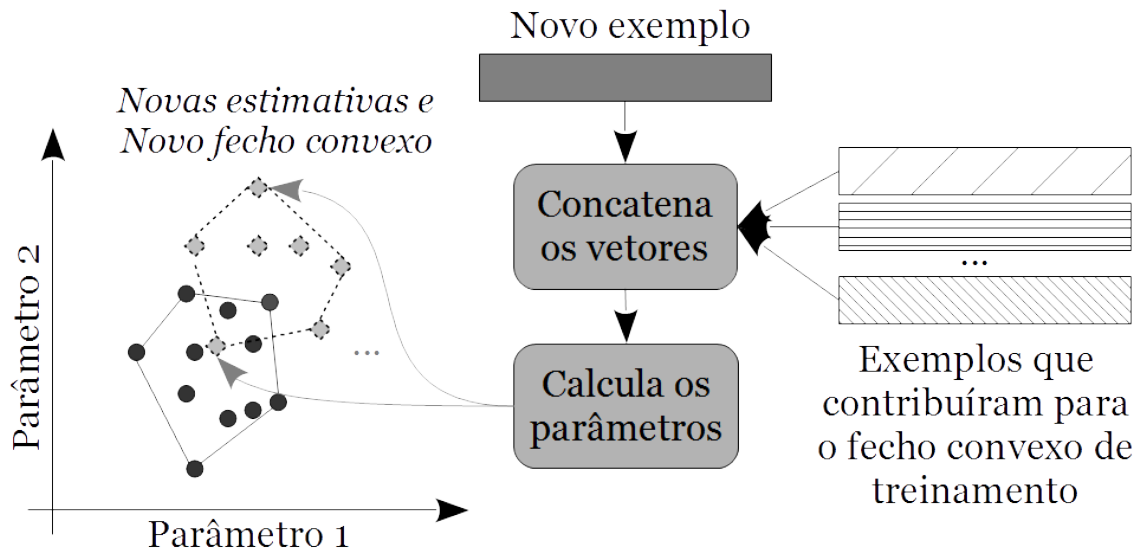


Figura 3.3: Geração de um novo fecho convexo através da combinação de um novo exemplo e dos exemplos que colaboraram para o fecho convexo da etapa de treinamento.

considerando que ela ocorre de forma sequencial. Além disso, o comportamento dos dados pode não ser muito bem representado utilizando somente um conjunto $\widehat{\Theta}^{(k)}$.

Para obter uma versão paralela do algoritmo de treinamento, foi definida uma segunda abordagem na qual calcula-se um conjunto $\widehat{\Theta}^{(k,i)}$ para cada atributo i. Consequentemente, cada atributo terá seu próprio fecho convexo $\mathcal{H}(\widehat{\Theta}^{(k,i)})$.

Essa abordagem pode ser considerada uma combinação de classificadores, já que ela utiliza um detector de anomalias por atributo. O resultado final é obtido através de uma fusão dos resultados utilizando, por exemplo, uma votação. A etapa de fusão deve ser realizada sequencialmente, enquanto as demais etapas, treinamento e detecção, podem ocorrer de forma paralela. Desta forma, espera-se que o método proposto consiga trabalhar adequadamente com bases de dados com atributos que possuam máximos e mínimos muito diferentes entre si. Para diferenciar do algoritmo citado anteriormente e facilitar a compreensão, esse algoritmo será chamado *Convex Hull Fusion Anomaly Detector* (CHF-AD).

3.3 Conjuntos de Dados

Para a verificar a eficiência dos métodos propostos é necessária a realização de testes e avaliação através das medidas apresentadas na Seção 2.4. Para isso, foram pesquisadas e selecionadas algumas bases de dados com as características de problemas de detecção de anomalias. Dentre as bases de dados sugeridas para serem usadas nesse trabalho existem

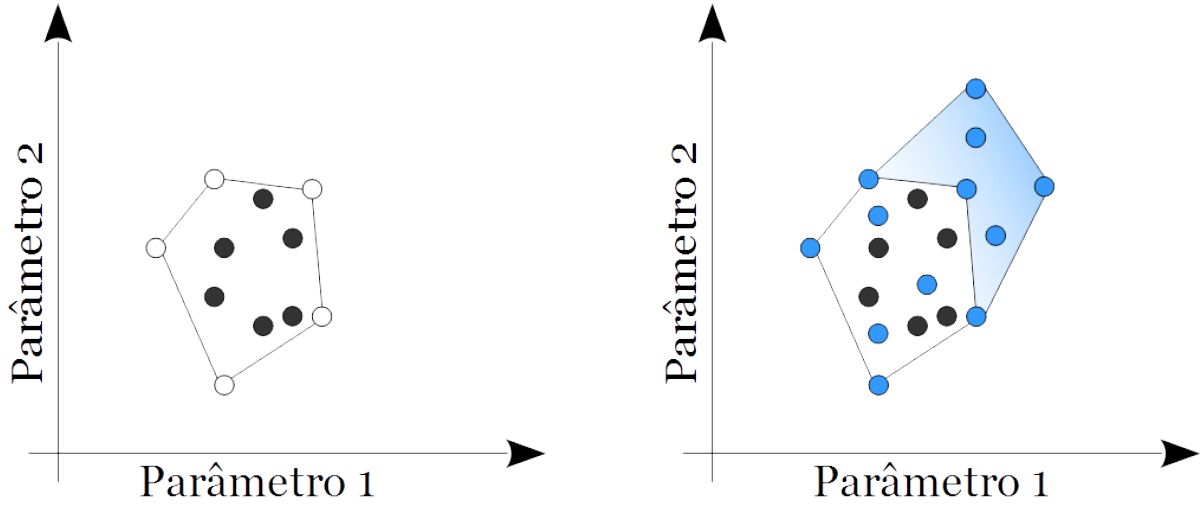


Figura 3.4: A intersecção entre o fecho convexo original (etapa de treinamento) e o novo fecho convexo (etapa de detecção) é utilizada para realizar a classificação dos novos exemplos. Nessa figura, o novo exemplo é uma anomalia. Os pontos azuis representam as estimativas calculadas a partir da concatenação da anomalia com os exemplos que contribuíram para o fecho convexo original.

bases com dados sintéticos e reais, sendo que as reais podem ser divididas entre bases obtidas a partir de sinais ou de imagens. Informações sobre as bases de dados podem ser observadas na Tabela 3.1, sendo a taxa de anomalias uma estimativa da proporção entre a quantidade de dados considerados anômalos e reais na base de dados. Estes conjuntos de dados pertencem a diversas áreas de aplicação, sendo a maioria de aplicação médica.

3.3.1 Conjutos de Dados Sintéticos

- **Gaussian vs. 2 distribuições:** para essa base de dados, cem amostras foram geradas aleatoriamente a partir de uma distribuição gaussiana para compor a classe normal. Para as anomalias utilizou-se seis amostras aleatórias geradas a partir de uma distribuição *Lithuanian* proposta por Raudys [26] e seis amostras também aleatórias de uma distribuição *Banana-shaped*, conforme descrita por Kuncheva [26].

- **Banana vs. 2 distribuições:** a classe normal foi composta por 500 exemplos aleatórios obtidos através da distribuição *Banana-shaped*. Para as anomalias, utilizou-se uma amostra aleatória de uma distribuição Gaussiana com 50 exemplos e uma amostra aleatória seguindo outra distribuição *Banana-shaped* também com 50 exemplos.

Algoritmo 2 Etapa de detecção do algoritmo CH-AD

Requer: Conjunto de exemplos que contribuíram para cada $\mathcal{H}(\widehat{\Theta}^{(k)})$ e o conjunto de limiares de detecção $T^{(k)}$ obtido através do conjunto de validação

para cada novo exemplo x **faça**
  Estimar $\widetilde{\theta}^{(1)}, \widetilde{\theta}^{(2)}, \widetilde{\theta}^{(3)}, \ldots$ com:
  para cada par de exemplos $y_1, y_2 \in \mathcal{H}(\widehat{\Theta}^{(k)}) \cup \{x\}$ **faça**
    $c_i \leftarrow$ concatenação de y_1, y_2
    Estimar cada parâmetro $\theta(i)$ a partir de c_i
  fim para
  para cada uma das k combinações diferentes de parâmetros **faça**
    Calcular $\mathcal{H}(\widetilde{\Theta}^{(k)})$
    $I \leftarrow (\mathcal{H}(\widehat{\Theta}^{(k)}) \cap \mathcal{H}(\widetilde{\Theta}^{(k)}))$
    $d \leftarrow (\mathcal{H}(\widehat{\Theta}^{(k)}) \setminus I) + (\mathcal{H}(\widetilde{\Theta}^{(k)}) \setminus I)$
    se $d < T$ **então**
      x é considerado normal
    senão
      x é considerado uma anomalia
    fim se
  fim para
  Retornar a classe votada pela maioria dos k detectores
fim para

- **Difficult 2 anomalies:** ambas classes seguem distribuições Gaussianas multivariadas. A classe normal é composta por 500 exemplos aleatórios. Foram geradas 50 anomalias, sendo estas divididas entre duas distribuições diferentes, fazendo com que haja exemplos acima e abaixo da classe normal.

- **Difficult 2 normals:** como a base de dados Difficult 2 anomalies, mas com as anomalias ocorrendo entre duas distribuições que representam a classe normal. Essa base de dados é composta por 525 exemplos sendo que 500 pertencem a classe normal e 25 são anomalias.

3.3.2 Conjuntos de Dados Reais

- **BreastR:** base de dados obtida através de imagens de mamografia para detecção de câncer de mama. Fornecida pelo grupo *Signal and Image Measurements and Processing (SIMP)* da Universidade de Roma "*Tor Vergata*". Essa base contém 122 casos que apresentaram tumores benignos e 25 casos com tumores malignos. Cada amostra possui 12 atributos que foram obtidos a partir da aplicação de *wavelets Gabor* em imagens de mamografia [38].

Tabela 3.1: Características das Bases de Dados

Dataset	Tipo	#Amostras	#Atributos	Taxa de anomalias
Gaussian-vs-2	sintético	112	2	10.7%
Banana-vs-2	sintético	600	2	16.7%
Difficult Two Anomalies	sintético	550	2	9.1%
Difficult Two Normals	sintético	525	2	4.8%
BreastR	real	147	12	17.0%
BreastW	real	698	9	34.5%
Ionosphere	real	351	32	35.9%
Parkinsons	real	193	22	23.8%
Produce Anomalies	real	289	128	8.7%
Green Coverage Texture	real	55	6	27.3%
Green Coverage Color	real	55	47	27.3%
Page-Blocks	real	5471	10	10.2%
Pima	real	768	8	34.9%
Landsat Sattelite 1	real	4435	36	30.8%
Landsat Sattelite 2	real	2000	36	33.6%

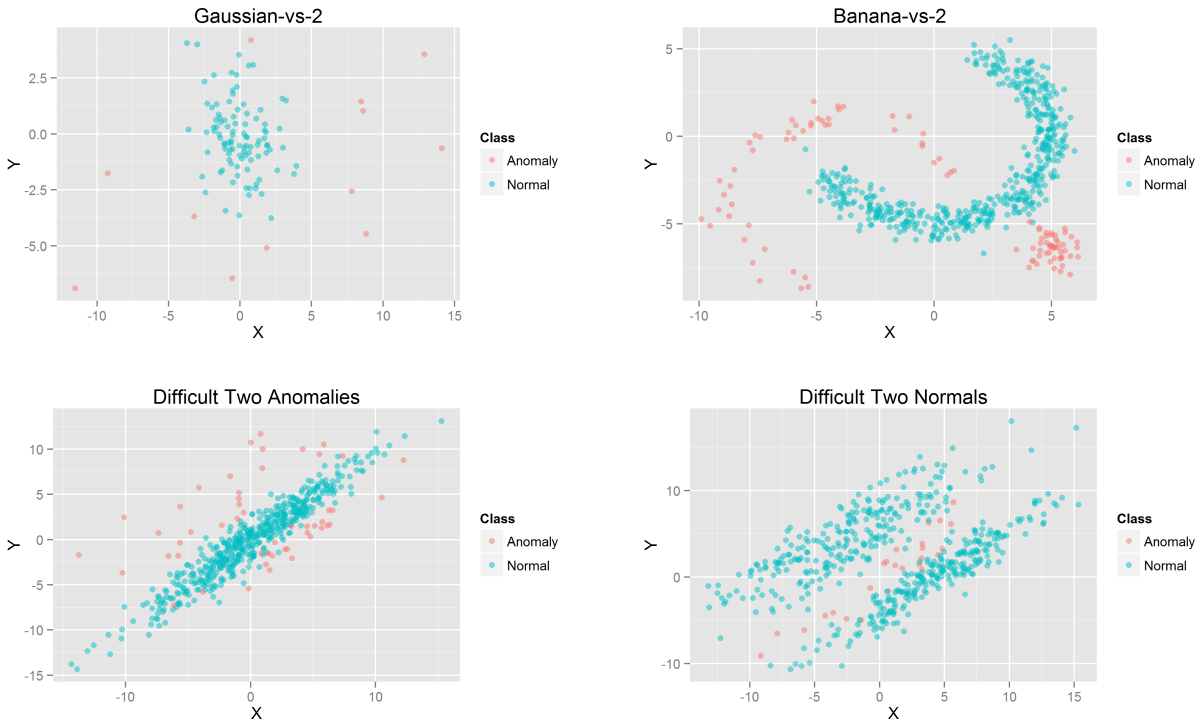


Figura 3.5: Gráficos de dispersão de cada uma das bases de dados sintéticas. Os exemplos em vermelho pertencem à classe anômala e os em azul à classe normal.

- **BreastW:** base de dados *Wisconsin Breast Cancer* obtida no repositório *UCI Machine Learning Repository* [11]. Essa base foi criada pelo Dr. William H. Holberg [34] no hospital da Universidade de Wisconsin. Ela contém 698 amostras com nove atributos inteiros cada. Destas, 457 foram classificadas como tumor benigno (classe normal) e 241 como tumor maligno (anomalias).

- **Ionosphere:** base de dados *Ionosphere* obtida no repositório *UCI Machine Learning Repository* [11]. Dados coletados por radares em um sistema em Labrador, formado por um *array* de 16 antenas de alta frequência transmitindo à $6,4kW$. Os dados que demonstraram evidências de existência de algum tipo de estrutura na ionosfera são considerados normais. Aqueles que passaram pela ionosfera e, portanto, não detectaram nenhuma estrutura, são anomalias [51]. Essa base de dados é composta por 351 amostras com 32 atributos contínuos cada, sendo que 225 amostras pertencem à classe normal e 126 são anomalias.

- **Parkinsons:** base de dados *Parkinsons* obtida no repositório *UCI Machine Learning Repository* [11]. Essa base de dados foi criada por Max Little da Universidade de Oxford e o Centro Nacional de Voz e Fala de Denver, Colorado, que gravaram áudio com fala de trinta e um pacientes. No estudo original foram publicados métodos de extração de atributos para identificação de distúrbios na fala por medições biomédicas de voz [32]. Dentre as 193 amostras que compõe essa base de dados, 147 são normais, pertencentes a pacientes saudáveis, e 46 são anomalias, pertencentes a pacientes com a doença de Parkinson.

- **Produce Anomalies:** essa base de dados é um subconjunto da base *Produce* [45] e é composta por 265 fotos de ameixas e 26 fotos de outras frutas processadas pelo mesma empresa distribuidora de frutas. As imagens foram obtidas sob iluminação artificial variada e com diversos fundos. O método de vetor de coerência de cores (CCV, do inglês *color coherence vectors*) [41] foi utilizado para extrair 64 atributos baseados em cor. Esse método cria histogramas: um para as cores pertencentes a regiões incoerentes, ou seja, regiões de pixels com a mesma cor contendo até T pixels, e o outro para as cores pertencentes às regiões coerentes com mais de T pixels. Para classificar as regiões entre coerentes e não-coerentes foi utilizado $T = 25$. Esse valor depende da resolução da imagem e, por esse motivo, foi encontrado experimentalmente. Algumas figuras pertencentes a essa base de dados podem ser vistas na Figura 3.6.

- **Green Coverage:** essa base da dados é um subconjunto de uma base de dados de sensoreamento remoto de baixo custo [44] formada por imagens suborbitais de uma

(a) (b)

Figura 3.6: Exemplos de imagens da base de dados *Produce Anomalies*. Coluna (a): normal; coluna (b): anomalias

lavoura de feijão. As imagens foram obtidas por um sistema de captura de imagens (canais RGB e infra-vermelho) anexado a um balão. Cada exemplo da base de dados é formado por uma sub-imagens de 100×100 pixels.

Esses exemplos são consideradas normais quando possuem cobertura verde completa (40 amostras) e anormais quando existem falhas nessa cobertura ou a presença de plantas desidratadas (15 amostras). As classes foram definidas por um agrônomo.

Antes da extração dos atributos, os canais foram convertidos para imagens de 8 bits. Os atributos da base *Green Coverage Texture* foram adquiridos utilizando seis atributos de Haralick [18] com matriz de co-ocorrência (0,1): entropia, probabilidade máxima, homogeneidade, uniformidade, contraste e correlação. Já os atributos da base *Green Coverage Color*, foram obtidos através da aplicação, na mesma base de imagens, do extrator de características *Color Coherence Vector* (CCV), que gera dois histogramas conforme a classificação de cada pixel, coerente ou incoerente [40]. Um pixel é considerado coerente caso faça parte de um componente conexo grande e, incoerente, caso contrário. Os requisitos para que um componente convexo seja considerado grande são definidos pelo usuário. Exemplos de imagens pertencentes a essa base de dados podem ser vistos na Figura 3.7.

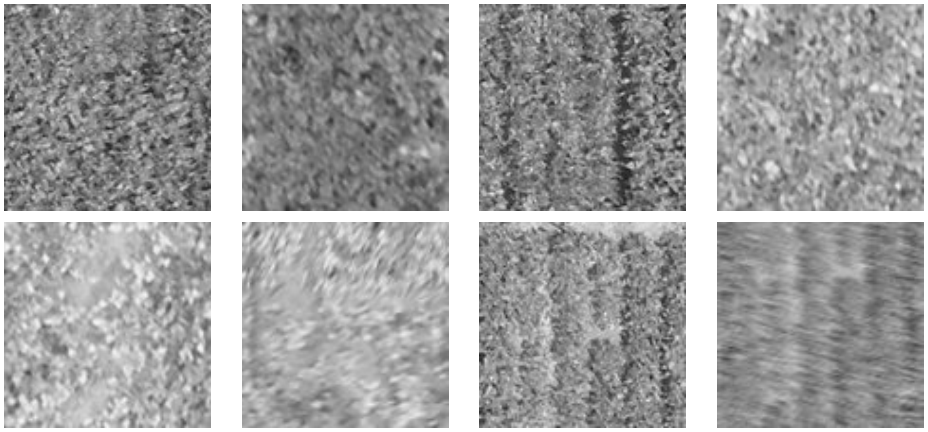


Figura 3.7: Exemplos de imagens da base de dados *Green Coverage*. Primeira linha: exemplos normais. Segunda linha: anomalias.

- **Page-Blocks:** base de dados *Page Blocks Classification* obtida no repositório *UCI Machine Learning Repository* [11]. Os 5471 exemplos foram obtidos de 54 documentos distintos. Cada exemplo consiste de um bloco de texto. O problema proposto é a classificação de todos os blocos do *layout* de uma página de um documento que for detectado através de um processo de segmentação. Todos os atributos são numéricos: altura, comprimento e área de um bloco, porcentagem de pixeis pretos

dentro do bloco, porcentagem de pixeis pretos após aplicação de um algoritmo *Run Length Smoothing (RLSA)*, número médio de transições branco-preto, número total de pixeis pretos no *bitmap* original do bloco, número total de pixeis pretos no *bitmap* do bloco após a aplicação do RLSA e número de transações branco-preto no *bitmap* original do bloco.

- **Pima:** base de dados *Pima Indians Diabetes* obtida no repositório *UCI Machine Learning Repository* [11]. Disponibilizada pelo *National Institute of Diabetes and Digestive and Kidney Diseases*. A seleção dos exemplos foi feita a partir da aplicação de várias restrições em uma base de dados maior. Em particular, todos os exemplos desse conjunto de dados foram obtidos de mulhers com pelo menos 21 anos da *Pima Indian heritage*. Cada exemplo contêm oito características: número de gravidezes, concentração plasmática de glicose após duas horas em um teste de tolerância à glicose, pressão arterial diastólica ($mm{\cdot}Hg$), espessura de dobras cutâneas tricipitais (mm), concentração sérica de glicose após duas horas em um teste de tolerância a glicose ($\frac{mu{\cdot}U}{ml}$), índice de massa corpórea ($\frac{weight\,in\,kg}{(height\,in\,m)^2}$), *Diabetes Pedigree Function* e idade (anos). Sendo que o *Diabetes Pedigree Function (DPF)* foi desenvolvido por Smith et al. [52] para cálculo de uma medida síntese do risco de diabetes devido à influência genética esperada de parentes afetados e não afetados por meio de informações de pais, avós, irmãos, tios e tias e primos de primeiro grau.

- **Landsat Sattelite:** base de dados *Statlog (Landsat Satellite)* obtida no repositório *UCI Machine Learning Repository* [11]. A base de dados consiste de valores multi-espectrais de pixeis em uma vizinhança 3×3 em imagens de satélite e a classificação associada com o pixel central em cada vizinhança. O objetivo é predizer a classificação, dado os valores multi-espectrais.

  Na amostra que compõe a base de dados , a classe de um pixel é codificada por um número. A base de dados original possuí sete classes: (1) *red soil*, (2) *cotton crop*, (3) *grey soil*, (4) *damp grey soil*, (5) *soil with vegetation stubble*, (6) *mixture class (all types present)*, (7) *very damp grey soil*. Não existem exemplos da sexta classe nessa base dedados. As outras seis foram dividias em dois grupos de forma a gerar um problema de detecção de anomalias. As três classe com menor número de amostras (2, 4 e 5) foram tratas como anomalias. Essa divisão foi sugerida em um artigo que realiza o *benchmark* de algoritmos de detecção de anomalias para grandes bases de dados [6].

  Esse conjunto também já está divido para realização de treinamento e testes. O arquivo que possui amostras separadas para realização do treinamento é utilizado

para criação dos conjuntos de treinamento e validação, seguindo as proporções pré-estabelecidas. O conjunto de teste é composto por todo o arquivo com amostras para teste.

3.4 Considerações Finais

No próximo capítulo são apresentados os experimentos realizados para comparar o desempenho dos métodos apresentados, CH-AD e CHF-AD, ao desempenho obtido por alguns métodos já conhecidos, explicados no Capítulo 2: métodos estatístico paramétrico usando as distribuições Gaussiana univariada (Gaussian-U) e multivariada (Gaussian-M), *Naive* Bayes, *One-Class SVM* (OC-SVM). Também são realizadas comparações dos resultados obtidos por diferentes versões dos métodos CH-AD e CHF-AD, utilizando algoritmos de treinamento quadráticos e lineares.

Capítulo

4

Resultados

Neste capítulo são apresentados os experimentos realizados e os seus resultados. Os experimentos têm como objetivo comparar os métodos propostos com métodos já existentes utilizados para a detecção de anomalias. Para comparação, foi utilizada a média e o desvio padrão da acurácia balanceada. Essas medidas foram obtidas através de repetidas execuções dos métodos para cada base de dados. Em cada execução os conjuntos de treinamento, validação e teste foram selecionados aleatoriamente.

Todos os experimentos foram feitos utilizando validação por subamostragem aleatória, dividindo a base de dados em conjuntos de treinamento, validação e teste. Esses conjuntos foram utilizados de diferentes modos dependendo do método testado. A subamostragem aleatória, também conhecida como validação cruzada de Monte Carlo ou *multiple hold-out* [42], mostrou ser assintoticamente consistente, resultando em predições mais pessimistas quando comparado à validação cruzada [48].

Os métodos utilizados para comparação foram: os métodos estatísticos paramétricos baseados na distribuição normal univariada (Gaussian-U) e multivariada (Gaussian-M), o método *Naive Bayes* e o método *one-class SVM* (OC-SVM). Para realizar os testes com o *one-class SVM* foram realizadas buscas em *grid* para definir os melhores valores para cada parâmetro: *kernel*, custo, ν e γ.

Quando necessária a utilização de um limiar para definir se um novo exemplo é uma anomalia, esse limiar foi obtido através do conjunto de validação. A amostra contida nesse

conjunto não foi utilizada durante o treinamento ou os testes, apenas o limiar encontrado para definir a classificação dos exemplos é utilizado durante os testes.

Para esse trabalho, foram realizados três experimentos distintos. No primeiro experimento, buscou-se descobrir qual o número de pares necessários na construção dos espaços de parâmetros para se obter uma boa taxa de detecção das anomalias. O espaço de parâmetros utilizado para realizar esses testes foi o obtido pela combinação da média e do desvio padrão.

No segundo experimento foram realizadas dez execuções para cada combinação entre algoritmo de classificação e base de dados. Assim como no primeiro experimento, o espaço de parâmetros utilizado foi composto pelos parâmetros da distribuição Gaussiana, média e desvio padrão, e dois algoritmos que usam espaços de parâmetros para realizar a detecção de anomalias foram testados (CH-AD, descrito na Seção 3.1, e CHF-AD, descrito na Seção 3.2), ambos com treinamento de ordem linear com relação ao tamanho do conjunto de treinamento.

Já no terceiro experimento, foram realizadas cem repetições e obtidos a média e o desvio padrão da acurácia balanceada. Em cima desses resultados, foram realizados testes para verificar a significância estatística das diferenças observadas entre as médias.

4.1 Experimento 1

Como citado no Capítulo 3, os algoritmos CH-AD e CHF-AD foram adaptados para que seu treinamento fosse realizado com uma complexidade linear, essas adaptações foram chamadas CH-AD (2) e CHF-AD (2). Todos os algoritmos (com treinamento quadrático e linear) foram testados e os resultados obtidos podem ser vistos na Tabela 4.1. Nas adaptações lineares dos algoritmos, ao invés de se considerar todos os possíveis pares de amostras do conjunto de treinamento, utiliza-se N pares (sendo N o número de amostras no conjunto de treinamento) selecionados de forma aleatória. Essa alteração diminui a complexidade do algoritmo de treinamento de quadrático para linear com relação ao número de amostras no conjunto de treinamento.

Na maioria dos experimentos realizados, os resultados apresentados pelos algoritmos CH-AD (2) e CHF-AD (2) foram próximos ou melhores do que os apresentados pelos algoritmos quadráticos. O único caso em que o algoritmo linear apresentou um desempenho significativamente inferior ao quadrático (observado por sobreposição de desvio padrão) foi para a base de dados Ionosphere utilizando o algoritmo CHF-AD. Acredita-se que isso ocorreu devido a menor representatividade individual do comportamento normal, ou seja, apesar das amostras utilizadas para o treinamento representarem bem o comportamento

normal quando utilizadas em grande quantidade, isso não se repete com o número de amostras reduzido.

Os desvios padrão observados nos testes realizados pelos algoritmos lineares foi maior. Isso ocorreu devido à aleatoriedade inserida ao selecionar os N pares e à representatividade, com relação ao comportamento normal, da amostra selecionada.

Tabela 4.1: Resultados - Acurácia Balanceada (Média e Desvio Padrão)

Dataset	CH-AD	CH-AD (2)	CHF-AD	CHF-AD (2)
Gaussian-vs-2	93.9±1.5%	92.5±4.9%	94.2±5.3%	92.3±6.8%
Ionosphere	71.0±2.7%	69.3±5.9%	**88.7**±4.4%	72.5±3.1%
Parkinsons	65.5±5.6%	63.1±8.7%	68.1±4.1%	66.9±6.9%
BreastR	52.1±4.9%	53.9±7.1%	58.2±3.9%	**64.9**±4.9%
BreastW	84.1±2.8%	**95.1**±3.6%	94.2±2.5%	**96.0**±1.4%

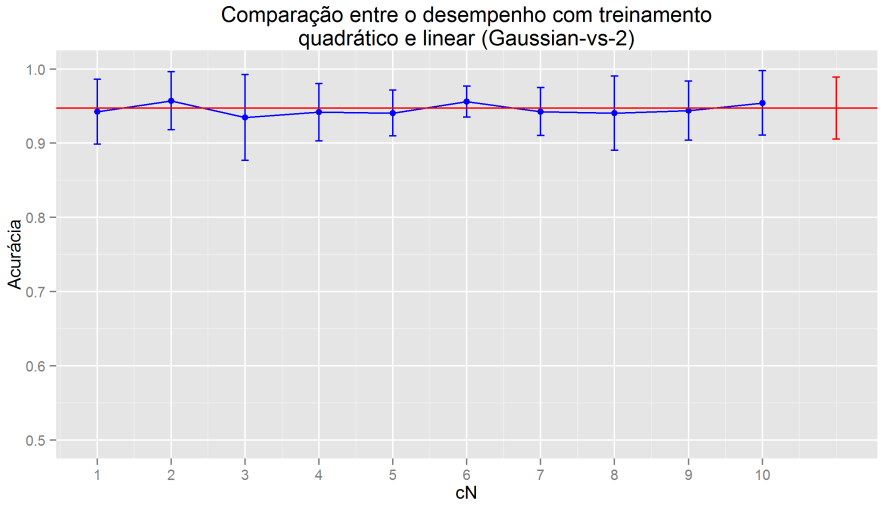
Uma possível melhoria nesse resultado pode ser obtida com o uso de $c \cdot N$ pares, sendo $c = 1, 2, \cdots, N$ uma constante definida como parâmetro do sistema. Foram feitos testes com diferentes valores de c para verificar o impacto na performance e se há diminuição do desvio padrão. Alguns gráficos obtidos através desses testes podem ser vistos na Figura 4.1. Após análise dos resultados obtidos, decidiu-se pela utilização de $2N$ pares. Apesar desse valor ter sido fixado para os experimentos apresentados nessa dissertação, permite-se que o usuário defina o valor que achar mais apropriado para sua aplicação.

4.2 Experimento 2

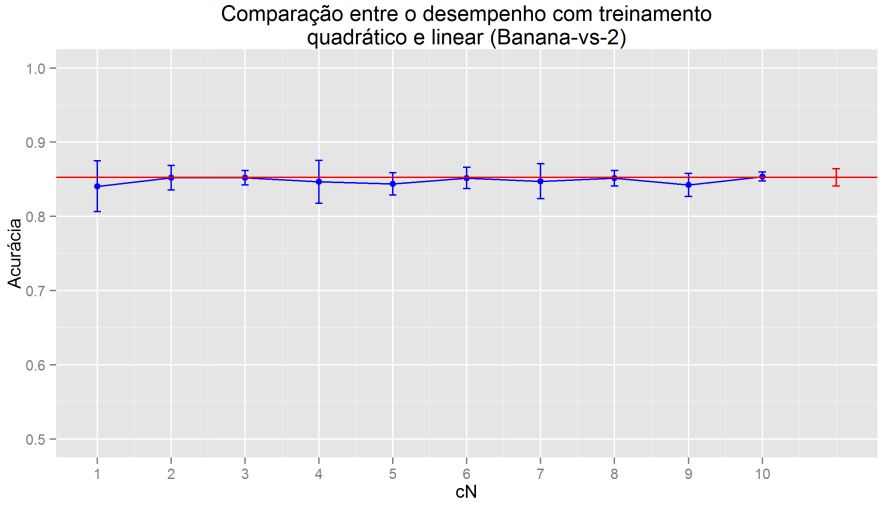
O segundo experimento realizado utilizou um único espaço de parâmetro de duas dimensões, constituído pelos parâmetros da distribuição Gaussiana, média e desvio padrão: $\widehat{\Theta}^{(1,2)} = (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$. Essa escolha foi feita devido a grande aplicabilidade da distribuição Gaussiana em problemas de classificação. Além disso, é possível comparar o método proposto com os métodos estatísticos paramétricos, que assumem essa mesma distribuição para os dados normais.

Para esse experimento, a subamostragem aleatória foi realizada dividindo-se a base de dados em conjuntos contendo 70%, 15% e 15% do seu número total de exemplos que foram utilizados para treinamento, validação e teste, respectivamente. Nos casos em que não é necessário utilizar um conjunto de validação, os exemplos que pertencem a esse conjunto foram incluídos no conjunto de treinamento.

Nesse experimento foram realizadas dez execuções com conjuntos de treinamento, validação e teste escolhidos de forma aleatória. A partir dessas execuções, foram calculados



(a)



(b)

Figura 4.1: Análise da progressão do desempenho (média e desvio padrão da acurácia balanceada) do algoritmo CH-AD conforme aumenta-se a constante c que define o número de pares $c \cdot N$ utilizados para gerar pontos nos espaços de parâmetros $\widehat{\Theta}$, onde N é o número de exemplos no conjunto de treinamento. A figura (a) foi obtida utilizando a base Gaussian-vs-2 e a figura (b), a base Banana-vs-2. A cor azul indica o desempenho obtido pelo algoritmo com treinamento linear e a cor vermelha, o desempenho do algoritmo com treinamento quadrático.

a média e o desvio padrão da acurácia balanceada, medida de avaliação descrita na Seção 2.4. Com essas medidas é possível comparar o desempenho geral do método e sua estabilidade em bases não balanceadas, como é o caso das bases utilizadas para detecção de anomalias, com métodos concorrentes.

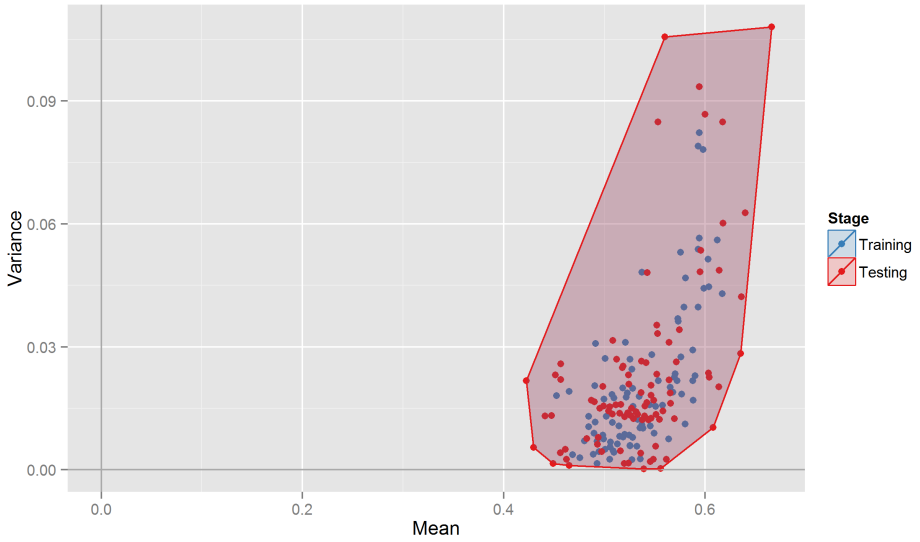
A média e o desvio padrão obtidos através de dez execuções dos métodos selecionados, para cada base de dados, podem ser observados na Tabela 4.2. Os resultados apresentados em negrito são os que demonstraram melhor desempenho considerando a comparação entre as médias e os desvios padrão. Alguns exemplos do espaço de parâmetros com os pontos e os fechos convexos gerados, conforme explicado na Seção 3.1, podem ser observados na Figura 4.2.

Tabela 4.2: Resultados - Acurácia Balanceada (Média e Desvio Padrão)

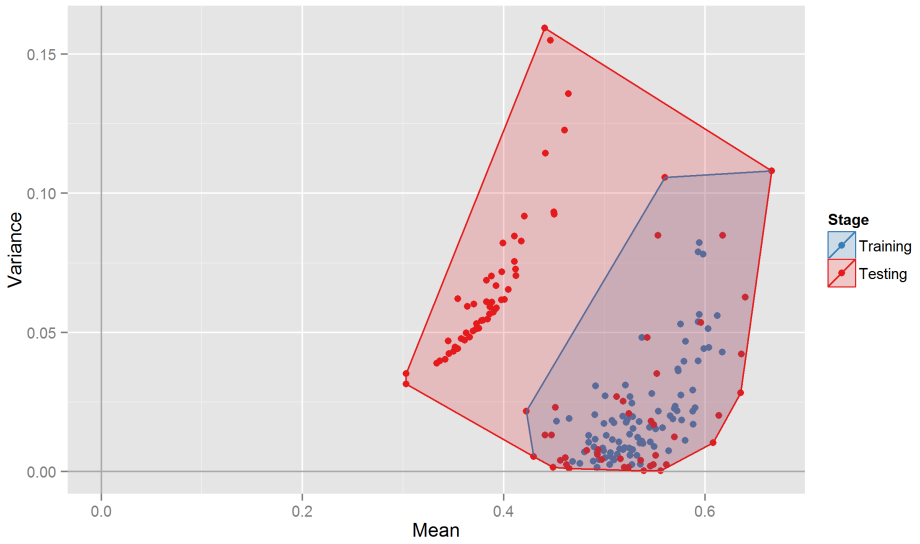
Dataset	Gaussian-U	Gaussian-M	Naive Bayes	OC-SVM	CH-AD	CHF-AD
Gaussian-vs-2	**93.8**±4.9%	**95.2**±1.4%	50.0±0.0%	80.1±0.1%	**93.9**±1.5%	**94.2**±5.3%
Ionosphere	71.2±5.2%	78.2±2.4%	78.5±8.6%	72.0±0.1%	77.2±2.7%	**89.7**±2.2%
Parkinsons	62.9±6.5%	63.9±5.2%	54.9±6.4%	67.3±0.2%	65.5±5.6%	**71.8**±3.9%
BreastR	67.2±7.5%	**70.1**±6.6%	50.0±0.0%	60.6±0.7%	52.1±2.9%	64.1±6.3%
BreastW	**94.1**±2.3%	92.6±2.0%	85.1±1.6%	90.8±0.5%	**93.1**±1.8%	**94.2**±2.5%

Os métodos propostos apresentaram bons resultados, comparáveis ou melhores do que os métodos utilizados para comparação em todas as bases de dados testadas com exceção da base BreastR. O desempenho apresentado para essa base de dados deve-se, provavelmente, à sobreposição entre as amostras das classes normal e anômala. Essa característica se manteve ao utilizar o espaço de parâmetros da distribuição gaussiana e assim o fecho convexo não foi capaz de realizar uma boa separação entre as classes. Entretanto, é possível que o método apresente resultados significativamente melhores para essa base de dados com a escolha de um outro espaço de parâmetros que realize uma melhor separação entre as classes. Além disso, os resultados apresentados para as demais bases demonstraram que utilizar um espaço de parâmetros pode auxiliar na detecção de anomalias.

Dentre os métodos competidores, os métodos supervisionados sofreram com a escassez de amostras da classe anômala, desta forma classificando de modo errado tais exemplos presentes no conjunto de teste. Os métodos estatísticos paramétricos apresentaram resultados próximos aos obtidos pelos métodos propostos. Entretanto, esses métodos apresentaram dificuldades em bases com alta dimensionalidade, diferentemente dos métodos propostos, cujos resultados demonstraram certa robustez nesses casos. Isso fica evidente



(a)



(b)

Figura 4.2: Exemplos de fechos convexos no espaço de parâmetros $\widehat{\Theta}^{(1,2)} = (\mu, \sigma) \in \mathbb{R} \times \mathbb{R}_+$ obtidos utilizando a base de dados Gaussian-vs-2. A região azul denota $\mathcal{H}(\widehat{\Theta}^{(1,2)})$, fecho convexo obtido através do conjunto de treinamento, e a região vermelha $\mathcal{H}(\widetilde{\Theta}^{(1,2)})$, fecho convexo obtido utilizando uma nova amostra. A detecção de uma amostra normal pode ser observada em (a) e de uma anomalia em (b).

nos resultados obtidos nas bases Parkinson e Ionosphere que podem ser observados na Tabela 4.2.

A utilização da combinação dos detectores baseados em fechos convexos foi muito eficiente para as bases de dados com grandes quantidades de atributos: Produce Anomalies (64), Ionosphere (32) e Parkinsons (22). Para as demais bases de dados, o método CH-AD, que utiliza somente um fecho convexo, apresentou resultado muito próximo do obtido pela combinação de fechos convexos e necessita de menos processamento para realizar o treinamento e a detecção.

4.3 Experimento 3

Nesse experimento utilizou-se um conjunto de parâmetros composto pelos quatro primeiros momentos estatísticos: média, variância, obliquidade e curtose. Esses parâmetros foram escolhidos de forma a diminuir o viés Gaussiano introduzido no algoritmo ao se utilizar apenas os parâmetros média e variância. Para cada base de dados, modela-se apenas o comportamento da classe normal e não se realiza nenhuma suposição sobre o comportamento das anomalias.

Para combinar os quatro parâmetros escolhidos decidiu-se utilizar todas as 11 combinações possíveis, gerando 6 espaços de duas dimensões, 4 espaços de três dimensões e 1 espaço de quatro dimensões. A decisão final foi tomada através da fusão entre as saídas de todos os detectores. Essa fusão foi realizada utilizando uma votação.

Assim como nos demais experimentos, as bases de dados foram separadas em três conjuntos: treinamento, validação e teste. O conjunto de treinamento foi criado com 70% de todos os exemplos pertencentes a classe normal. Já o conjunto de validação contém 15% dos exemplos pertencentes a classe normal e 5% das anomalias. A quantidade de anomalias usada para o conjunto de validação foi definido utilizando a proporção de exemplos comumente disponível em bases de dados reais [7]. Os exemplos restantes foram incluídos no conjunto de testes.

Para o método *Naive* Bayes (método de classificação), que não utiliza conjunto de validação, os exemplos contidos no conjunto de validação foram incluídos no conjunto de treinamento, ficando assim com dois conjuntos: treinamento, contendo 85% dos exemplos normais e 5% as anomalias, e teste, com os demais exemplos.

Foram realizadas cem execuções para cada combinação entre método de detecção e base de dados. Dessas cem repetições, foram obtidas a média e o desvio padrão da acurácia balanceada (Seção 2.4). Esses resultados podem ser vistos na Tabela 4.3.

Tabela 4.3: Resultados - Acurácia Balanceada (Média e Desvio Padrão)

Base de dados	Gaussian-U	Gaussian-M	NaiveBayes	OC-SVM	CH-AD
Gaussian-vs-2	72.4±14.7%	74.1±14.9%	50.0±0.0%	78.3±5.4%	93.6±3.9%
Banana-vs-2	73.5±4.5%	85.6±5.9%	52.5±2.2%	66.6±9.7%	84.5±2.0%
Difficult 2 Anomalies	58.2±3.7%	83.1±6.5%	50.0±0.1%	63.1±6.5%	73.2±7.0%
Difficult 2 Normals	45.5±6.3%	44.5±5.7%	50.0±0.4%	68.8±2.2%	71.9±7.9%
Green Coverage Texture	61.3±9.1%	73.2±13.3%	50.0±0.0%	57.8±14.5%	72.2±9.2%
Green Coverage Color	50.0±0.0%	50.0±0.0%	50.0±0.0%	50.0±0.0%	78.2±12.4%
Produce Anomalies	50.0±0.0%	50.0±0.0%	72.7±14.1%	51.1±2.0%	77.1±5.0%
Pageblocks	85.7±1.0%	89.3±0.9%	72.6±4.2%	79.6±0.6%	73.9±3.2%
Pima	65.8±3.7%	65.9±3.5%	56.9±1.9%	58.0±1.8%	58.5±3.0%
Landsat Sattelite	50.0±0.0%	55.1±0.3%	73.1±1.3%	63.4±0.5%	97.5±0.3%

Para verificar a existência significância estatística nas diferenças encontradas nos resultados apresentados na Tabela 4.3, utilizou-se o teste de Friedman [13]. Esse teste é a variação não paramétrica semelhante à análise paramétrica de variância em duas vias (*two way analysis of variance*), cujo objetivo é determinar se é possível concluir, a partir de uma amostra de resultados, que há diferença entre o desempenho dos métodos testados.

O primeiro passo para realizar o teste estatístico é converter os resultados originais em rankings. Tais rankings variam de 1 a k, sendo k o número de métodos testados, para cada problema, separadamente, onde o método que apresentou o melhor desempenho receber o ranking 1, o segundo melhor, o ranking 2, e assim progressivamente.

O teste de Friedman precisa do resultado dos rankings médios de cada algoritmo, $R_j = \frac{1}{n}\sum_i r_i^j$, em que r_i^j é o ranking do j-ésimo de k algoritmos obtido quando aplicado na i-ésima de n bases de dados. Assumindo a hipótese nula, que afirma que todos os algoritmos tem comportamento similar e, assim, seus rankings R_j devem ser iguais, a estatística de Friedman:

$$\mathcal{X}_F^2 = \frac{12n}{k(k+1)}\left[\sum_j R_j^2 - \frac{k(k+1)^2}{4}\right]$$

é distribuída de acordo com $\mathcal{X}_F^2$ com $k-1$ graus de liberdade, quando n e k são grandes o suficiente. Para um número menor de algoritmos e bases de dados, os valores críticos exatos foram calculados [49]. Para esse trabalho, o valor é 9.280, considerando $k = 5$, $n = 10$ e $\alpha = 0.05$. Iman e Davenport [22] mostraram que a estatística de Friedman $\mathcal{X}_F^2$ apresenta um comportamento conservador e propuseram uma estatística melhor:

$$F_F = \frac{(n-1)\mathcal{X}_F^2}{n(k-1)\mathcal{X}_F^2}$$

que é distribuída de acordo com distribuição F com $k-1$ e $(k-1)(n-1)$ graus de liberdade. Os p-valores podem ser calculados utilizando aproximações normais [1]. Caso a hipótese nula seja rejeitada, pode-se prosseguir com os testes *post hoc*. Os rankings obtidos através do teste de Friedman e os p-valores obtidos seguindo a estatística de Friedman e de Iman-Davenport podem ser vistos nas Tabelas 4.4 e 4.5, respectivamente.

Tabela 4.4: Ranking médio dos algoritmos (*Friedman*)

Métodos	**Ranking**
Gaussian-U	3.5
Gaussian-M	2.5
Naive Bayes	4.05
OC-SVM	3.15
CH-AD	1.8

Tabela 4.5: p-valores dos testes

Teste	**p-valor**	**Hipótese nula** ($\alpha = 0,05$)
Friedman	0.01551902	rejeitada
Iman-Davenport	0.00897189	rejeitada

Além do teste *Aligned Friedman*, também foram calculadas estimativas de contraste baseadas em medianas entre os métodos testados, apresentadas na Tabela 4.6. Tais estimativas foram obtidas através da análise das diversas execuções dos algoritmos de classificação em diversas bases de dados, apresentadas anteriormente. O procedimento para cálculo das estimativas de contraste assume que as diferenças esperada entre o desempenho dos métodos são iguais para todas as bases de dados e que tais diferenças são consequências das diferenças entre os métodos [9].

Para encontrar as estimativas de contraste, calcula-se, para cada par dos k algoritmos testados no experimento, a diferença entre o desempenho entre os algoritmos para cada uma das n bases de dados, $D_{i(u,v)} = x_{iu} - x_{iv}$, sendo $i = 1, \cdots, n$, $u = 1, \cdots, k$, $v = 1, \cdots, k$ e $u < v$. Encontra-se, então, a mediana para cada um dos conjuntos de diferenças (estimadores não ajustados). Essa mediana é chamada Z_{uv}, onde $Z_{uv} = Z_{vu}$ e $Z_{uu} = 0$.

Então, calcula-se a média m_u de cada conjunto de estimadores não ajustados:

$$m_u = \frac{\sum_{j=1}^{k} Z_{uj}}{k}, u = 1, \cdots, k.$$

e estima-se o contraste $m_u - m_v$.

Tabela 4.6: Estimativa de contraste baseada em medianas

	Gaussian-U	Gaussian-M	Naive Bayes	OC-SVM	CH-AD
Gaussian-U	0,000	-0,04577	0,07763	-0,003525	-0,1553
Gaussian-M	0,04577	0,000	0,1234	0,04225	-0,1095
Naive Bayes	-0,07763	-0,1234	0,000	-0,08115	-0,2329
OC-SVM	0,003525	-0,04225	0,08115	0,000	-0,1517
CH-AD	0,1553	0,1095	0,2329	0,1517	0,000

Pode-se observar através das estimativas de contraste apresentadas na Tabela 4.6 que o método CH-AD obteve valores positivos quando comparado com todos os outros métodos, indicando que esse obteve o melhor desempenho, nesse experimento, dentre todos os métodos testados. O método Naive Bayes apresentou o pior desempenho geral e os métodos Gaussian-M e OC-SVM apresentaram desempenho superior ao Gaussian-U. Tais resultados ajudam na comparação dos métodos porém não fornecem uma probabilidade do erro associado com a rejeição da hipótese nula.

Para comparar o método proposto CH-AD aos demais métodos foi utilizado o teste *post hoc* de Holm. Nesse teste, obtêm-se p-valores para cada comparação com o método proposto. O p-valor obtido indica a probabilidade de se obter um resultado que seja, pelo menos, tão extremo quanto o observado, assumindo que a hipótese nula, de que não há diferença entre os métodos, é verdadeira. Quanto menor o p-valor obtido, mais forte são as evidências contra a hipótese nula. Na Tabela 4.7 podem ser vistos os resultados do teste de Holm obtidos para esse experimento, sendo que o procedimento de Holm rejeita as hipóteses que apresentaram p-valor ≤ 0.025.

Quando o p-valor é utilizado para múltiplas comparações, esse reflete o erro de probabilidade de uma dada comparação, sem levar em consideração as demais comparações pertencentes a mesma família. Por exemplo, caso seja realizada a comparação entre k métodos e em cada comparação o nível de significância seja α, a probabilidade de não se cometer um erro Tipo I em uma única comparação é de $(1 - \alpha)$. Sendo assim, a probabilidade de não cometer um erro Tipo I em $k - 1$ comparações é $(1 - \alpha)^{(k-1)}$. Consequentemente, a probabilidade de se cometer um ou mais erros Tipo I é $1-(1-\alpha)^{(k-1)}$. Considerando $\alpha = 0.05$ e $k = 10$ tem-se que a probabilidade de se cometer um ou mais erros Tipo I é 0.37, valor considerado alto [15].

Para ajustar o valor de α de forma a compensar as diversas comparações, o teste de Holm inicialmente ordena os p-valores $p_1, p_2, \cdots, p_{k-1}$ do menor para o maior, tal que $p_1 \leq p_2 \leq \cdots \leq p_{k-1}$, sendo k o número de métodos comparados, e $H_1, H_2, \cdots, H_{k-1}$ as hipóteses correspondentes. O procedimento de Holm rejeita de H_1 a H_{i-1}, se i for o menor inteiro tal que $p_i > \alpha/(k - i)$.

Tabela 4.7: Teste de Holm para $\alpha = 0.05$ (Friedman)

i	Algoritmo	$z = (R_0 - R_i)/SE$	p-valor	$\alpha/(k-i)$	Hipótese nula
1	Naive Bayes	3.18198051	0.00146272	0.0125	rejeitada
2	Gaussian-U	2.40416306	0.01620954	0.01666667	rejeitada
3	OC-SVM	1.90918831	0.05623780	0.025	não rejeitada
4	Gaussian-M	0.98994949	0.32219881	0.05	não rejeitada

Os resultados do teste de Holm indicaram que há diferença com significância estatística entre o método proposto CH-AD e os métodos Naive Bayes e Gaussian-U. Já com relação aos métodos OC-SVM e Gaussian-M, não foi possível confirmar que existe significância estatística nas diferenças apresentadas nos resultados apresentados na Tabela 4.3.

4.4 Experimento 4

No quarto experimento realizado, utilizou-se apenas os espaços de parâmetros como uma forma de projeção dos exemplos em um novo espaço, gerando assim uma nova base de dados. Nesse experimento, foi criada uma base de dados a partir da base Gaussian-vs-2: Gaussian-EP. Nesta nova base, todos os $\frac{N(N-1)}{2}$ pontos possíveis no espaço de parâmetros foram utilizados, sendo N o número total de exemplos da base original. O espaço de parâmetros utilizado é composto pelos quatro primeiros momentos estatísticos: média, variância, obliquidade e curtose.

Para analisar se a utilização de espaços de parâmetros auxilia na detecção de anomalias foi utilizado o método *One-Class SVM* (OC-SVM). Primeiramente, fixou-se o *kernel* linear e realizou-se uma busca em *grid* para os demais parâmetros (custo, ν e γ). Em um segundo teste, a busca em *grid* foi realizada para todos os parâmetros, incluindo o *kernel*. Foram realizadas 10 repetições, obtendo-se a média e o desvio padrão da acurácia balanceada. Os resultados obtidos podem ser vistos na Tabela 4.8

Tabela 4.8: Resultados - SVMs utilizando espaço de parâmetros

	Gaussian-vs-2	Gaussian-EP
Linear	59.5±6.1%	71.5±14.7%
Kernel	81.7±1.8%	82.6±0.4%

Os resultados obtidos mostraram uma melhoria considerável quando o *kernel* linear foi fixado e o espaço de parâmetros foi utilizado com todos os $\frac{N(N-1)}{2}$ pontos possíveis. Isso indica que melhor separadas no espaço de parâmetros, do que no espaço originial. Já para a busca em grid realizada em todos os parâmetros, o resultado foi estável. Assim, pode-se enteder que a geração do espaço de parâmetros tem efeito parecido ao efeito de

um kernel. No entanto, ao invés de aumentar a dimensionalidade, o espaço de parâmetros tem potêncial para diminuí-la. Tais resultados demonstram que um espaço de parâmetros pode auxiliar em um problema de detecção de anomalias, com uma possível redução da dimensão VC (dimensão Vapnik-Chervonenkis) [57], ainda que este trabalho não tenha comprovado tal redução.

4.5 Considerações Finais

Os métodos propostos apresentaram resultados semelhantes ou melhores do que os métodos concorrentes nos experimentos realizados e apresentados neste capítulo, dessa forma, demonstrando a viabilidade da utilização desses métodos para realizar a detecção de anomalias.

Além disso, os métodos propostos possibilitam a utilização de diversas distribuições ou conjunto de parâmetros, além da combinação de parâmetros, tornado-o adaptável às necessidades do usuário. Entretanto, é necessário que o espaço de parâmetros escolhido seja composto por pelo menos duas dimensões, para que seja possível calcular o fecho convexo. Espaços de parâmetros com alta dimensionalidade também podem causar problemas no cálculo do fecho convexo, uma vez que essa tarefa é computacionalmente custosa. Para esses casos é possível utilizar uma combinação de classificadores em subconjuntos do espaço de parâmetros e realizar uma fusão para obter o resultado da detecção, semelhante ao que foi apresentado na descrição do método CHF-AD (Sessão 3.2) para cada atributo.

Também deve-se considerar que os métodos apresentados nesse trabalho são semi-supervisionados e portanto não necessitam de exemplos da classe anômala para realizar o treinamento. Esse fator torna os métodos mais versáteis, possibilitando que sejam aplicados em situações onde não é possível obter amostras da classe anômala.

Capítulo

5

Conclusão

Nesta dissertação foi apresentado um novo espaço, chamado espaço de parâmetros, que busca auxiliar a realização de detecção de anomalias através da observação do relacionamento entre os exemplos, além de um *framework* que utiliza espaços de parâmetros para realizar tal detecção. Acredita-se que a utilização desses espaços pode facilitar a realização da detecção de anomalias, pois esses permitem observar variações na relação entre dois exemplos, ao invés de observar os exemplos individualmente. Além disso, tais espaços podem ser criados de acordo com a aplicação ou base de dados alvo, permitindo uma grande adaptabilidade. Uma análise da base de dados alvo pode ser feita de forma a identificar quais os parâmetros que melhor modelam o comportamento da classe normal para aquela aplicação.

O uso de fechos convexos permitiu definir um limite entre o comportamento normal e anômalo, sem que as regiões tenham que ser simétricas ou regulares, como explorado nos métodos estatísticos. Além disso, é possível paralelizar o algoritmo utilizando diversos detectores em diferentes espaços de parâmetros e combiná-los para obter um resultado final. Os métodos propostos trabalham de forma semi-supervisionada, não necessitando de exemplos da classe anômala para realizar o treinamento. Essa abordagem permite uma maior aplicabilidade dos métodos em situações em que o número de exemplos da classe anormal disponível é muito escasso ou inexistente.

A principal limitação imposta pela utilização de espaços de parâmetros é a necessidade de se utilizar pelo menos dois parâmetros diferentes para que seja possível realizar o cálculo

do fecho convexo. Além disso, esse cálculo pode ser muito custoso quando realizado em espaços de alta dimensionalidade. Para lidar com esse problema, sugere-se a utilização de múltiplos detectores em subespaços de menor dimensionalidade e a realização da fusão dos resultados obtidos.

Nos experimentos realizados durante o desenvolvimento deste trabalho, os métodos propostos, que utilizaram os espaços de parâmetros para realizar a detecção, apresentaram desempenho similar ou superior ao dos métodos concorrentes testados. O mesmo espaço de parâmetros foi utilizado para todas as bases dentro de um mesmo experimento. Acredita-se que o desempenho pode ser melhorado caso o espaço seja escolhido individualmente para modelar o comportamento normal de cada base de dados.

O método proposto possibilita diversas adaptações para potenciais aplicações de detecção de anomalias. Estudos podem ser feitos de forma a verificar quais os melhores parâmetros a serem utilizados conforme o comportamento da classe normal modelada. Além disso, pode ser interessante realizar uma adaptação dos métodos de forma a criar classificadores especialistas em regiões do espaço de características. Outras adaptações dos métodos possuem grande potencial, como a utilização de exemplos anômalos durante a etapa de treinamento.

Referências Bibliográficas

[1] Abramowitz, M., and Stegun, I. A. *Handbook of mathematical functions: with formulas, graphs, and mathematical tables*. No. 55. Courier Dover Publications, 1972.

[2] Ando, S. Clustering needles in a haystack: an information theoretic analysis of minority and outlier detection. In *Proc. 7th Int. Conf. Data Mining* (2007), pp. 13–22.

[3] Barnett, V., and Lewis, T. *Outliers in statistical data*. John Wiley & Sons, 1994.

[4] Boyd, S., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2009.

[5] Breve, F. A., Ponti-Junior, M. P., and Mascarenhas, N. D. Multilayer perceptron classifier combination for identification of materials on noisy soil science multispectral images. In *Computer Graphics and Image Processing, 2007. SIBGRAPI 2007. XX Brazilian Symposium on* (2007), IEEE, pp. 239–244.

[6] Carrasquilla, U. Benchmarking algorithms for detecting anomalies in large datasets. *MeasureIT, Nov* (2010), 1–16.

[7] Chandola, V., Banerjee, A., and Kumar, A. Anomaly detection: a survey. *ACM Computing Surveys 41*, 3 (2009), 15.

[8] Dasarathy, B. V. Nearest neighbor (NN) norms: NN pattern classification techniques.

[9] Doksum, K. Robust procedures for some linear models with one observation per cell. *The Annals of Mathematical Statistics* (1967), 878–883.

[10] Fawcett, T., and Provost, F. J. Activity monitoring: noticing interesting changes in behavior. In *Proc. 5th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (1999), pp. 53–62.

[11] Frank, A., and Asuncion, A. UCI machine learning repository, 2010.

[12] Freund, Y., and Schapire, R. E. Experiments with a new boosting algorithm. In *Proc. 13th International Conference on Machine Learning (ICML-96)* (1996), pp. 148–156.

[13] Friedman, M. The use of ranks to avoid the assumption of normality implicit in the analysis of variance. *Journal of the American Statistical Association 32*, 200 (1937), 675–701.

[14] Fumera, G., and Roli, F. A theoretical and experimental analysis of linear combiners for multiple classifier systems. *Pattern Analysis and Machine Intelligence, IEEE Transactions on 27*, 6 (2005), 942–956.

[15] García, S., Fernández, A., Luengo, J., and Herrera, F. Advanced nonparametric tests for multiple comparisons in the design of experiments in computational intelligence and data mining: Experimental analysis of power. *Information Sciences 180*, 10 (2010), 2044–2064.

[16] Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. Physiobank, physiotoolkit, and physionet: Components of a new research resource for complex physiologic signals. *Circulation 101*, 23 (2000), e215–e220.

[17] Gope, C., and Kehtarnavaz, N. Affine invariant comparison of point-sets using convex hulls and hausdorff distances. *Pattern Recognition 40*, 1 (2007), 309–320.

[18] Haralick, R. M., Shanmugam, K., and Dinstein, I. H. Textural features for image classification. *Systems, Man and Cybernetics, IEEE Transactions on*, 6 (1973), 610–621.

[19] Hawkins, D. *Identification of outliers*. Chapman and Hall, 1980.

[20] He, Z., Deng, S., Xu, X., and Huang, J. Z. A fast greedy algorithm for outlier mining. In *Advances in Knowledge Discovery and Data Mining*. Springer, 2006, pp. 567–576.

[21] Hodge, V. J., and Austin, J. A survey of outlier detection methodologies. *Artificial Intelligence Review 22*, 2 (2004), 85–126.

[22] Iman, R. L., and Davenport, J. M. Approximations of the critical region of the friedman statistic. *Communications in Statistics-Theory and Methods 9*, 6 (1980), 571–595.

[23] Jain, A. K., and Dubes, R. C. *Algorithms for clustering data*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA, 1988.

[24] Juszczak, P., Tax, D. M. J., Pekalska, E., and Duin, R. P. W. Minimum spanning tree based one-class classifier. *Neurocomputing 72*, 7–9 (2009), 1859–1969.

[25] Keogh, E., Lonardi, S., and Ratanamahatana, C. Towards parameter-free data mining. In *Proc. 10th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining* (2004), pp. 206–215.

[26] Kuncheva, L. Artificial data. *School of Informatics, University of Wales, Bangor* (1996).

[27] Kuncheva, L. *Combining pattern classifiers: methods and algorithms*. Wiley-Interscience, 2004.

[28] Lee, W., and Xiang, D. Information-theoretic measures for anomaly detection. In *Security and Privacy, 2001. S&P 2001. Proceedings. 2001 IEEE Symposium on* (2001), IEEE, pp. 130–143.

[29] Leung, K., and Leckie, C. Unsupervised anomaly detection in network intrusion detection using clusters. In *Proceedings of the Twenty-eighth Australasian conference on Computer Science-Volume 38* (2005), Australian Computer Society, Inc., pp. 333–342.

[30] Liao, Y., and Vemuri, V. R. Use of k-nearest neighbor classifier for intrusion detection. *Computers & Security 21*, 5 (2002), 439–448.

[31] Lin, J., Keogh, E., Fu, A., and Van Herle, H. Approximations to magic: Finding unusual medical time series. In *Computer-Based Medical Systems, 2005. Proceedings. 18th IEEE Symposium on* (2005), IEEE, pp. 329–334.

[32] Little, M., McSharry, P., Roberts, S., Costello, D., and Moroz, I. Exploiting nonlinear recurrence and fractal scaling properties for voice disorder detection. *BioMedical Engineering OnLine 6* (2007), 23.

[33] Majumdar, S. N., Comtet, A., and Randon-Furling, J. Random convex hulls and extreme value statistics. *J Stat Phys 138*, 6 (2010), 955–1009.

[34] Mangasarian, O. L., and Wolberg, W. H. Cancer diagnosis via linear programming. *SIAM News 23*, 5 (1990), 1–18.

[35] Markou, M., and Singh, S. Novelty detection: a review - part 2: neural network based approaches. *Signal processing 83*, 12 (2003), 2499–2521.

[36] Markou, M., and Singh, S. Novelty detection: a review part 1: statistical approaches. *Signal processing 83*, 12 (2003), 2481–2497.

[37] Meier, R., Ackermann, F., Herrmann, G., Posch, S., and Sagerer, G. Segmentation of molecular surfaces based on their convex hull. In *Int. Conf. on Image Processing (ICIP 95)* (1995), vol. 3, pp. 552–555.

[38] Mencattini, A., Salmeri, M., Casti, P., Pepe, M. L., Mangieri, F., and Ancona, A. Local active contour models and gabor wavelets for an optimal breast region segmentation. *Computer Assisted Radiology and Surgey (CARSÂ ´12)* (2012).

[39] Mukkamala, S., Janoski, G., and Sung, A. Intrusion detection using neural networks and support vector machines. In *Neural Networks, 2002. IJCNN'02. Proceedings of the 2002 International Joint Conference on* (2002), vol. 2, IEEE, pp. 1702–1707.

[40] Pass, G., and Zabih, R. Histogram refinement for content-based image retrieval. In *Applications of Computer Vision, 1996. WACV'96., Proceedings 3rd IEEE Workshop on* (1996), IEEE, pp. 96–102.

[41] Pass, G., Zabih, R., and Miller, J. Comparing images using color coherence vectors. In *ACM Multimedia 96* (1996), pp. 65–73.

[42] Picard, R., and Cook, R. Cross-validation of regression models. *Journal of the American Statistical Association 79*, 387 (1984), 575–583.

[43] Ponti, M., and Mascarenhas, N. Material analysis on noisy multispectral images using classifier combination. In *Image Analysis and Interpretation, 2004. 6th IEEE Southwest Symposium on* (2004), IEEE, pp. 1–5.

[44] Ponti, M. P. Segmentation of low-cost remote sensing images combining vegetation indices and mean shift. *Geoscience and Remote Sensing Letters, IEEE 10*, 1 (2013), 67–70.

[45] Rocha, A., Hauagge, D. C., Wainer, J., and Goldenstein, S. Automatic fruit and vegetable classification from images. *Computers and Electronics in Agriculture 70*, 1 (2010), 96–104.

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

[47] Scholkopf, B., and Smola, A. J. *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001.

[48] Shao, J. Linear model selection by cross-validation. *Journal of the American Statistical Association 88*, 422 (1993), 486–494.

[49] Sheskin, D. J. *Handbook of parametric and nonparametric statistical procedures*. crc Press, 2003.

[50] Shewhart, W. A. Economic control of quality of manufactured product.

[51] Sigillito, V. G., Wing, S. P., Hutton, L. V., and Baker, K. B. Classification of radar returns from the ionosphere using neural networks. *Johns Hopkins APL Technical Digest 10* (1989), 262–266.

[52] Smith, J. W., Everhart, J., Dickson, W., Knowler, W., and Johannes, R. Using the adap learning algorithm to forecast the onset of diabetes mellitus. In *Proceedings of the symposium on computer applications and medical care* (1988), vol. 261, p. 265.

[53] Takahashi, T., Kudo, M., and Nakamura, A. Construction of convex hull classifiers in high dimensions. *Pattern Recognition Letters 32*, 16 (2011), 2224–2230.

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

[55] Tax, D., and Duin, R. Support vector data description. *Machine Learning 54*, 1 (2004), 45–66.

[56] Teng, H. S., Chen, K., and Lu, S. Adaptive real-time anomaly detection using inductively generated sequential patterns. In *Research in Security and Privacy, 1990. Proceedings., 1990 IEEE Computer Society Symposium on* (1990), IEEE, pp. 278–284.

[57] Vapnik, V. *The nature of statistical learning theory.* springer, 2000.

[58] Wolpert, D. H. The lack of a priori distinctions between learning algorithms. *Neural computation 8*, 7 (1996), 1341–1390.

[59] Wong, W., Moore, A., Cooper, G., and Wagner, M. Bayesian network anomaly pattern detection for disease outbreaks. In *Machine Learining International Workshop* (2003), vol. 20, p. 808.