UNIVERSIDADE FEDERAL DO CEARÁ
DEPARTAMENTO DE ENGENHARIA DE TELEINFORMÁTICA
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE TELEINFORMÁTICA

REWBENIO ARAÚJO FROTA

# AVALIAÇÃO DE ALGORITMOS DE REDES NEURAIS ARTIFICIAIS EM TAREFAS DE DETECÇÃO DE NOVIDADES: UMA ABORDAGEM UNIFICADORA

FORTALEZA
2005

**REWBENIO ARAÚJO FROTA**

# AVALIAÇÃO DE ALGORITMOS DE REDES NEURAIS ARTIFICIAIS EM TAREFAS DE DETECÇÃO DE NOVIDADES: UMA ABORDAGEM UNIFICADORA

Dissertação submetida à Coordenação do Curso de Pós-Graduação em Engenharia de Teleinformática, da Universidade Federal do Ceará, como parte dos requisitos exigidos para obtenção do grau de Mestre em Engenharia de Teleinformática.

Orientador: Prof. Dr. Guilherme de Alencar Barreto

Co-Orientador: Prof. Dr. João César Moura Mota

FORTALEZA

2005

F961a Frota, Rewbenio Araújo

Avaliação de algoritmos de redes neurais artificiais em tarefas de detecção de novidades: uma abordagem unificadora / Rewbenio Araújo Frota. 2005.

115f.,il.,enc.

Orientador: Prof. Dr. Guilherme Alencar Barreto;
Co-orientador: Prof. Dr. João César Moura Mota.
Dissertação (Mestrado) em Engenharia de Teleinformática – Universidade Federal do Ceará, Fortaleza, 2005.

1. Redes neurais 2. Detecção de novidades I. Título

C.D.D. 621.3
C.D.U. 621.3

**Rewbenio Araújo Frota**

# Avaliação de Algoritmos de Redes Neurais Artificiais em Tarefas de Detecção de Novidades: Uma Abordagem Unificadora

Esta Dissertação foi julgada adequada para a obtenção do título de Mestre em Engenharia de Teleinformática e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia de Teleinformática da Universidade Federal do Ceará.

Rewbenio Araújo Frota

Rewbenio Araújo Frota

Banca Examinadora:

Guilherme de Alencar Barreto

Prof. Guilherme de Alencar Barreto, Dr.

Prof. João César Moura Mota, Dr.

Profa. Fátima Nelsizeuma Sombra de Medeiros, Dra

Prof. Aluízio Fausto Ribeiro, Dr.

Fortaleza, 08 de julho de 2005

*Dedico este trabalho a minha mãe,*

*Antonia de Castro, pelo seu amor*

*e sua constante luta que tudo tornou possível.*

# *Agradecimentos*

A Deus, por trazer luz e verdade à minha vida.

Ao meu orientador e co-orientador, Prof. Dr. Guilherme de Alencar Barreto e Prof. Dr. João César Moura Mota, pela amizade que criamos, pela orientação segura e paciente nos momentos necessários.

Ao meu amigo Júlio Pimentel, pelo altruísmo que permitiu minha dedicação exclusiva aos estudos de mestrado.

Ao meu amigo Sanderson Lima, pelas discussões e sugestões sempre positivas.

Ao meu amigo Gustavo Souza, companheiro de laboratório, pela inestimável ajuda em diversos momentos.

A todos os professores e funcionários do Departamento de Engenharia de Teleinformática que de forma direta ou indireta participaram do desenvolvimento deste trabalho.

A minha família pelo apoio durante esta jornada.

A minha amada noiva Mayomi, que tem sido sempre um incentivo para que eu faça o melhor que posso.

À FUNCAP (Fundação Cearense de Amparo à Pesquisa) por ter custeado meus estudos de mestrado.

*"...não há nada de novo*
*debaixo do sol."*

**Eclesiastes** 1:9

# *Resumo*

Detecção de Novidades é uma tarefa de reconhecimento de padrões cujo objetivo está em anunciar a ocorrência de novos eventos, ou novas observações, a partir de um modelo estatístico pré-estabelecido para os dados observados. Por ser uma área de crescente interesse para os campos de aprendizado de máquinas e mineração de dados, uma vasta gama de métodos está disponível na literatura, dentre os quais podem ser citados os discriminantes estatísticos lineares e não-lineares, identificação recursiva de sistemas, sistemas especialistas, redes neurais artificiais, lógica nebulosa, computação evolucionária, dentre inúmeros outros. Contudo, poucos estudos analisando conjuntamente o desempenho dos vários métodos ora mencionados vem sendo realizados, fato este que pode ser explicado, pelo menos em parte, pela grande dificuldade de se comparar os diferentes paradigmas de modelagem existentes. Comparações de desempenho são importantes para avaliar, por exemplo, qual técnica funciona melhor em determinado tipo de dados, ou qual é menos sensível (mais robusta) à presença de observações discrepantes (*outliers*) nos dados modelados. Isto posto, nesta dissertação busca-se dar algumas contribuições à análise comparativa de técnicas de detecção de novidades, propondo uma metodologia geral para comparar o desempenho de sistemas de detecção de novidades usando redes neurais artificiais. A idéia subjacente à metodologia proposta consiste em utilizar os modelos estatísticos dos dados, obtidos pelos vários algoritmos de redes neurais estudados, para calcular limiares de decisão para os testes de detecção de novidades. Tais limiares são baseados no conceito de intervalo de confiança *bootstrap* não-paramétrico. Por fim, uma característica importante desta metodologia é que ela permite comparar sob um mesmo arcabouço teórico tanto algoritmos supervisionados, quanto não-supervisionados. Tal generalidade é ilustrada através de duas aplicações, a saber, detecção de anomalias em sistemas de comunicação sem fio de terceira geração e detecção de tumores em mamografias.

**Palavras-chave**: Detecção de Novidades, Detecção de Outliers, Detecção de Anomalias, Redes Neurais Artificiais.

# *Abstract*

Novelty detection is a pattern recognition task whose goal is to report the occurrence of novel events, or new observations, from a previously computed statistical model of the data. Being the focus of an increasing attention in machine learning and data mining fields, a wide range of methods for novelty detection are available in literature, among them it is possible to cite the linear and nonlinear statistical discriminants, recursive identification of systems, expert systems, artificial neural networks, fuzzy logic, evolutionary computing, among others. However, few works evaluating the performance of the methods just mentioned have been published. This fact can be, at least, partially explained by the inherent complexity of comparing different modeling methods. Performance comparisons are important to evaluate, for example, which technique works best on certain types of data or which one is less sensitive (more robust) to outliers present in the data used to build the model. Thus, the work developed in this dissertation attempts to contribute to the comparative analysis of novelty detection methods, proposing a general methodology to compare the performance of neural-based novelty detection systems. The rationale underlying the proposed methodology consists in using the statistical models for the data, obtained from various artificial neural networks, to compute decision thresholds for novelty detection tests. Such thresholds are based on the the concept of nonparametric *bootstrap* confidence intervals. Finally, regarding the new methodology, it is worth noting that it allows both supervised and unsupervised neural algorithms to be compared under a common framework. Such a generality is illustrated through two distinct applications, namely, anomaly detection in cellular systems and cancer detection in mammographies.

**Keywords**: Novelty Detection, Outlier Detection, Anomaly Detection, Artificial Neural Networks, Bootstrap Resampling.

# *Lista de Figuras*

# *Lista de Tabelas*

# *Lista de Siglas*

| | |
|---|---|
| 3G | Terceira Geração ($3^{rd}$ *Generation*) |
| AAMLP | Autoassociador MLP (*MLP autoassociator*) |
| BER | *Bit Error Rate* |
| CDMA | *Code Division Multiple Access* |
| EQ | Erro de Quantização |
| ER | Erro de Reconstrução |
| ERB | Estação Rádio-Base |
| FER | *Frame Error Rate* |
| FSCL | *Frequency Sensitive Competitive Learning* |
| GMLP | *Gaussian Multilayer Perceptron* |
| IQR | Intervalo interquartis (*interquartile range*) |
| MLP | *Multilayer Perceptron* |
| NGA | *Neural-Gas Algorithm* |
| OLAM | *Optimal Linear Autoassociative Memory* |
| RBF | *Radial Basis Function* |
| RNA | Redes Neurais Artificiais |
| SOM | *Self-Organizing Maps* |
| UM | Unidade Móvel |
| WTA | *Winner Take All* |

# *Lista de Símbolos*

| | |
|---|---|
| $\Re$ | conjunto dos números reais |
| $H_0$ | hipótese nula |
| $H_1$ | hipótese alternativa |
| $\alpha$ | nível de significância estatística (probabilidade de erro tipo I) |
| $\beta$ | probabilidade de erro tipo II |
| $N_\varrho$ | $100(1-\varrho)$ percentil |
| $p$ | valor-$p$ |
| $x$ | variável escalar, i.e., $x \in \Re$ |
| $\mathbf{x}$ | variável vetorial, i.e., $\mathbf{x} \in \Re^n$ |
| $\bar{x}$ | média amostral de valores escalares |
| $\bar{\mathbf{x}}$ | vetor de médias amostrais |
| $\sigma$ | desvio padrão de distribuição de valores escalares |
| $P(A)$ | probabilidade de ocorrência do evento $A$ |
| $\mathbf{C_x}$ | matriz de covariância de distribuição de vetores |
| $\rho$ | limiar de teste estatístico |
| $Q_i$ | $i$-ésimo quartil |
| $D$ | distância de Mahalanobis |
| $t$ | índice indicativo da iteração durante uma época do treinamento de uma rede neural artificial |
| $i$ | índice de um neurônio numa rede neural competitiva |
| $r_i$ | posição do neurônio $i$ na rede auto-organizável de Kohonen |
| $i^*$ | neurônio vencedor em rede neural competitiva |
| $\mathbf{w}_i$ | vetor de pesos associados ao neurônio $i$ em uma rede neural |
| $\eta$ | taxa de aprendizagem das redes neurais artificiais |
| $\mathbf{X}$ | matriz de dados |
| $\mathbf{X}^T$ | matriz transposta de $\mathbf{X}$ |
| $\mathbf{X}^{-1}$ | matriz inversa de $\mathbf{X}$ |
| $\mathbf{X}^*$ | matriz pseudo-inversa de $\mathbf{X}$ |
| $\lvert\cdot\rvert$ | módulo (valor absoluto) |
| $\lvert\lvert\cdot\rvert\rvert$ | norma euclidiana |
| $f_i$ | ponderador associado ao neurônio $i$ na rede FSCL |
| $h(\cdot)$ | função vizinhança (redes SOM e Neural-Gas) |

| | |
|---|---|
| $\vartheta$ | largura da vizinhança na rede SOM |
| $\Phi$ | mapeamento não-linear |
| $\chi$ | espaço contínuo dos dados de entrada |
| $\mathcal{A}$ | espaço dos neurônios no arranjo geométrico da rede SOM |
| $k$ | posição do neurônio na lista ordenada construída para a rede Neural-Gas |
| $\lambda$ | representa a largura da vizinhança na rede Neural-Gas |
| $\mathbf{e}_q$ | vetor erro de quantização em redes competitivas |
| $e_q$ | erro de quantização em redes competitivas |
| $\mathbf{d}$ | vetor de saídas desejadas para redes supervisionadas |
| $\hat{\mathbf{F}}[\cdot]$ | mapeamento entre entrada e saída estimado pelas redes supervisionadas |
| $\tilde{\mathbf{x}}$ | vetor novidade |
| $\mathcal{L}$ | subespaço linear |
| $\mathcal{L}^{\perp}$ | subespaço linear ortogonal a $\mathcal{L}$ |
| $\mathbf{I}_n$ | matriz identidade $n \times n$ |
| $w_{ij}$ | peso associado à ligação entre entrada $j$ e neurônio $i$ da camada intermediária de uma rede supervisionada |
| $m_{ki}$ | peso associado à ligação entre neurônio $i$ da camada intermediária e saída $k$ de uma rede supervisionada |
| $\xi$ | fator de momento associado ao treinamento da rede MLP |
| $\epsilon$(época) | erro quadrático médio por época de treinamento da rede supervisionadas |
| $\gamma$ | abertura da função de ativação gaussiana para redes GMLP e RBF |
| $e_r$ | erro de reconstrução do Autoassociador MLP |
| $\varphi(\cdot)$ | função de ativação de um neurônio de rede supervisionada |
| $\theta_i$ | limiar de ativação de um neurônio $i$ de rede supervisionada |
| $u_i$ | ativação do neurônio $i$ da camada escondida de uma rede supervisionada |
| $y_i$ | saída do neurônio $i$ na camada de saída de uma rede supervisionada |

# *Sumário*

# 1 Introdução

*Há na comédia* **Verso e Reverso**, *de José de Alencar, um personagem que não vê ninguém entrar em cena, que não lhe pergunte:*

*– Que há de novo?*

*Esse personagem cresceu com os trinta e tantos anos que lá vão, engrossou, bracejou por todos os cantos da cidade, onde ora ressoa a cada instante:*

*– Que há de novo?*

*Ninguém sai de casa que não ouça a infalível pergunta, primeiro ao vizinho, depois aos companheiros de bond. Se ainda não a ouvimos ao próprio condutor do bond, não é por falta de familiaridade, mas porque os cuidados políticos ainda o não distraíram da cobrança das passagens e da troca de idéias com o cocheiro, porém, chega a seu tempo e compensa o pedido.*

**Extraído da crônica** *Pergunta e Resposta*, **de Machado de Assis, escrita em 1893.**

Não vem de hoje, como a passagem acima revela, esse interesse pelo novo, pelo que não é usual. A necessidade de obter novas informações acerca do mundo ao seu redor é uma das características mais marcantes do ser humano, que desde a primeira infância é atraído pelo novo, numa maneira instintiva de aprender, de se defender, em suma, de sobreviver. Aquilo que se destaca da imensidão do fluxo contínuo de informação a que são submetidos diariamente os sentidos de uma pessoa ou animal é, via de regra, mais importante para sua sobrevivência e, por isso, deve ser apreendido (processado e armazenado) para uso posterior.

A busca pelo novo, como fonte essencial de informação, não é importante apenas do ponto de vista cognitivo, mas também do tecnológico. Com a modernização e a crescente automatização de diversas funções até então exercidas por especialistas humanos, surgiu a necessidade cada vez maior de se projetar sistemas automáticos capazes de descobrir novas informações "escondidas" em grandes massas de dados.

**Detecção de Novidades** é uma tarefa de reconhecimento de padrões cujo objetivo

consiste, grosso modo, em reportar a ocorrência de novos eventos, ou novas observações, a partir de um modelo estatístico pré-estabelecido para os dados observados (MARSLAND, 2003). Esta tem sido uma área de crescente interesse nos campos de aprendizado de máquinas e mineração de dados (*data mining*). Isto se deve à demanda existente em certas aplicações por mecanismos automáticos para detecção de padrões que não se adéquam bem ao modelo construído para representar os dados. A maioria destas aplicações, tais como as listadas a seguir, está relacionada a situações em que se deseja monitorar continuamente o estado (condição) de funcionamento de determinado sistema.

- Monitoramento de máquinas elétricas (LI et al., 2002; TANAKA et al., 1995; HARRIS, 1993);
- Processamento de imagens (SINGH; MARKOU, 2004);
- Detecção de alvos de radares (CARPENTER et al., 1997);
- Detecção de tumores em mamografias (ROSE; TAYLOR, 2004);
- Aprendizado de trajetória de robôs móveis (MARSLAND et al., 2002, 2000);
- Reconhecimento de dígitos manuscritos (Le Cun et al., 1990);
- Segurança em redes de computadores (DAO; VEMURI, 2002; ZHANG et al., 2001; HÖGLUND et al., 2000);
- Controle estatístico de processos (GUH et al., 1999);
- Monitoramento de sistemas de telecomunicações (LAIHO et al., 2005; BARRETO et al., 2004; RAIVIO et al., 2001; LAIHO et al., 2001), entre outros.

Em virtude da vasta gama de trabalhos e aplicações em diferentes áreas da engenharia e demais ciências, detecção de novidades também tem sido denominada **detecção de anomalias**, **detecção de intrusos**, **detecção de pontos discrepantes** (*outlier*[1] *detection*), **monitoramento de condição** (*condition monitoring*) e **detecção de uso inadequado** (*misuse detection*).

Várias técnicas e métodos, tais como análise de discriminantes lineares, identificação recursiva de sistemas, redes neurais artificiais, lógica nebulosa, sistemas especialistas, computação evolucionária, dentre outros, têm sido utilizados para lidar com o problema da detecção de novidades. O modelo estatístico dos dados observados, quando dotado de

[1] Tem-se utilizado também os termos **pontos discordantes** ou **valores extremos** como tradução para a palavra inglesa *outlier* (TRIOLA, 1999). Nesta dissertação é mantido o termo em inglês, já bastante difundido na literatura especializada em língua portuguesa.

mecanismos capazes de detectar novos eventos ou observações, é genericamente chamado de **Filtro Detector de Novidades**, ou simplesmente, detector de novidades.

Apesar da diversidade de técnicas apontadas acima, muitas lançam mão de métodos e conceitos pertencentes ao domínio da disciplina de reconhecimento de padrões, reduzindo o problema de detecção de novidades a um dos seguintes problemas

- **Classificação Bivalente (ou Binária):** o conjunto de dados disponível para modelar o comportamento do sistema em estudo é composto exclusivamente por dados pertencentes a uma única classe, usualmente representando o comportamento normal ou esperado do sistema. Esse tipo de dado de treinamento é também conhecido como exemplo positivo. O objetivo, então, é classificar um novo vetor de entrada como pertencente ou não àquela única classe conhecida. É importante destacar que, dentro da área de reconhecimento de padrões, é comum se referir a este tipo de classificação como sendo a detecção de novidades propriamente dita.
- **Classificação Multivalente (ou Multicategórica):** o conjunto de dados de treinamento contém vetores de dados oriundos de diferentes classes. Eles podem, então, ser representativos tanto do comportamento normal (positivo) como de vários comportamentos anormais (negativo), de modo a tornar possível a construção de um classificador para o sistema em questão. O objetivo é, então, classificar um novo vetor de entrada em uma das classes conhecidas, ou em nenhuma delas (rejeição).

O projeto de detectores de novidade pode, então, ser definido como a tarefa na qual uma descrição do que é **previamente conhecido** sobre o sistema é modelado a partir de um conjunto de dados normais e/ou anormais, de modo que observações subseqüentes sejam avaliadas, segundo uma métrica de comparação específica, como algo usual ou não.

Do exposto, os principais desafios encontrados durante o projeto de um detector de novidades são listados abaixo:

**(a)** coleta e rotulação de dados confiáveis;

**(b)** estratégias de pré-processamento (e.g. mudança de escala, remoção de *outliers*, seleção/extração de características, etc.);

**(c)** definição de um classificador adequado e suas características ( se paramétrico ou não-paramétrico, se supervisionado ou não-supervisionado, etc.);

**(d)** escolha da estratégia de classificação a ser utilizada, seja binária ou multicategórica, escolha que está fortemente relacionada aos dados coletados;

**(e)** escolha da métrica de avaliação de novidade (fortemente relacionada com a escolha do classificador), e

**(f)** cálculo de limiares de decisão, a partir dos quais a decisão sobre a novidade de certo evento é tomada.

Como pode ser verificado em artigos de revisão do estado da arte em detecção de novidades recentemente publicados (HODGE; AUSTIN, 2004; MARKOU; SINGH, 2003a, 2003b; MARSLAND, 2003), grandes esforços têm sido devotados aos itens (c)-(f), ou seja, projeto de classificadores poderosos e técnicas de determinação de limiares de decisão, enquanto uma atenção bem menor vem sendo dada a aspectos relacionados aos itens (a)-(b), ou seja, com a qualidade da coleta, rotulação e pré-processamento dos dados, bem como suas influências no desempenho desses classificadores.

No que diz respeito à qualidade do conjunto de dados, a maioria dos trabalhos consultados assume implicitamente que seus dados são livres de pontos discrepantes ou que, se tais pontos existem, eles são previamente conhecidos. Coloquialmente, pode-se dizer que *outliers* são **pontos fora da curva**, pois não partilham da mesma distribuição de probabilidades que a maioria das demais observações. Desta forma, pode-se dizer que *outliers* são observações diferentes das usuais e que, por isso, podem ser caracterizados como novidades.

A ocorrência de *outliers* tem causas variadas, tais como

- fontes de ruído e erros de medidas durante a coleta dos dados;
- falhas sistêmicas;
- comportamento inesperado do sistema (comportamento fraudulento);
- mudança do ponto de operação do sistema;
- flutuações estatísticas inerentes ao conjunto de dados.

Ao entender *outliers* como novidades, as mesmas técnicas utilizadas para detecção de novas observações podem ser utilizadas para encontrar *outliers* em uma determinada massa de dados, eliminando-os se for necessário. Neste caso, técnicas de detecção de novidades funcionariam como **Filtros Eliminadores de Outliers**. Isto posto, nesta dissertação utilizam-se indistintamente os termos detecção de *outliers* e detecção de novidades.

É importante mencionar também que a etapa de rotulação dos dados, ainda que realizada por um especialista, está susceptível a incorreções. Mesmo sob hipótese de que

os dados são livres de *outliers*, é muito difícil, se não impossível, saber de antemão se os dados coletados, no que diz respeito ao número de exemplos positivos e/ou negativos, são suficientes para prover uma descrição verossímil da natureza estatística subjacente ao sistema. Por exemplo, para algumas aplicações, o número de exemplos negativos pode ser demasiado pequeno, já que muitas vezes tais exemplos são raros ou difíceis de se coletar. Essa questão é de grande importância, uma vez que se sabe que para um bom desempenho na classificação, o número de exemplos por classe deve ser idealmente balanceado (WEBB, 2002).

A questão levantada no parágrafo anterior é particularmente verdadeira para classificadores poderosos, tais como aqueles baseados nas redes neurais supervisionadas multicamadas MLP e RBF, nas quais é usual que o modelo sofra um ajuste excessivo (*overfitting*) aos dados de treinamento (LAWRENCE et al., 1998). Neste caso, recomenda-se excluir os poucos exemplos negativos disponíveis do processo de construção do modelo dos dados, tratando a tarefa da detecção de novidades como um problema da classificação binária, na qual se treina o classificador somente com os exemplos positivos (normais). As observações excluídas são usadas posteriormente para testar o desempenho do sistema de detecção de novidades.

Alguns autores (SINGH; MARKOU, 2004; AUGUSTEIJN; FOLKERT, 2002; VASCONCELOS et al., 1995), entretanto, sugerem a inclusão de exemplos negativos como **outliers conhecidos** (*known outliers*) ou **outliers falsos** (*fake outliers*) durante o processo de construção do modelo dos dados. Este tipo de *outlier* é construído artificialmente seja pela alteração dos valores numéricos das variáveis em questão (por exemplo, inserindo ruído gaussiano aleatório com uma dada variância) ou simplesmente mudando o rótulo de alguns exemplos de negativo para positivo, passando estes a integrar o conjunto de observações positivas que é utilizado para construir o modelo estatístico dos dados. Esses autores argumentam que tal procedimento pode ser benéfico para o sistema da detecção de novidades, melhorando seu desempenho como um todo.

Do exposto nos parágrafos anteriores pode-se perceber que o projeto de um filtro detector de novidades, além de exigir as etapas básicas de projeto de qualquer classificador, também exige etapas adicionais de escolha da métrica de avaliação da novidade e de cálculo do(s) limiar(es) de decisão. Maiores detalhes sobre o problema de detecção de novidades são apresentados em capítulos posteriores desta dissertação. A seguir são discutidas as motivações para o presente trabalho.

## 1.1 Motivação

As características e dificuldades no projeto de detectores de novidades expostas na Seção anterior e a revisão bibliográfica levada a cabo nesta dissertação, indicou alguns pontos que necessitam de maior atenção, a saber

- **grande fragmentação da área:** diversos métodos são propostos e não há uma maneira satisfatória de compará-los em diferentes aplicações.
- **indefinição na escolha dos limiares de decisão:** podem ser usados testes com um ou mais limiares para a decisão e a maneira como estes limiares são calculados nem sempre é bem fundamentada teoricamente.
- **indefinição sobre a estratégia de classificação:** tanto classificação binária quanto a classificação multicategórica têm sido utilizadas em detecção de novidades sem indicação de qual abordagem é mais apropriada para projetos de detectores de novidades.
- **indefinição quanto ao tratamento dos** *outliers***:** este tópico está relacionado ao item logo acima, e diz respeito à questão de usar ou não *outliers* falsos durante o treinamento dos classificadores.

Os tópicos supracitados serviram de motivação para o desenvolvimento do presente trabalho que, por sua vez, busca propor estratégias unificadoras que minimizem as dificuldades ora impostas ao projeto de detectores de novidades usando redes neurais artificiais. Os objetivos específicos deste trabalho estão detalhados a seguir.

## 1.2 Objetivos da Dissertação

Tendo em vista os tópicos mencionados nas seções anteriores, concernentes ao projeto de detectores de novidades e às motivações deste trabalho, os principais objetivos desta dissertação são os seguintes:

1. propor um método de cálculo de limiares de decisão baseado em intervalos de confiança *bootstrap* não-paramétrico e em redes neurais não-supervisionadas competitivas;
2. generalizar o método proposto para permitir seu uso também por redes neurais supervisionadas;

3. propor uma metodologia geral para comparação de desempenho de diferentes detectores de novidades baseados em redes neurais artificiais; e

4. empregar a metodologia proposta como arcabouço comum para comparação de desempenho de detectores de novidades baseados em redes neurais supervisionadas e não-supervisionadas.

## 1.3 Organização Geral desta Dissertação

O restante do documento está organizado em cinco capítulos e dois apêndices. Um breve comentário sobre cada um deles é feito a seguir.

- **Capítulo 2: Detecção de Novidades – Descrição do Problema**.

  Este capítulo define melhor o problema de detecção de novidades e apresenta brevemente as abordagens estatísticas clássicas utilizadas no projeto de detectores de novidades.

- **Capítulo 3: Redes Neurais Artificiais para a Detecção de Novidades**.

  Este capítulo tem por objetivo descrever sucintamente as arquiteturas de redes neurais avaliadas neste trabalho, de modo a facilitar o entendimento dos modelos descritos nos capítulos posteriores.

- **Capítulo 4: Detecção de Novidades Usando Redes Neurais Competitivas**.

  Este capítulo propõe um novo método para detecção de novidades usando redes neurais competitivas e limiares de decisão obtidos de intervalos de confiança *bootstrap* não-paramétrico. Compara-se aqui o desempenho do método proposto com outros já disponíveis na literatura numa aplicação de detecção de anomalias em sistemas de comunicação celular de terceira geração (3G).

- **Capítulo 5: Detecção Neural de Novidades: Uma Abordagem Unificadora**.

  Este capítulo propõe uma metodologia geral para o cálculo de limiares de decisão para a detecção de novidades. Esta abordagem permite comparar diferentes sistemas de detecção de novidades que variam quanto ao tipo de rede neural utilizada (supervisionadas e não-supervisionadas). Um dos pontos fortes dessa metodologia é mostrar que métodos de cálculo de limiares de decisão, previamente utilizados em detectores de novidades baseados em redes neurais competitivas, podem ser utilizados em detectores de novidades baseados em redes supervisionadas. Os diversos sistemas de detecção de novidades baseados em redes neurais descritos no capítulo

anterior são, então, comparados usando a metodologia proposta na detecção de tumores em mamografias.

- **Capítulo 6: Conclusão e Trabalhos Futuros**.

  Este capítulo conclui a dissertação e propõe futuros desdobramentos e evoluções possíveis a partir do que é desenvolvido no presente trabalho.

- **Apêndice A: Pré-processamento de Dados**.

  Este apêndice discute algumas questões relevantes na preparação dos dados quando se usa RNAs. Algumas estratégias-padrão de preparo dos dados e métodos de pré-processamento utilizados neste trabalho são discutidos.

- **Apêndice B: Simulação de Sistemas CDMA2000**.

  Este apêndice descreve sucintamente as principais características do simulador de sistema celular CDMA2000 empregado para a geração dos dados utilizados no Capítulo 4.

## 1.4 Produção Científica

Durante o desenvolvimento deste trabalho de mestrado, o candidato teve como produção científica os seguintes artigos:

1. Barreto, G. A., Mota, J. C. M., Souza, L. G. M., **Frota, Rewbenio A.** & Aguayo, L. (2005). "Condition Monitoring of 3G Cellular Networks Through Competitive Learning", *IEEE Transactions on Neural Networks*, v. 16, n. 5, p. 1064 - 1075.

2. **Frota, Rewbenio A.**, Barreto, G. A. & Mota, J. C. M. (2005). "Proposta de Uma Metodologia Não-Paramétrica para Avaliação de Redes Neurais em Tarefas de Detecção de Novidades", *Anais do VII Simpósio Brasileiro de Automação Inteligente (SBAI)*, São Luis – MA.

3. Barreto, G. A., Mota, J. C. M., Souza, L. G. M., **Frota, Rewbenio A.**, Aguayo, L., Yamamoto, S. & Macedo, P. (2004). "Competitive Neural Networks for Fault Detection and Diagnosis in 3G Cellular Systems", *LNCS – Lecture Notes in Computer Science*, v. 3124, p. 207 - 213.

4. Barreto, G. A., Mota, J. C. M., Souza, L. G. M. & **Frota, Rewbenio A.** (2004). "Nonstationary Time Series Prediction Using Local Models Based on Competitive Neural Networks", *LNAI – Lecture Notes in Artificial Intelligence*, v. 3029, p. 1146 - 1155.

5. Barreto, G. A., Mota, J. C. M., Souza, L. G. M., **Frota, Rewbenio A.**, Aguayo, L., Yamamoto, S. & Macedo, P. (2004). "A New Approach to Fault Detection and Diagnosis in Cellular Systems Using Competitive Learning", *Anais do VIII Simpósio Brasileiro de Redes Neurais (SBRN)*, São Luis – MA, IEEE Press, v. I.

6. Barreto, Guilherme A., **Frota, Rewbenio A.** & Medeiros, Fátima N. S. (2004). "On the Classification of Mental Tasks: A Performance Comparison of Neural and Statistical Approaches", *Machine Learning for Signal Processing*, ed. Piscataway, NJ: IEEE Press, v. XIV, p. 529-538.

## 1.5 Conclusão

Neste Capítulo, o problema de detecção de novidades é apresentado, sendo discutidas as principais dificuldades envolvidas no projeto de detectores de novidades. São apresentadas também as motivações para se propor estratégias que permitam comparar, sob bases teóricas comuns, diversos detectores neurais de novidades. Por fim, são apresentados os objetivos específicos desta dissertação, bem como a produção científica resultante.

Após esta breve introdução ao assunto de interesse, a etapa seguinte consiste na descrição técnica do problema de detecção de novidades e as soluções mais usuais, propostas com o passar dos anos.

# 2 Detecção de Novidades – Descrição do Problema

Neste capítulo o problema de detecção de novidades é definido de forma mais completa que a introdução feita no capítulo anterior. Diversos conceitos e definições são apresentados de modo a facilitar o entendimento de capítulos posteriores. É também enfatizado o fato de a detecção de novidades estar fortemente relacionada ao problema estatístico de detecção de *outliers*.

Ainda neste capítulo são descritos alguns métodos estatísticos clássicos de detecção de *outliers*. O objetivo é mostrar como estas técnicas são aplicadas à detecção de novidades propriamente dita e, a partir delas, estabelecer uma motivação para os novos métodos propostos neste trabalho.

## 2.1 Introdução

Diversos autores propuseram ao longo dos anos várias definições sobre o significado o termo *outlier*, sem que se tenha ainda uma definição universalmente aceita (HODGE; AUSTIN, 2004). Grosso modo, tenta-se resumir aqui as idéias comuns a todas elas na seguinte definição:

> *Um* **outlier** *é uma observação (escalar ou vetorial) que difere marcadamente dos outros membros da amostra a que ele pertence, tal como indicado pelo modelo escolhido para representar esta amostra.*

Uma definição alternativa, porém geral o suficiente para abranger também os termos "novidade" e "anomalia", é dada a seguir:

> *Um* **outlier** *ou uma* **novidade** *é uma observação que parece ser inconsistente com aquelas usadas para construir o modelo dos dados.*

Usualmente, para uma determinada massa de dados, é possível conceber algum modelo estatístico a partir do qual novas observações são avaliadas como oriundas ou não da mesma distribuição de dados que gera o modelo. Aquelas observações que não podem ser satisfatoriamente explicadas pelo modelo são considerados *outliers* ou novidades. Conforme mencionado no Capítulo 1, tais observações discrepantes podem ter origens diversas, tais como erros na medição, coleta ou rotulação de dados, mudança na dinâmica do sistema, um evento raro ou uma falha do sistema.

Embora sejam áreas com uma grande similaridade, há na literatura especializada trabalhos referindo-se ora à detecção de novidades, ora à detecção de *outliers*. Tal distinção é justificada por uma sutil diferença entre elas, exposta a seguir.

**Detecção de Outliers** consiste em procurar, numa dada massa de dados, aquelas observações que não são suficientemente similares às demais, sendo um problema inerente à massa de dados que se tem em mãos. A detecção de *outliers* vem sendo utilizada tradicionalmente como um meio eficaz de limpeza de massas de dados, retirando padrões que podem dificultar a modelagem e levar a erros de classificação ou de predição.

**Detecção de Novidades** trata de dados que chegam continuamente, e o que se busca são pontos que não sejam suficientemente similares à maioria dos pontos precedentes usados para construir o modelo dos dados.

De modo mais formal, para um instante inicial qualquer, $t_0$, considera-se a massa $M_1$ de dados gerados antes de $t_0$ como o conjunto de dados já observados, enquanto a massa $M_2$ de dados coletados após $t_0$ como os dados a serem categorizados. Encontrar pontos de $M_2$ que são *outliers* usando o modelo construído a partir de $M_1$ é o mesmo que encontrar observações posteriores que não são semelhantes à maioria de observações anteriores, ou seja, que não podem ser colocadas na mesma categoria que as observações anteriores. Logo, após o modelo construído, detecção de novidades será, na prática, igual à detecção de *outliers*.

Assim, para todos os propósitos desta dissertação, o termo "novidade" é usado de uma forma bastante geral, abrangendo os termos *outlier* e anomalia.

## 2.2 Princípios da Detecção de Novidades

Há diversas questões importantes relacionadas à detecção de novidades que podem ser resumidas em termos de alguns princípios, úteis na avaliação qualitativa do projeto e do desempenho de detectores de novidades (MARKOU; SINGH, 2003a).

O **princípio do compromisso** diz que, antes de tudo, um método de detecção de novidades deve maximizar a detecção de observações novas (detecção verdadeira) en-

quanto minimiza o número de falsas detecções. Este compromisso deve ser, na medida do possível, controlável experimentalmente.

O **princípio da escala uniforme dos dados** afirma que deve ser possível que todos os dados de teste e de treinamento se encontrem dentro da mesma escala de valores após a normalização ou outro método de pré-processamento (SINGH; MARKOU, 2004).

O **princípio do número mínimo de parâmetros** diz que um método de detecção de novidades deve ser projetado de modo a minimizar o número de parâmetros que são ajustados pelo usuário. Este princípio também é chamado **Princípio da Parcimônia** (*Principle of Parsimony*), ou ainda, **Princípio da Navalha de Occam** (*Occam's Razor*).

O **princípio da generalização** sustenta que o sistema deve ser capaz de generalizar sem confundir a informação generalizada com novidade (TAX; DUIN, 1998).

O **princípio da independência** visa garantir que o método de detecção de novidades seja independente do número de características (dimensão dos vetores de dados) e do número de classes disponíveis. Deve também apresentar desempenho aceitável em conjuntos de dados não-balanceados (no que diz respeito ao número de exemplos por classe), reduzido número de observações e presença de ruído.

O **princípio do adaptabilidade** diz que um sistema que reconheça observações novas durante o teste deve ser capaz de usar esta informação para retreinamento.

Finalmente, o **princípio da complexidade computacional** visa garantir que a complexidade computacional de um mecanismo detector de novidades seja a menor possível, uma vez que um grande número de aplicações neste campo são realizadas em tempo real.

Os princípios discutidos acima, apesar de servirem como guias no momento do projeto e avaliação de detectores novidades, em geral não são adequados para uma comparação quantitativa do desempenho dos mesmos, uma vez que dois sistemas detectores de novidades podem observar todos os princípios acima na sua concepção e, ainda assim, apresentar desempenhos bastante distintos. Conforme mencionado no capítulo anterior, um dos objetivos desta dissertação é justamente propor uma metodologia geral de comparação de desempenho, bem-fundamentada estatísticamente.

## 2.3 Detecção de Novidades Formulada como Teste de Hipóteses

A maioria das técnicas de detecção de novidades podem ser caracterizadas sob o formalismo dos testes estatísticos de hipóteses, uma vez que usualmente essas técnicas

comparam alguma medida de novidade com um ou mais limiares de decisão.

Testes de hipóteses são usados quando existe interesse em decidir sobre a verdade ou não de uma hipótese específica, por exemplo, se duas amostras têm a mesma média, ou se determinado parâmetro populacional assume um valor em particular. Assim, uma hipótese estatística é uma alegação, ou afirmação, sobre uma propriedade de uma amostra, observação ou parâmetro populacional (TRIOLA, 1999).

Formula-se uma hipótese com o intuito de aceitá-la ou rejeitá-la, segundo as evidências estatísticas disponíveis. Esta decisão, rejeitar ou não uma hipótese, é o objetivo do teste de hipóteses. Em tais testes, a hipótese em estudo é chamada **Hipótese Nula** e é denotada por $H_0$. Há ainda uma outra hipótese relacionada a $H_0$, verdadeira quando esta é falsa, que é denominada **Hipótese Alternativa**, cuja representação é $H_1$.

Os testes de hipóteses seguem uma diretriz geral da estatística segundo a qual deve-se analisar uma amostra para distinguir os resultados que podem facilmente ocorrer (eventos prováveis ou conhecidos) daqueles que dificilmente ocorrem (eventos improváveis ou desconhecidos). O uso desses testes permite, então, explicar a ocorrência de resultados muito improváveis, revelando que ocorreu efetivamente um evento raro (detecção de novidades), ou que o cenário não é como se supunha (falha na modelagem).

Para testes de hipóteses, além de $H_0$ e $H_1$, convém definir os seguintes elementos:

- **Estatística de teste:** é uma estatística (medida) amostral, ou seja, um valor baseado nos dados amostrais. Utiliza-se a estatística de teste para tomar uma decisão sobre a aceitação/rejeição da hipótese nula.
- **Nível de significância ($\alpha$):** é definido como a probabilidade de se rejeitar a hipótese nula quando ela é verdadeira. O valor de $\alpha$ é tipicamente predeterminado. O nível de confiança do teste é definido como $1 - \alpha$.
- **Região crítica:** é o conjunto de todos os valores da estatística de teste que levam à rejeição da hipótese nula.
- **Valor crítico:** é o valor, ou valores, que delimita(m) a região crítica. Os valores críticos dependem da natureza da hipótese nula, da distribuição amostral e do nível de significância do teste ($\alpha$). Em detecção de novidades, os valores críticos são chamados **limiares de decisão**.

Quando se formula uma conclusão a respeito de $H_0$, dois tipos de erro são possíveis de ocorrer:

- **Erro Tipo I**: ocorre se $H_0$ for rejeitada quando, de fato, ela é verdadeira. A probabilidade de cometer um erro Tipo I é dada pelo nível de significância, $\alpha$,

cujo valor é definido pelo investigador tendo em conta as conseqüências de tal erro. Tenta-se fazer o nível de significância o mais baixo possível, de forma a proteger a hipótese nula e prevenir, tanto quanto possível, que o investigador cometa falsas intervenções. Dependendo da área de aplicação, o erro Tipo I é também conhecido como **Alarme Falso**, **Detecção Falsa** ou ainda **Falso Positivo**.

- **Erro Tipo II**: ocorre se $H_0$ não for rejeitada quando, na verdade, deveria ser. A probabilidade de se cometer este erro é denotada por $\beta$ (de valor geralmente desconhecido). O erro Tipo II é também conhecido como **Ausência de Alarme** ou **Falso Negativo**.

Define-se o **poder** de um teste de hipótese como a probabilidade de rejeitar a hipótese nula quando ela é falsa. Numericamente define-se o poder de um teste como $1 - \beta$. Em geral, quanto maior o tamanho da amostra ($N$), maior o poder do teste. Sempre que possível, é desejável decidir sobre o tamanho adequado da amostra antes de conduzir o estudo, de forma que os resultados do teste de hipóteses tenham poder suficiente para responder à questão científica de interesse.

O nível de significância $\alpha$ deve ser fixado de acordo com a probabilidade de acerto que se deseja ter nos testes. Sendo conveniente, $\alpha$ pode ser diminuído até tão próximo de zero quanto se queira, mas isso resultará em intervalos (região crítica) de amplitude cada vez maiores e testes com poder cada vez menor, devido ao conseqüente aumento no valor de $\beta$, o que significa perda de precisão no teste.

Com relação ao controle de ambos os tipos de erro, o ideal seria obter intervalos com elevado nível de confiança (ou seja, $1 - \alpha$ alto), pequena probabilidade de erro e grande precisão (poder do teste). Isso porém requer uma amostra suficientemente grande, pois, para $N$ fixo, confiança e precisão variam em sentidos opostos.

## 2.4 Tipos de Testes de Novidades

Testes de hipóteses para detecção de novidades podem ser divididos em dois grupos principais quanto ao número de limiares (valores críticos) usados, e, conseqüentemente, à natureza do teste (região crítica): teste de limiar simples (teste unilateral) ou de limiar duplo (teste bilateral).

Em geral, na literatura sobre detecção de novidades, limiares são calculados a partir da distribuição amostral da estatística de teste através dos **percentis**. Um percentil é uma medida da posição relativa de uma unidade amostral em relação a todas as outras. Tecnicamente falando, o percentil de uma distribuição de valores é um número $P_\varrho$ tal que uma percentagem $100(1 - \varrho)\%$ de valores da população são menores ou iguais a $P_\varrho$. Por

exemplo, o 75 percentil (também conhecido como o quantil 0,75) é um valor tal que 75% dos valores da variável estão abaixo dele (TRIOLA, 1999).

Alguns percentis recebem atenção e nomenclatura especiais. Os percentis cujos valores de $100(1-\varrho)$ são os números 25, 50 e 75 são chamados, respectivamente, **primeiro quartil** (simbolizado por $Q_1$), **segundo quartil** ($Q_2$), que corresponde à mediana da distribuição, e **terceiro quartil** ($Q_3$).

É importante salientar a diferença entre percentis e percentagens. Um percentil é relacionado somente com a posição relativa de uma observação, quando comparada com os outros valores. Desse modo, se um estudante que acerta 75% de um teste, mas cuja nota é o 40 percentil, significa que somente 40% da turma teve nota pior que aquele estudante e 60% saiu-se melhor.

### 2.4.1 Testes de Detecção de Novidades com Limiar Simples

Nesta Seção, são descritos os testes de hipóteses que utilizam um único limiar para avaliar o grau de novidade de uma observação. São os testes mais usuais em detecção de novidades, seja quando se empregam técnicas estatísticas clássicas (MARKOU; SINGH, 2003a) ou quando se empregam técnicas baseadas em redes neurais artificiais (MARKOU; SINGH, 2003b).

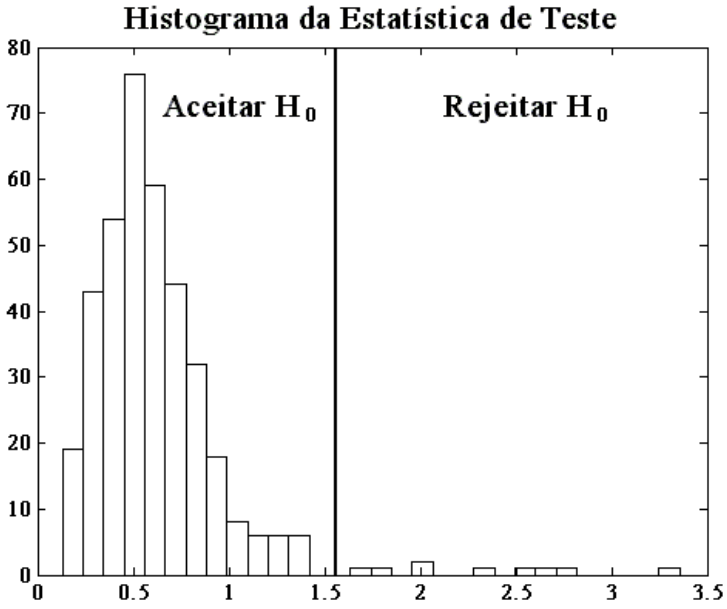


**Figura 2.1:** Teste unilateral tipo valor-$p$: a linha vertical marca o limiar de decisão.

Geralmente existe uma medida de incoerência (ou medida de novidade) entre uma dada observação e o modelo construído para representá-la. A idéia é que quanto maior

for essa medida de incoerência, menor a probabilidade de que aquela observação seja um membro legítimo da amostra utilizada para a construção do modelo. Desta forma, pequenos valores desta medida de incoerência praticamente garantem que a observação é um legítimo representante do conjunto de dados.

O teste de limiar simples mais conhecido é o teste do **valor-*p*** (TRIOLA, 1999; SPIEGEL, 1984). Por definição, o valor-$p$ é a probabilidade de se observar o valor da estatística de teste tão ou mais extremo do que o valor observado, assumindo que a hipótese nula é verdadeira. Em outras palavras, o valor-$p$ é a probabilidade de cometer o erro de Tipo I com os dados de uma amostra específica. Compara-se o valor-$p$ com o nível de significância escolhido e toma-se a decisão. Se o valor-$p$ for menor que o nível de significância escolhido rejeita-se $H_0$, caso contrário, aceita-se $H_0$. A Figura 2.1 ilustra um teste de limiar simples.

Os testes unilaterais mais utilizados são capazes de detectar novidades apenas numa região onde a medida de novidade adotada tem valores elevados. É, entretanto, possível sob certas circunstâncias, tais como presença de *outliers* no conjunto de dados utilizados para a construção do modelo ou ajuste excessivo do modelo aos dados, que um pequeno valor para a medida de novidade não seja mais um indicativo seguro de que uma nova entrada não seja uma novidade. Para tratar destes casos, é necessário o uso de testes bilaterais, capazes de detectar novidades tanto na região que contém valores elevados da medida de novidade, como aquela que contém valores incomumente pequenos.

### 2.4.2 Testes de Detecção de Novidades com Limiar Duplo

Seja um conjunto de $N$ observações de uma variável $x \in \mathbb{R}$, $\bar{x}$ a média e $\sigma$ o desvio padrão da distribuição dos dados. Uma observação é declarada como uma novidade se estiver fora do intervalo

$$(\bar{x} - k\sigma, \bar{x} + k\sigma), \tag{2.1}$$

na qual o valor de $k$ é usualmente 1,96 ou 2,58. A justificativa destes valores para $k$ está no fato de se presumir que a distribuição é gaussiana e, portanto, espera-se ter cerca de 95% (99%, respectivamente) dos dados no intervalo centrado na média $\bar{x}$ com abertura igual a $k$ desvios-padrão (TRIOLA, 1999).

Da Equação (2.1), a observação $x$ é considerada uma novidade se

$$\frac{|x - \bar{x}|}{\sigma} > k. \tag{2.2}$$

A principal limitação do teste acima é a suposição de uma distribuição normal para os dados, algo que freqüentemente não ocorre. Além disso, o desvio-padrão é altamente sensível à presença de *outliers* na massa de dados. A Figura 2.2 ilustra a forma geral de

um teste de hipóteses do tipo bilateral.

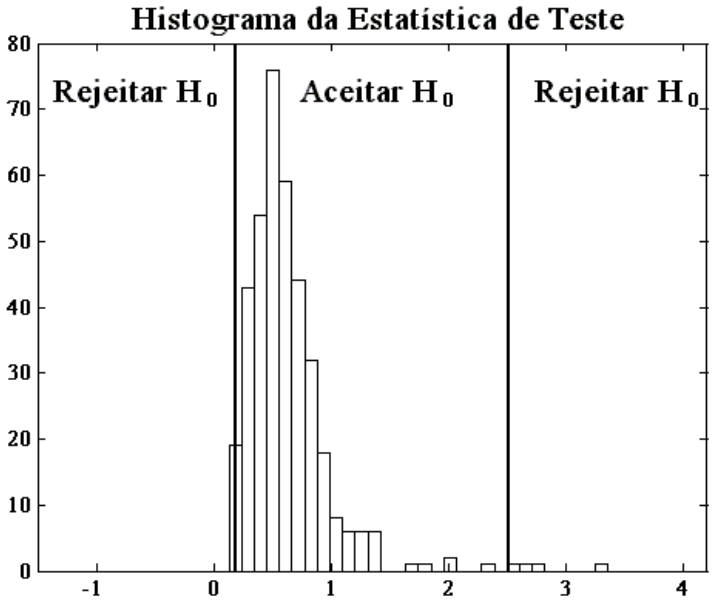


**Figura 2.2:** Teste bilateral: as linhas verticais marcam os limiares de decisão.

Outro método clássico para detectar e visualizar novidades (ou *outliers*), é conhecido como *boxplot*[1] (TUKEY, 1977). Um *boxplot* é um modo gráfico de se expor *outliers* presentes numa dada amostra (conjunto de $N$ observações). Num *boxplot* é possível fazer uma distinção entre dois tipos distintos de novidades: as suaves e as extremas. Uma observação $x$ é declarada uma **novidade extrema** se $x$ estiver fora intervalo $[Q1-3IQR, Q3+3IQR]$, ou uma **novidade suave** se $x$ estiver fora intervalo $[Q1 - 1,5IQR, Q3 + 1,5IQR]$, em que $IQR = Q3 - Q1$ é chamado intervalo interquartis (*interquartile range* – $IQR$). Os valores 1,5 e 3 são escolhidos através da comparação com uma distribuição gaussiana. Os principais *softwares* estatísticos incluem a ferramenta *boxplot* entre seus métodos gráficos de análise de dados. A Figura 2.3 exemplifica o uso do *boxplot*.

Um outro método de detecção de novidades baseados em limiares duplos é definido a seguir. Em virtude de sua grande importância para este trabalho, optou-se por descrevê-lo em uma subseção própria.

### 2.4.3 Intervalos de Confiança

Intervalos de confiança surgem da necessidade de se avaliar a qualidade de estimativas pontuais de parâmetros populacionais. Um intervalo da confiança para um parâmetro

[1] O termo em português é **diagrama de caixas (TRIOLA, 1999)**, mas, por seu uso pouco freqüente em publicações em língua portuguesa, o termo em inglês continuará sendo usado neste texto.

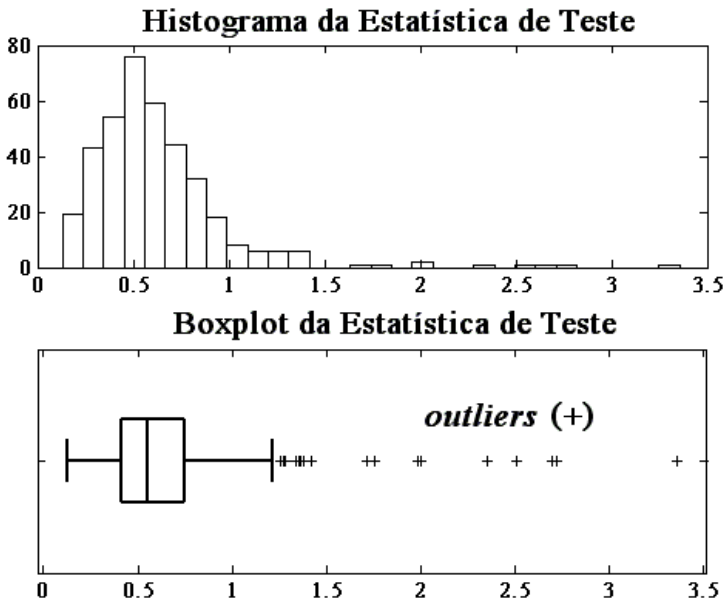


**Figura 2.3:** Visualização de novidades e *outliers* a partir de uma amostra de dados qualquer.

populacional desconhecido fornece uma estimativa do parâmetro e uma medida da confiança nessa estimativa, indicando quão boa é uma estimativa pontual (TRIOLA, 1999).

Seja $X \in \mathbb{R}$ uma variável aleatória cuja distribuição de probabilidades depende de um parâmetro desconhecido $\varepsilon$. Dada uma amostra aleatória de $X$, chamados valores amostrais $x_1, x_2, \ldots, x_N$. As duas estatísticas $L_1$ e $L_2$ definem um intervalo de confiança de $100(1-\alpha)\%$ para $\varepsilon$ se

$$P(L_1 \leq \varepsilon \leq L_2) = 1 - \alpha. \tag{2.3}$$

O intervalo de confiança pode ser centrado, ou seja,

$$P(L_1 > \varepsilon) = P(L_2 < \varepsilon) = \frac{\alpha}{2}. \tag{2.4}$$

Em geral, assume-se uma distribuição gaussiana para as observações da amostra $\mathcal{X}$. Por exemplo, assumindo que a estimativa amostral da média da população $\mu$ é denotada $\bar{x}$, seu intervalo de confiança é calculado pela seguinte expressão:

$$\bar{x} - z_{\alpha/2}\sigma < \mu < \bar{x} + z_{\alpha/2}\sigma, \tag{2.5}$$

em que $\sigma$ é o desvio-padrão da população, e $z_{\alpha/2}$ é um valor crítico, chamado $z-score$, que garante que a área entre $-z_{\alpha/2}$ e $z_{\alpha/2}$, sob a curva da função densidade de probabilidade de uma variável aleatória gaussiana dada por $z = \frac{x-\mu}{\sigma}$, é igual a 1-$\alpha$, conforme mostra a

Figura 2.4.

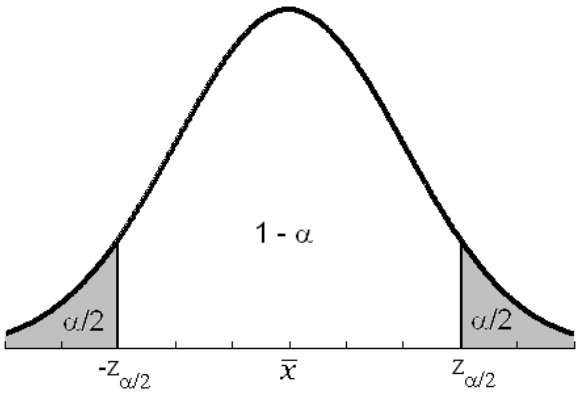


**Figura 2.4:** Intervalo de confiança simétrico para a média $\mu$ de uma distribuição gaussiana.

A razão subjacente ao uso do subescrito $\alpha/2$ é que a área de cada uma das regiões restantes (caudas esquerda e direita) é igual a $\alpha/2$. Alguns valores críticos comumente utilizados são 1,96 e 2,58, correspondendo a graus de confiança de 95% e 99%, respectivamente. Vale ressaltar que, para estes valores críticos, a Equação (2.5) é exatamente igual à Equação (2.1).

Naturalmente, espera-se que o intervalo da confiança contenha o valor verdadeiro do parâmetro, mas, ocasionalmente, o elemento aleatório é suficientemente grande para distorcer a estimativa de $\varepsilon$ de modo que o valor verdadeiro acabe por se encontrar fora do intervalo de confiança. A probabilidade que um intervalo de confiança contenha o valor verdadeiro do parâmetro é $1 - \alpha$. Obviamente, quanto menor é $\alpha$ maior deve ser o intervalo da confiança.

## 2.5 Tipos de Novidades e de *Outliers*

Novidades e, por extensão, *outliers* podem ser observações univariadas (escalares) ou multivariadas (vetores), de acordo com a dimensão do espaço em que são definidos. Portanto, há métodos de detecção de novidades univariadas e multivariadas.

A detecção de novidades univariadas consiste, basicamente, em aplicar sobre as observações algum dos testes (de limiar simples ou duplo) anteriormente descritos.

Já para as novidades multivariadas é necessário maior cuidado. É comum pensar que estas podem ser detectadas através de testes para a detecção de novidades univariados aplicados a cada componente do vetor de dados, mas isso não é verdade. Por exemplo, na Figura 2.5, o ponto em destaque no canto inferior esquerdo é um *outlier* multivariado. Pode-se observar no entanto que, separadamente, cada uma de suas componentes não são

*outliers* univariados, pois em cada uma das dimensões, os valores estão dentro da faixa do "estatisticamente" normal. Por outro lado, uma observação vetorial pode ter em suas componentes *outliers* univariados, mas ela globalmente não é um *outlier* multivariado. Um exemplo é o ponto no canto inferior direito da mesma Figura, que não pode ser considerado um *outlier* multivariado, por estar de acordo com a clara correlação linear entre as componentes $x_1$ e $x_2$ observada na grande maioria dos demais pontos (marcados com '*').

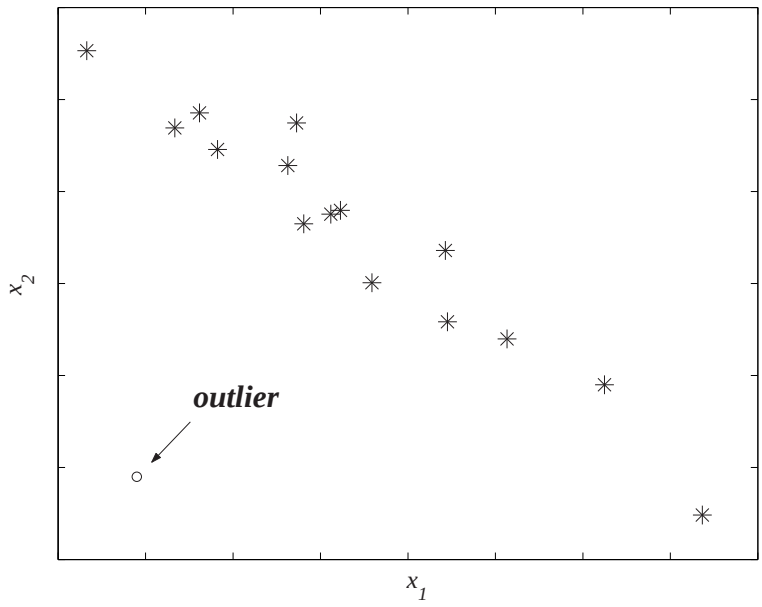


**Figura 2.5:** Exemplo de *outlier* multivariado não detectável por métodos de detecção de *outliers* univariados.

Há diversos métodos para detectar novidades multivariadas. Em geral, calcula-se uma medida de novidade escalar derivada dos dados multivariados e a partir dessa medida as novidades são encontradas através de testes de hipóteses univariados. Os métodos mais usuais são baseados no cálculo da distância de Mahalanobis (WORDEN et al., 2000) e na aplicação de um dos tipos de teste de hipóteses descritos na Seção 2.4.

Seja $\mathbf{x} \in \Re^n$ uma observação de uma amostra de observações multivariadas[2] de tamanho $N$. Seja $\bar{\mathbf{x}}$ o centróide da amostra de dados, que é um vetor $n$-dimensional com as médias de cada componente. Seja $\mathbf{X}$ a matriz de dados original com as colunas centradas em suas médias. Então,

$$\mathbf{C_x} = \frac{1}{(N-1)}\mathbf{X}^T\mathbf{X}, \tag{2.6}$$

[2]Em um contexto classificação supervisionada, deve-se também saber as classes a que cada um dos exemplos pertence. Um método comum de rotulação de dados é incluir as classes como a última coluna da matriz de dados. O objetivo é detectar todos os exemplos que parecem ser incomuns, estes são as novidades multivariadas.

na qual $\mathbf{C_x} \in \Re^{n \times n}$ é a matriz de covariância das $n$ características. A versão multivariada da Equação (2.2) é

$$D_M(\mathbf{x}, \bar{\mathbf{x}}) = \left[(\mathbf{x} - \bar{\mathbf{x}})^T \mathbf{C}_\mathbf{x}^{-1} (\mathbf{x} - \bar{\mathbf{x}})\right]^{\frac{1}{2}} > k, \tag{2.7}$$

na qual $D_M$ é a distância de Mahalanobis do vetor $\mathbf{x}$ ao centróide $\bar{\mathbf{x}}$ da amostra de dados.

Usualmente, um vetor com um valor alto para $D_M$ é considerado uma novidade ou *outlier* (KNORR et al., 2000). O valor do limiar de decisão $k$ nas Equações (2.2) e (2.7) é determinado pelo valor crítico $\chi^2_{g,1-\alpha}$ da distribuição chi-quadrado com $g$ graus de liberdade e nível de significância $\alpha$ (usualmente $\alpha = 0{,}05$).

Rocke & Woodruff (1996) sustentam que métodos baseados na distância de Mahalanobis têm bom desempenho na identificação de novidades e *outliers* dispersos, aleatoriamente distribuídos em torno da massa principal de dados. Entretanto, quando as novidades estão concentradas, ou seja, se apresentam na forma de um agrupamento localizado em uma região específica, a distância de Mahalanobis não os detecta satisfatoriamente. Isto ocorre porque tal grupo de novidades ou *outliers* está sujeitos aos efeitos de **ocultação** (*masking*) e **arrastão** (*swamping*)[3].

O **efeito ocultação** ocorre quando uma novidade oculta uma segunda novidade que está próxima dela. De modo que a última pode ser detectada somente após a eliminação da primeira novidade. Em termos matemáticos, o fenômeno da ocultação ocorre quando um grupo de novidades atrai as estimativas de média e da matriz de covariância na direção dele, e a distância $D_M$ resultante da novidade ao centróide fica pequena.

O **efeito arrastão** ocorre quando uma novidade "arrasta" uma outra observação usual como novidade, se a última for considerada novidade somente sob a presença da primeira. Em outras palavras, se a primeira novidade for eliminada, a outra observação pode transformar-se numa amostra "usual". O fenômeno do arrastão ocorre quando um grupo de novidades atrai as estimativas de média e da matriz de covariância da massa de dados para ele e afasta das demais observações "usuais", e a distância resultante destas observações "usuais" ao centróide aumenta, fazendo com que elas sejam confundidas com novidades.

A título de ilustração destes fenômenos, na Figura 2.6 utilizou-se a distância de Mahalanobis e o teste do valor-$p$ para detectar novidades na base de dados de Hawkins et al. (1984). Esta conhecida base de dados é composta de 75 vetores $\mathbf{x} \in \Re^4$ dos quais 14 são alterados para serem novidades ou mais precisamente, *outliers*. Observa-se na figura que

[3]Não há na literatura técnica em língua portuguesa uma tradução formal para estes termos *masking* e *swamping*. Esta dissertação emprega os vocábulos **ocultação** e **arrastão** oriundos de livre tradução realizada pelo autor.

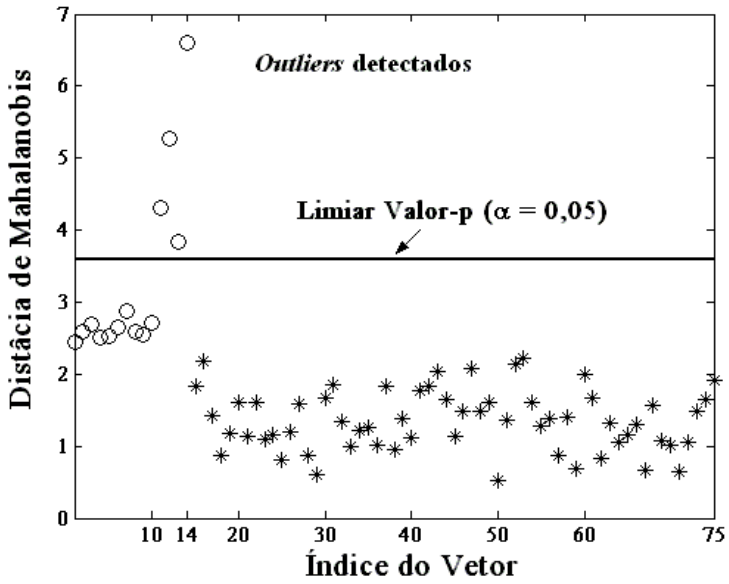


**Figura 2.6:** Exemplo de ocultação de *outliers*. *Outliers* reais são indicados por '○' e dados normais por '*'.

apenas 4 dos 14 são detectados, devido ao efeito ocultação.

Ambos os efeitos mencionados podem ser resolvidos empregando-se estimativas robustas do centróide e da matriz de covariância, que pela definição, são menos afetadas por *outliers*, uma vez que é menos provável que eles sejam inseridos no cálculo das estatísticas robustas, tendo, assim, menor influência sobre as estimativas dos parâmetros usados no cálculo da distância de Mahalanobis.

Entre os estimadores robustos do centróide e da matriz de covariância estão o mínimo determinante da matriz de covariância (*Minimum Covariance Determinant – MCD*) e o Elipsóide de mínimo volume (*Minimum Volume Ellipsoid – MVE*), ambos introduzidos por Rousseeuw & Leroy (1996). Uma análise mais detalhada dos métodos robustos foge do escopo desta dissertação, mas pode ser encontrada nos trabalhos de Hardin & Rocke (2004), Adrover (1998) e Atkinson (1994).

## 2.6 Detecção de Novidades Usando Redes Neurais Artificiais

Mais recentemente, a tecnologia de redes neurais artificiais tem sido bastante utilizada para a detecção de novidades (MARKOU; SINGH, 2003b), principalmente devido à sua habilidade de tratar problemas não-lineares e de algumas propriedades, tais como capacidade de aprendizado e generalização. Redes neurais são capazes de encontrar discri-

minantes não-lineares a partir dos quais pode-se separar os vetores usuais das novidades. Tais discriminantes, em geral, não computam explicitamente o centróide nem a matriz de covariância dos dados. Estas características tornam as arquiteturas de redes neurais menos propensas aos fenômenos de ocultação e arrastão.

A comparar as redes neurais aos métodos estatísticos tradicionais, algumas questões ligadas à detecção de novidades são mais críticas, tais como sua habilidade de generalização e o custo computacional das mesmas. Neste sentido, algumas redes são mais indicadas que outras, a depender da aplicação de interesse.

## 2.7 Conclusão

Este capítulo foi dedicado à definição mais formal do problema de detecção de novidades, realçando as semelhanças e diferenças principais com a detecção de *outliers*. Um ajuste na abordagem referente aos dados faz com que se possa tratar a detecção de novidades usando técnicas de detecção de *outliers*.

Os principais métodos estatísticos de detecção de *outliers* forão descritos. Estes métodos, de forma geral, funcionam como testes de hipóteses, que podem ser dois tipos: unilaterais e bilaterais.

Mostrou-se também que novidades e *outliers* podem ser univariados ou multivariados. Em geral, não se recomenda procurar por novidades ou *outliers* multivariados apenas observando o comportamento de suas componentes. Assim, há a necessidade de testes específicos para a detecção de novidades (*outliers*) multivariadas, por exemplo, através do uso da distância de Mahalanobis.

Para resolver algumas dificuldades encontradas nos métodos clássicos de detecção de novidades, tais como a necessidade de inversão da matriz de covariância ou de se supor uma distribuição paramétrica para os dados e, ainda, os efeitos ocultação e arrastão, sugeriu-se o uso de redes neurais artificiais. O capítulo seguinte será dedicado à apresentação das redes neurais que têm sido usadas para tratar o problema de detecção de novidades.

# 3 Redes Neurais Artificiais para Detecção de Novidades

## 3.1 Introdução

Este capítulo tem por objetivo apresentar sucintamente as arquiteturas de redes neurais avaliadas nesta dissertação, como forma de facilitar a compreensão dos métodos de detecção de novidades que são propostos nos capítulos seguintes. As descrições das arquiteturas apresentadas neste capítulo são baseadas nos livros de Kohonen (2001), Haykin (1999) e Kosko (1992). Referências adicionais são citadas quando necessárias.

Redes neurais artificiais podem ser, de forma geral, divididas em duas categorias: redes com aprendizado supervisionado e redes com aprendizado não-supervisionado. No caso supervisionado, cada entrada apresentada à rede vem acompanhada de uma resposta (saída) desejada e, então, os pesos sinápticos da rede são ajustados, de forma que a saída seja a mais próxima possível daquela desejada. No caso não-supervisionado, a rede neural detecta padrões e características estatísticas do espaço de entrada, de forma a construir uma representação compacta desse espaço no conjunto de pesos sinápticos de seus neurônios.

As arquiteturas de redes neurais descritas neste capítulo são listadas a seguir:

- **Redes não-supervisionadas:** *Winner-Take-All* (WTA), *Frequency-Sensitive Competitive Learning* (FSCL), *Self-Organizing Map* (SOM) e *Neural-Gas* (NGA).
- **Redes supervisionadas:** *Multilayer Perceptron* (MLP), *Radial Basis Functions* (RBF) , MLP autoassociador (AAMLP) e Filtro Linear Detector de Novidades.

As arquiteturas não-supervisionadas são discutidas em primeiro lugar, em seguida, as arquiteturas supervisionadas.

## 3.2 Redes Neurais Não-Supervisionadas Competitivas

Redes neurais não-supervisionadas tentam extrair características estatísticas predominantes nos dados de entrada e constroem, de forma auto-organizada (i.e. sem auxílio externo e sem conhecimento prévio), uma representação reduzida do espaço de entrada, codificando-a em seus pesos sinápticos.

Para utilizar uma rede neural não-supervisionada é preciso ter em mãos um número finito de $N$ exemplos de treinamento, cada um deles representado como um vetor $\mathbf{x}(t) \in \mathbb{R}^n$, ou seja

$$\mathbf{x}(t) = \begin{pmatrix} x_1(t) \\ \vdots \\ x_n(t) \end{pmatrix}, \tag{3.1}$$

em que $t = 1, 2, \ldots, N$, indica o instante de apresentação deste vetor à rede durante o treinamento da mesma.

Cada componente $x_i(t)$ carrega alguma informação relevante para a análise em questão, sendo denominada normalmente de **característica** ou **atributo**. Dessa forma, um vetor $\mathbf{x}(t)$ é, normalmente, chamado de **vetor de características** ou **vetor de atributos** no contexto de reconhecimento de padrões (WEBB, 2002). Através do mapeamento dessas características é que as redes não-supervisionadas podem construir sua própria representação do espaço de entrada. Parte dos métodos propostos neste trabalho utilizam um subgrupo das redes neurais não-supervisionadas, chamadas **redes neurais competitivas**.

Redes competitivas constituem uma das principais classes de redes neurais artificiais (RNAs), nas quais um único neurônio ou um pequeno grupo deles, chamados *neurônios vencedores*, são ativados de acordo com o grau de proximidade entre seus vetores de pesos e o vetor de entrada atual, grau este medido segundo alguma métrica (HAYKIN, 1999). Esse tipo de algoritmo é comumente utilizado em tarefas de reconhecimento e classificação de padrões, tais como formação de agrupamentos (*clustering*), quantização vetorial e classificação de padrões. Nestas aplicações, o vetor de pesos associado ao neurônio vencedor é visto como um *protótipo* representativo de um determinado grupo de vetores de entrada. A Figura 3.1 ilustra a arquitetura geral das redes neurais competitivas tratadas nesta dissertação.

Os modelos neurais competitivos avaliados neste trabalho são baseados na distância euclidiana como métrica utilizada para a determinação do neurônio vencedor. Neste caso, tem-se que um certo neurônio $i$ é escolhido como o neurônio vencedor, simbolizado como

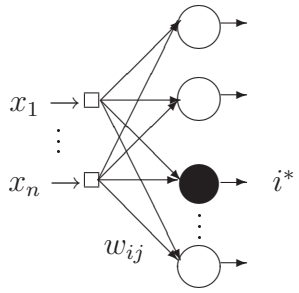


**Figura 3.1:** Arquitetura geral de uma rede neural competitiva.

$i^*(t)$, para o vetor de entrada atual, se a seguinte relação for satisfeita:

$$i^*(t) = \arg\min_{\forall i}\{\|\mathbf{x}(t) - \mathbf{w}_i(t)\|\} \tag{3.2}$$

na qual $i^*(t)$ é o índice do neurônio vencedor na rede, $\mathbf{x}(t)$ é um vetor de entrada da rede na iteração $t$ e $\mathbf{w}_i(t) \in \mathbb{R}^n$, é o vetor de pesos associado ao neurônio $i$. Os algoritmos descritos a seguir têm sua operação baseada na Equação (3.2).

### 3.2.1 Rede WTA

No algoritmo competitivo mais simples, conhecido como *Winner-take-all* (WTA), durante a fase de teinamento apenas o neurônio vencedor tem seu vetor de pesos $\mathbf{w}_{i^*}(t)$, atualizado em resposta a um dado vetor de entrada $\mathbf{x}(t)$. O treinamento da rede WTA é resumido a seguir

1. seleção aleatória de um exemplo de treinamento $\mathbf{x}(t)$ como vetor de entrada atual.
2. busca pelo neurônio vencedor, $i^*(t)$, para o vetor de entrada $\mathbf{x}(t)$, usando a Equação (3.2).
3. atualização do vetor de pesos do neurônio vencedor, $\mathbf{w}_i^*(t)$, pela equação

$$\mathbf{w}_{i^*}(t+1) = \mathbf{w}_{i^*}(t) + \eta[\mathbf{x}(t) - \mathbf{w}_{i^*}(t)], \tag{3.3}$$

em que $0 < \eta < 1$ denota o passo de aprendizagem. Em geral, os valores iniciais dos pesos são atribuídos de forma aleatória e eqüiprovável dentro do intervalo $[0, 1]$. Alternativamente, os vetores de pesos iniciais podem ser selecionados a partir do próprio conjunto de vetores de treinamento.

Pode-se mostrar que o vetor de pesos de um determinado neurônio $i$ converge para o centróide (centro de gravidade) do conjunto de vetores de treinamento para o qual o neurônio $i$ é selecionado vencedor (HAYKIN, 1999). Para tanto, basta perceber que os

valores esperados de $\mathbf{w}_i(t+1)$ e $\mathbf{w}_i(t)$ são iguais para $t \to \infty$. Daí, simbolizando por $\mathbf{w}_i^o$ o valor final do vetor de pesos $\mathbf{w}_i$, a seguinte expressão pode ser obtida

$$E\left\{\eta[\mathbf{x} - \mathbf{w}_i^o]\right\} = 0 \qquad \Rightarrow \qquad \mathbf{w}_i^o = \frac{\int_{V_i} \mathbf{x} p(\mathbf{x}) d\mathbf{x}}{\int_{V_i} p(\mathbf{x}) d\mathbf{x}}, \tag{3.4}$$

em que $V_i$ é o conjunto de vetores de treinamento para o qual o neurônio $i$ é selecionado vencedor.

A fim de aumentar a probabilidade de convergência do algoritmo para um mínimo global, é comum fazer com que o passo de aprendizagem decresça com o tempo. Em todas as redes competitivas adotadas nesta dissertação, utiliza-se um decaimento exponencial dado por

$$\eta(t) = \eta_0 \left(\frac{\eta_T}{\eta_0}\right)^{(t/T)}, \tag{3.5}$$

tal que $\eta_0$ e $\eta_T$ ($\eta_T \ll \eta_0$) são os valores inicial e final de $\eta$. A velocidade de decaimento é controlada pelo parâmetro $T$, que simboliza o número máximo de iterações de treinamento.

A despeito de sua simplicidade, a rede WTA é afetada por algumas questões que comprometem seriamente seu desempenho. A primeira delas diz respeito à **escolha dos valores iniciais dos pesos da rede**, pois dependendo desses valores alguns neurônios podem dominar o treinamento, sendo sempre selecionados como vencedores, enquanto outros nunca o são. As unidades não selecionadas são chamadas de **unidades mortas** (*dead units*) (HERTZ et al., 1991). A outra questão refere-se ao fato de haver uma **valorização excessiva da informação contida na entrada $\mathbf{x}(t)$ mais recente**, já que, pela própria natureza do algoritmo, as entradas apresentadas à rede no início do treinamento têm menos influência no valor final dos pesos dos neurônios que aquelas apresentadas ao final do treinamento.

Para minimizar os problemas descritos acima, é comum modificar o algoritmo da rede WTA, criando variantes mais eficientes. As principais maneiras de se fazer isso são a alteração da Equação (3.2) ou da Equação (3.3). Algumas destas modificações dão origem aos três algoritmos seguintes.

### 3.2.2 Rede FSCL

O primeiro algoritmo, chamado *Frequency-Sensitive Competitive Learning* (FSCL) altera a Equação (3.2), penalizando neurônios que são escolhidos vencedores com muita freqüência, de modo a permitir vitórias de outros neurônios (AHALT et al., 1990)

$$i^*(t) \quad = \quad \arg\min_{\forall i}\{f_i(t) \cdot \|\mathbf{x}(t) - \mathbf{w}_i(t)\|\} \tag{3.6}$$

$$f_i(t) = \left[\frac{c_i}{t}\right]^z, \tag{3.7}$$

em que $c_i$ é o número de vezes que o neurônio $i$ é escolhido vencedor até o instante $t$, e $z > 1$ é uma constante. O ajuste dos pesos continua sendo feito de acordo com a Equação (3.3). Nota-se que a presença de $f_i(t)$, como elemento ponderador da distância euclidiana, ajuda a minimizar a ocorrência de unidades mortas.

### 3.2.3 Rede SOM

O segundo algoritmo, é chamado *Self-Organizing Map* (SOM), proposto por Kohonen (2001). Esta rede difere das redes competitivas anteriormente descritas pelo fato de seus neurônios estarem dispostos em uma grade fixa, geralmente uni- ou bi-dimensional, de modo que se possa definir uma relação de **vizinhança** espacial entre neurônios desta grade. Assim, a Equação (3.3) é alterada pela inserção do conceito de vizinhança, que é o conjunto de neurônios que estão em torno do neurônio vencedor $i^*(t)$. Durante o treinamento, os vetores de pesos dos neurônios na vizinhança do neurônio vencedor também passam a ser ajustados, de acordo com a seguinte regra de aprendizagem

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \eta(t)h(i^*, i; t)[\mathbf{x}(t) - \mathbf{w}_i(t)], \tag{3.8}$$

em que $h(i^*, i; t)$ é a função de vizinhança, geralmente do tipo gaussiana, ou seja:

$$h(i^*, i; t) = \exp\left(-\frac{\|\mathbf{r}_i(t) - \mathbf{r}_{i^*}(t)\|^2}{\vartheta^2(t)}\right), \tag{3.9}$$

em que $\vartheta(t)$ define o raio de influência da função de vizinhança, enquanto $\mathbf{r}_i(t)$ e $\mathbf{r}_{i^*}(t)$ são, respectivamente, as posições dos neurônios $i$ e $i^*$ no arranjo geométrico da rede.

A função de vizinhança funciona como uma espécie de janela de ponderação, fazendo com que os neurônios mais próximos do neurônio vencedor atual tenham seus vetores de pesos atualizados mais intensamente que aqueles neurônios que estão mais distantes do neurônio vencedor. O neurônio vencedor tem seus pesos reajustados com maior intensidade, visto que para ele tem-se $h(i^*, i; t) = 1$. Para todos os outros neurônios, tem-se $h(i^*, i; t) < 1$.

Por questões de convergência e estabilização do aprendizado, a função de vizinhança deve decrescer no tempo, ou seja, o raio de influência $\vartheta(t)$ decai com o decorrer do treinamento de modo semelhante à Equação (3.5)

$$\vartheta(t) = \vartheta_0 \left(\frac{\vartheta_T}{\vartheta_0}\right)^{(t/T)}, \tag{3.10}$$

tal que $\vartheta_0$ e $\vartheta_T$ ($\vartheta_T \ll \vartheta_0$) são os valores inicial e final de $\vartheta$. Em suma, a Equação (3.10)

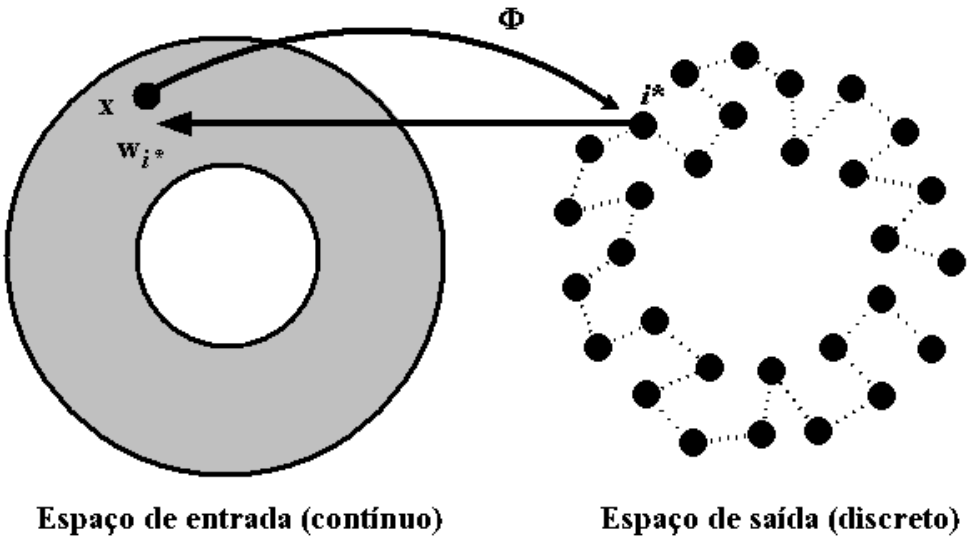


**Figura 3.2:** Projeção implementada pela rede SOM.

faz com que a vizinhança diminua com o passar das iterações de treinamento.

É importante enfatizar que se os neurônios da rede SOM estão dispostos em uma grade uni-dimensional, tem-se que $\mathbf{r}_i(t) \in \mathbb{R}$, ou seja, a posição de um neurônio $i$ qualquer coincide com seu próprio índice, $\mathbf{r}_i(t) = i$. Neste caso, cada neurônio possui apenas vizinhos à direita e à esquerda. Contudo, se os neurônios da rede SOM estão dispostos em uma grade bidimensional, tem-se que $\mathbf{r}_i(t) \in \mathbb{R}^2$, ou seja, a posição de um neurônio $i$ na grade é dada pelas coordenadas $(x_i, y_i)$ em relação a uma origem pré-fixada. Neste caso, um neurônio pode ter vizinhos à esquerda, à direita, acima, abaixo e diagonalmente.

Em razão de sua arquitetura peculiar e de seu algoritmo de treinamento, a rede SOM implementa uma projeção não-linear $\Phi$ do espaço de entrada contínuo $\chi \subset \mathbb{R}^n$ (espaço dos dados), em um espaço de saída discreto $\mathcal{A}$, representado pelo espaço das coordenadas dos neurônios na grade, tal que $\dim(\mathcal{A}) \ll n$. Matematicamente, esta projeção pode ser simbolizada por

$$\Phi{:}\chi \rightarrow \mathcal{A}. \tag{3.11}$$

A rede SOM tem tido grande utilização em aplicações de mineração de dados e reconhecimento de padrões. Grande parte do seu sucesso se deve à combinação de dois princípios essenciais de **auto-organização de sistemas** (MALSBURG, 2003): competição entre neurônios por recursos limitados, implementada pela Equação (3.2); e cooperação, implementada pela função vizinhança. O resultado da atuação destes dois princípios na rede SOM é uma projeção $\Phi$ que preserva relações de proximidade espacial entre os dados de entrada, ou seja, o mapeamento preserva a topologia do espaço de entrada no espaço de saída (HAYKIN, 1999), conforme ilustrado na Figura (3.2, na qual $\dim(\chi) = n = 2$

e $\dim(\mathcal{A}) = 1$, os pontos pretos correspondem às coordenadas dos vetores de pesos do $i$-ésimo neurônio. Neurônios que são vizinhos na grade uni-dimensional são conectados por linhas tracejadas.

Pode-se expressar a propriedade de preservação de topologia da rede SOM da seguinte forma (HERTZ et al., 1991). Sejam $\mathbf{x}_1$ e $\mathbf{x}_2$ dois vetores no espaço de entrada $\chi$, $\mathbf{r}_{i_1^*}$ e $\mathbf{r}_{i_2^*}$ as coordenadas dos neurônios vencedores para $\mathbf{x}_1$ e $\mathbf{x}_2$, respectivamente. Diz-se que a rede SOM, corretamente treinada, preserva a topologia do espaço de entrada se a seguinte relação for observada

$$\|\mathbf{x}_1 - \mathbf{x}_2\| \to 0 \qquad \Rightarrow \qquad \|\mathbf{r}_{i_1^*} - \mathbf{r}_{i_2^*}\| \to 0 \tag{3.12}$$

ou seja, se quaisquer dois vetores estão fisicamente próximos no espaço de entrada, então eles terão neurônios vencedores espacialmente próximos na rede.

Devido à propriedade de preservação de topologia, a rede SOM é capaz de construir uma **aproximação do espaço de entrada**, ou seja, ela constrói uma aproximação discreta do espaço de entrada, na qual cada neurônio da rede representa uma determinada região do espaço de entrada que define sua **região de atração** ou **campo receptivo**. Esta região é conhecida também como **célula de Voronoi**. Assim, uma das principais aplicações da rede SOM é a categorização de dados não-rotulados em agrupamentos (*clusters*) e sua posterior utilização na classificação de vetores de características que não estavam presentes durante o treinamento.

A propriedade de preservação de topologia permite à rede SOM fazer uma **estimação pontual da função densidade de probabilidade**, o que siginifica que o mapeamento da rede SOM reflete variações na estatística do espaço de entrada, ou seja, regiões no espaço de entrada $\chi$ nas quais as observações $\mathbf{x}$ têm uma alta probabilidade de ocorrência são povoadas com um maior número de neurônios, possuindo, conseqüentemente, uma melhor resolução do que regiões em $\chi$ nas quais as observações $\mathbf{x}$ são retiradas com baixa probabilidade de ocorrência.

### 3.2.4 Rede Neural-Gas

O algoritmo Neural-Gas (NGA) (MARTINETZ; SCHULTEN, 1991) modifica simultaneamente as Equações (3.2) e (3.3). Não se busca diretamente um único vencedor, mas sim o ordenamento de todos eles de forma crescente das distâncias euclidianas dos respectivos vetores de pesos à entrada atual $\mathbf{x}(t)$. Desta forma o neurônio vencedor $i_1$ é o primeiro da lista, o segundo mais próximo $i_2$, e assim até o neurônio $i_N$, cujos pesos estão mais

distantes da entrada, genericamente

$$\|\mathbf{x}(t) - \mathbf{w}_{i_1}(t)\| < \cdots < \|\mathbf{x}(t) - \mathbf{w}_{i_k}(t)\| < \ldots < \|\mathbf{x}(t) - \mathbf{w}_{i_N}(t)\|. \tag{3.13}$$

Então, o ajuste dos pesos é feito segundo a equação

$$\mathbf{w}_i(t+1) = \mathbf{w}_i(t) + \eta(t) h_\lambda(k,t)[\mathbf{x}(t) - \mathbf{w}_i(t)], \tag{3.14}$$

na qual $h_\lambda(k,t)$ funciona de modo equivalente à função vizinhança da rede SOM, sendo definida pela equação

$$h_\lambda(k,t) = \exp\left\{-\frac{k-1}{\lambda(t)}\right\}, \tag{3.15}$$

com $k$ representando a posição do neurônio na lista ordenada definida pela Equação (3.13). A variável $\lambda(t)$ decai com o tempo, como na Equação (3.5), equivalendo-se ao conceito de largura da vizinhança da rede SOM.

A rede Neural-Gas também é capaz de preservar a topologia do espaço de entrada no espaço dos pesos dos neurônios. Contudo, esta propriedade é alcançada sem que os seus neurônios sejam dispostos segundo o arranjo geométrico fixo da rede SOM.

### 3.2.5 Vantagens das Redes Neurais Competitivas

Uma das vantagens das redes competitivas é que elas são capazes de realizar quantização vetorial (GRAY, 1984), que pode ser entendida como uma estimação pontual da função densidade de probabilidade dos dados de entrada, codificada pelos vetores de pesos.

A quantização vetorial consiste numa compressão de dados na qual, naturalmente, ocorre alguma perda de informação que pode ser avaliada por uma medida denominada *erro de quantização*. O vetor de erros de quantização, $\mathbf{e}_q(t)$, indica a qualidade da estimação, sendo definido como a diferença entre o vetor de entrada atual e o vetor de pesos do neurônio vencedor correspondente, ou seja,

$$\mathbf{e}_q(t) = \mathbf{x}(t) - \mathbf{w}_{i^*}(t), \tag{3.16}$$

sobre o qual opera uma medida de distância, em geral, euclidiana (PRINCIPE et al., 2000), dando origem a uma grandeza escalar denominada **erro de quantização** associado ao vetor $\mathbf{x}(t)$

$$e_q[\mathbf{x}(t)] = \|\mathbf{x}(t) - \mathbf{w}_{i^*}(t)\| = \|\mathbf{e}_q(t)\| = \sqrt{\sum_{j=1}^{n}[x_j(t) - w_{i^*j}(t)]^2}, \tag{3.17}$$

na qual $n$ é a dimensão de $\mathbf{x}(t)$. A Figura 3.3 mostra o *vetor erro de quantização* $\mathbf{e}_q$, cujo

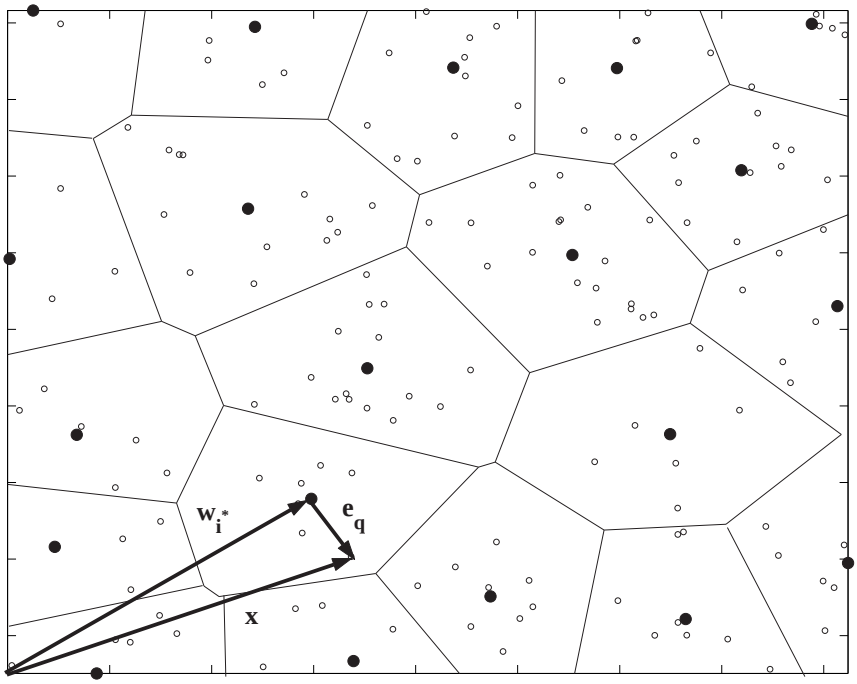


**Figura 3.3:** Ilustração do vetor erro de quantização $\mathbf{e}_q$. Os círculos abertos ('○') simbolizam os vetores de dados, enquanto os círculos fechados ('●') simbolizam os vetores de pesos (centróides).

módulo corresponde ao erro de quantização definido na Equação (3.17).

Duas importantes vantagens oferecidas pelas redes competitivas para o problema de detecção de novidades. A primeira é a **compressão de dados**, uma vez que, após a estabilização da rede com um limiar de erro de quantização aceitável, seus pesos podem ser usados em vez dos próprios dados, a quantidade de vetores de pesos (protótipos) é bastante reduzida em relação à quantidade de dados, portanto, tem-se uma redução considerável de esforço computacional. Além de reduzir o custo computacional, trabalhar com os protótipos aumenta a robustez do algoritmo, visto que os protótipos extraem qualidades estatísticas médias, filtrando flutuações aleatórias que porventura estejam presentes nos dados originais. Isso pode ser verificado através da iteração da Equação (3.3). A segunda é a **simplificação dos critérios de qualidade**, pois todos os testes podem ser realizados utilizando-se como critério o erro de quantização.

A partir da próxima Seção são apresentadas as arquiteturas de redes neurais supervisionadas avaliadas nesta dissertação.

## 3.3 Redes Neurais Supervisionadas

Nas redes supervisionadas existe uma exigência quanto à saída que ela deve apresentar (saída desejada). O treinamento de tais redes não é terminado enquanto não seja alcançado um nível aceitável de semelhança entre a saída atual da rede e aquela desejada. Estas são as redes mais populares e de maior uso em aplicações práticas, devido ao seu bom desempenho em tarefas de aproximação de funções e classificação de padrões, fruto da combinação de propriedades computacionais importantes, tais como não-linearidade, capacidade de aprendizado e generalização.

Redes supervisionadas necessitam de informação externa, $\mathbf{d}(t) \in \mathbb{R}^m$, que indique a saída desejada para cada vetor de entrada, $\mathbf{x}(t) \in \mathbb{R}^{(n+1)}$. Esta informação é fornecida através de pares de vetores $\{\mathbf{x}(t), \mathbf{d}(t)\}$, $t = 1, 2, \ldots, N$. Cada vetor de entrada é representado como

$$\mathbf{x}(t) = \begin{pmatrix} x_0(t) \\ x_1(t) \\ \vdots \\ x_n(t) \end{pmatrix} = \begin{pmatrix} -1 \\ x_1(t) \\ \vdots \\ x_n(t) \end{pmatrix} \tag{3.18}$$

em que o índice $t$, indica a ordem de apresentação deste vetor à rede durante o treinamento. A entrada $x_0(t) = -1$ é mantida sempre fixa, sendo usada com o objetivo de permitir uma formulação única para o ajuste dos limiares de ativação dos neurônios nas redes supervisionadas. O vetor de saída é comumente representado da seguinte forma

$$\mathbf{d}(t) = \begin{pmatrix} d_1(t) \\ \vdots \\ d_m(t) \end{pmatrix} \tag{3.19}$$

e representa o vetor de saídas desejadas associado ao vetor de entrada atual. Ainda, $x_j(t)$ denota uma componente qualquer do vetor de entrada $\mathbf{x}(t)$ e $d_k(t)$ denota a $k$-ésima componente do vetor de saídas desejadas $\mathbf{d}(t)$.

Assume-se que os vetores, $\mathbf{x}(t)$ e $\mathbf{d}(t)$ estão relacionados segundo alguma lei de causa-e-efeito desconhecida, $\mathbf{F}(\cdot)$, ou seja,

$$\mathbf{d}(t) = \mathbf{F}[\mathbf{x}(t)], \tag{3.20}$$

sendo que se deseja simular o comportamente de $\mathbf{F}[\cdot]$, com base apenas nos pares de vetores $\{\mathbf{x}(t), \mathbf{d}(t)\}$ disponíveis. Para isto pode-se utilizar uma rede neural supervisionada para gerar uma **aproximação** de $\mathbf{F}[\cdot]$, denotada por $\hat{\mathbf{F}}[\cdot]$, tal que

$$\mathbf{y}(t) = \hat{\mathbf{F}}[\mathbf{x}(t)], \tag{3.21}$$

na qual $\mathbf{y}(t)$ é a saída gerada pela rede neural que, espera-se, seja muito próxima da saída desejada $\mathbf{d}(t)$.

É de amplo conhecimento que redes neurais supervisionadas multicamadas são **aproximadores universais de funções** (PRINCIPE et al., 2000; HAYKIN, 1999; HERTZ et al., 1991), ou seja, são capazes de aproximar mapeamentos entrada-saída, contínuos ou descontínuos, com grau de precisão arbitrário. Esta propriedade é uma das responsáveis pela ampla popularização do uso de redes neurais artificiais em tarefas de reconhecimento de padrões. Assim, vale destacar que a formulação geral apresentada anteriormente se aplica a arquiteturas de redes neurais envolvidas tanto em tarefas de aproximação de funções, quanto em tarefas de classificação de padrões.

As redes supervisionadas se adaptam às diversas aplicações pela variação de suas arquiteturas, nas próximas seções apresentam-se algumas que têm sido bastante utilizadas no tratamanto do problema geral de detecção de novidades.

### 3.3.1 Filtro Linear Detector de Novidades

Um dos primeiros métodos empregados para a detecção de novidades, chamado **Filtro de Novidades**, é proposto por Kohonen & Oja (1976). Seu desenvolvimento matemático é um caso especial da **Memória Associativa Linear Ótima** (*Optimal Linear Associative Memory - OLAM*) (KOHONEN, 1989), na qual se tenta obter um mapeamento linear $\mathbf{d} = \mathbf{Ax}$, a partir de um conjunto finito de pares entrada-saída $\{\mathbf{x}(t), \mathbf{d}(t)\}$, $t = 1, \ldots, N$.

Para a detecção de novidades, interessa-se apenas na versão auto-associativa (*autoassociative OLAM*). Neste caso, o par $\{\mathbf{x}(t), \mathbf{d}(t)\}$ torna-se o par redundante $\{\mathbf{x}(t), \mathbf{x}(t)\}$. Assim, dado um conjunto de vetores $\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(N)$, é possível calcular a matriz $\mathbf{A}$ da seguinte forma

$$\mathbf{A} = \mathbf{X}^*\mathbf{X}, \tag{3.22}$$

em que as colunas de $\mathbf{X}$ são os vetores de treinamento $\mathbf{x}(t)$ e

$$\mathbf{X}^* = \mathbf{X}^T(\mathbf{X}\mathbf{X}^T)^{-1}, \tag{3.23}$$

em que $\mathbf{X}^*$ denota a matriz pseudo-inversa de $\mathbf{X}$ (HOWARD, 2001).

Considere que os vetores conhecidos $\mathbf{x}(t)$ descrevem algum subespaço linear único $\mathcal{L}[\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(N)]$ do $\Re^n$, ou alternativamente,

$$\mathcal{L} = \mathcal{L}[\mathbf{x}(1), \mathbf{x}(2), \ldots, \mathbf{x}(N)] = \left\{ \mathbf{x} \in \Re^n \mid \mathbf{x} = \sum\nolimits_{t=1}^{N} a(t)\mathbf{x}(t) \right\}, \tag{3.24}$$

em que $a(1), \ldots, a(N)$ são valores reais arbitrários pertencentes ao domínio $(-\infty, \infty)$.

Pode-se mostrar que a matriz $\mathbf{A}$ comporta-se como um **operador de projeção**. O operador $\mathbf{A}$ projeta $\Re^n$ em $\mathcal{L}$. Existe um outro operador, chamado **operador dual** que projeta $\Re^n$ em $\mathcal{L}^\perp$, que é o subespaço ortogonal complementar $\{\mathbf{v} \in \Re^n \mid \mathbf{v}^T\mathbf{z} = 0, \forall \mathbf{z} \in \mathcal{L}\}$.

Pode-se mostrar que o operador dual é dado por $\mathbf{I}_n - \mathbf{X}^*\mathbf{X}$, na qual $\mathbf{I}_n$ denota a matriz identidade $n \times n$. Assim, cada vetor em $\Re^n$ pode ser unicamente decomposto da seguinte forma

$$\mathbf{x}(t) = (\mathbf{X}^*\mathbf{X})\mathbf{x}(t) + (\mathbf{I}_n - \mathbf{X}^*\mathbf{X})\mathbf{x}(t) = \hat{\mathbf{x}}(t) + \tilde{\mathbf{x}}(t), \tag{3.25}$$

em que a projeção $\hat{\mathbf{x}}(t)$ é uma medida do que é conhecido sobre a entrada $\mathbf{x}(t)$ com base nos vetores de treinamento armazenados na matriz $\mathbf{A}$ conforme mostrado na Equação (3.22).

Por sua vez, a projeção $\tilde{\mathbf{x}}(t)$ é chamada de **vetor novidade**, uma vez que ele é uma medida do que é maximamente desconhecido sobre o vetor de entrada $\mathbf{x}(t)$. Assim, o módulo de $\tilde{\mathbf{x}}(t)$ pode ser usado para fins de detecção de novidades. Nessas aplicações, quanto maior a norma $||\tilde{\mathbf{x}}(t)||$, menos certeza se tem ao afirmar que o vetor $\hat{\mathbf{x}}(t)$ é pertencente ao subespaço linear $\mathcal{L}$. Em Kohonen & Oja (1976), o filtro linear de novidades é implementado como uma rede neural adaptativa recorrente (rede com realimentação) e totalmente conectada.

### 3.3.2 Rede Perceptron Multicamada

Tipicamente, uma rede **Perceptron Multicamada** (*Multilayer Perceptron*, MLP) é constituída de uma **camada de entrada** que recebe os sinais, uma ou mais **camadas intermediárias**, compostas por neurônios somadores com função de ativação não-linear e uma **camada de saída**, também composta por neurônios somadores (que podem ser lineares).

A rede contém uma ou mais camadas escondidas, aquelas que não fazem parte nem da entrada nem da saída, são essas camadas que tornam a rede capaz de extrair progressivamente as características mais significativas do espaço de entrada.

Outra característica é que essas redes mostram um alto grau de **conectividade**, determinado pelas sinapses da rede, interligações entre os neurônios das diferentes camadas, em que cada uma delas está associada a um valor numérico chamado **peso sináptico**.

Nas definições e cálculos mostrados a seguir, considera-se uma arquitetura de rede neural do MLP com apenas uma camada escondida de neurônios, treinada com o algoritmo de **retropropagação do erro** (*Error Backpropagation*).

A Figura 3.4 mostra a arquitetura geral das redes supervisionadas utilizadas nesta dissertação, compostas de uma camada escondida e neurônios lineares na saída.

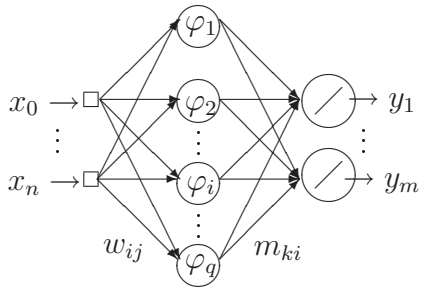


**Figura 3.4:** Arquitetura geral de uma rede neural supervisionada.

O vetor de pesos associado ao $i$-ésimo neurônio da camada escondida, também chamada de **camada oculta** ou **camada intermediária**, é representado como

$$\mathbf{w}_i(t) = \begin{pmatrix} w_{i0}(t) \\ \vdots \\ w_{in}(t) \end{pmatrix} = \begin{pmatrix} \theta_i(t) \\ \vdots \\ w_{in}(t) \end{pmatrix}, \tag{3.26}$$

em que $\theta_i(t)$ é o limiar associado ao neurônio $i$. Os neurônios desta camada são chamados de neurônios escondidos por não terem acesso direto à saída da rede MLP, onde são calculados os erros de aproximação.

De modo semelhante, o vetor de pesos associado ao $k$-ésimo neurônio da camada de saída é representado como

$$\mathbf{m}_k(t) = \begin{pmatrix} m_{k0}(t) \\ \vdots \\ m_{kq}(t) \end{pmatrix} = \begin{pmatrix} \theta_k(t) \\ \vdots \\ m_{kq}(t) \end{pmatrix}, \tag{3.27}$$

na qual $\theta_k(t)$ é o limiar associado ao neurônio de saída $k$. O número de neurônios da camada escondida é simbolizado por $q \geq 2$.

O treinamento da rede MLP se dá em duas fases: uma direta e outra reversa.

**Sentido Direto:** esta etapa de funcionamento da rede MLP envolve o cálculo das ativações e saídas de todos os neurônios da camada escondida e de todos os neurônios da camada de saída. Assim, o fluxo de sinais (informação) se dá dos neurônios de entrada para os neurônios de saída, passando obviamente pelos neurônios da camada escondida. Por isso, diz-se que a informação está se propagando no sentido **direto**, ou seja

Entrada → Camada Intermediária → Camada de Saída.

Assim, após a apresentação de um vetor de entrada $\mathbf{x}$, na iteração $t$, o primeiro

passo é calcular as ativações dos neurônios da camada escondida

$$u_i(t) = \sum_{j=0}^{n} w_{ij}(t)x_j(t) = \mathbf{w}_i^T(t)\mathbf{x}(t), \quad i = 1, \ldots, q \tag{3.28}$$

em que $T$ denota a operação de transposição dos vetores e $q$ indica o número de neurônios da camada escondida. Em seguida, as saídas correspondentes são calculadas por

$$v_i(t) = \varphi_i[u_i(t)] = \varphi_i\left[\sum_{j=0}^{p} w_{ij}(t)x_j(t)\right] = \varphi_i\left[\mathbf{w}_i^T(t)\mathbf{x}(t)\right], \tag{3.29}$$

tal que a função de ativação $\varphi$ assume geralmente uma das seguintes formas

$$\varphi_i[u_i(t)] = \frac{1}{1+\exp[-u_i(t)]}, \qquad \text{(Logística)} \tag{3.30}$$

$$\varphi_i[u_i(t)] = \frac{1-\exp[-u_i(t)]}{1+\exp[-u_i(t)]}, \qquad \text{(Tangente Hiperbólica).} \tag{3.31}$$

O segundo passo consiste em repetir as operações das Equações (3.28) e (3.29) para os neurônios da camada de saída

$$u_k(t) = \sum_{i=0}^{q} m_{ki}(t)v_i(t), \quad k = 1, \ldots, m \tag{3.32}$$

na qual $m \geq 1$ é o número de neurônios de saída.

Em seguida, as saídas dos neurônios da última camada são calculadas pela Equação

$$y_k(t) = \varphi_k[u_k(t)] = \varphi_k\left[\sum_{i=0}^{q} m_{ki}(t)v_i(t)\right], \tag{3.33}$$

tal que a função de ativação $\varphi_k$ assume geralmente uma das formas definidas nas Equações (3.30) e (3.31).

**Sentido Reverso:** esta etapa de funcionamento da rede MLP envolve o cálculo de gradientes locais e o ajuste dos pesos de todos os neurônios da camada escondida e da camada de saída. Assim, o fluxo de informação se dá dos neurônios de saída para os neurônios da camada escondida. Por isso, diz-se que a informação está se propagando no sentido **reverso**, ou seja,

Camada de Saída $\rightarrow$ Camada Escondida.

Assim, após os cálculos das ativações e saídas na fase direta, o primeiro passo da fase reversa consiste em calcular os gradientes locais $\delta_k(t)$ dos neurônios da camada

de saída

$$\delta_k(t) = e_k(t)\varphi'[u_k(t)], \quad k = 1, \ldots, m \tag{3.34}$$

em que $e_k(t)$ é o erro entre a saída desejada $d_k(t)$ para o neurônio $k$ e saída gerada por ele, $y_k(t)$

$$e_k(t) = d_k(t) - y_k(t). \tag{3.35}$$

A derivada $\varphi'[u_k(t)]$ na Equação 3.34 assume diferentes expressões, dependendo da escolha da função de ativação. Assim, tem-se as seguintes possibilidades

$$\varphi'_k[u_k(t)] = y_k(t)[1 - y_k(t)], \quad \text{se } \varphi_k[u_k(t)] \text{ é a função logística} \tag{3.36}$$

$$\varphi'_k[u_k(t)] = \frac{1}{2}[1 - y_k^2(t)], \quad \text{se } \varphi_k[u_k(t)] \text{ é a tangente hiperbólica.} \tag{3.37}$$

O segundo passo da fase reversa consiste em calcular os gradientes locais $delta_i(t)$ dos neurônios da camada escondida

$$\delta_i(t) = \varphi'_i[u_i(t)] \sum_{k=1}^{m} m_{ki}\delta_k(t), \quad i = 1, \ldots, q \tag{3.38}$$

tal que a derivada $\varphi'[u_i(t)]$ é calculada da mesma forma que nas Equações (3.36) e (3.37).

O terceiro passo da fase reversa corresponde ao processo de atualização ou ajuste dos parâmetros (pesos sinápticos e limiares) da rede MLP com uma camada escondida. Assim, para a camada escondida tem-se que a regra de atualização dos pesos, $w_{ij}$, é dada por

$$\begin{aligned} w_{ij}(t+1) &= w_{ij}(t) + \Delta w_{ij}(t) \\ &= w_{ij}(t) + \eta\delta_i(t)x_j(t), \end{aligned} \tag{3.39}$$

em que $\eta$ é a taxa de aprendizagem. E para camada de saída tem-se que a regra de atualização dos pesos, $m_{ki}$, é dada por

$$\begin{aligned} m_{ki}(t+1) &= m_{ki}(t) + \Delta m_{ki}(t) \\ &= m_{ki}(t) + \eta\delta_k(t)y_i(t). \end{aligned} \tag{3.40}$$

Além da rede MLP padrão, há algumas variações desta rede que também são utilizadas em aplicações de detecção de novidades, a saber, a rede **MLP Gaussiana** (*Gaussian MLP*, GMLP) e a rede **MLP Autoassociativa** (*Autoassociative MLP*, AAMLP).

### 3.3.3 Rede MLP Gaussiana - GMLP

Para melhorar o desempenho de MLP nas tarefas de detecção de novidades/*outliers* (VASCONCELOS et al., 1995) sugeriu o uso de uma variante da MLP, rede **MLP Gaussiana** (*Gaussian MLP*, GMLP), proposta originalmente por Dawson & Schopflocher (1992). Nesta rede, utiliza-se uma função de ativação gaussiana para os neurônios da camada escondida

$$\varphi_i[u_i(t)] = \exp\left[\frac{-u_i^2(t)}{\gamma^2}\right]. \tag{3.41}$$

Segundo Vasconcelos et al. (1995), essa simples modificação fornece melhores resultados, devido ao fato que a função de ativação gaussiana força o campo receptivo de um neurônio a ser mais seletivo, sendo ativo somente para uma região estreita do espaço da entrada, uma vez que tende a produzir regiões fechadas que cercam os dados do treinamento.

### 3.3.4 Rede MLP Autoassociativa - AAMLP

A rede MLP é usada também para tarefas da detecção de novidades com uma arquitetura autoassociativa (PETSCHE et al., 1996; JAPKOWICZ et al., 1995). A rede **MLP Autoassociativa** (*Autoassociative MLP*, AAMLP) é projetada para aprender um mapeamento entrada-saída no qual os vetores alvo (saídas desejadas) são os próprios vetores de entrada. Isto é executado, usualmente, com uma camada escondida cujo o número dos neurônios seja mais baixo do que a dimensão dos vetores da entrada, embora essa não seja uma obrigatoriedade.

A rede é treinada para **reconstruir** o melhor possível os vetores de entrada. Neste sentido, o autoassociador pode ser visto como uma versão não-linear do filtro linear detector de novidades apresentado na Seção 3.3.1.

O uso de autoassociadores para detecção de novidades reside na idéia de que ele deve poder reconstruir adequadamente vetores semelhantes aos dados de treinamento, mas deve ter desempenho pobre ao tentar reconstruir padrões diferentes daqueles usados no seu treinamento. Deve-se, portanto, definir uma medida para avaliar quão bom é o mapeamento (ou a reconstrução) provida pelo autoassociador. Uma medida usual é o erro de reconstrução, calculado do seguinte modo

$$e_r(\mathbf{x}, \mathbf{y}) = \|\mathbf{x}(t) - \mathbf{y}(t)\|, \tag{3.42}$$

na qual $\mathbf{x}(t)$ é a entrada (= saída desejada) e $\mathbf{y}(t)$ é a saída real da rede.

Assim, a detecção de novidades ou de padrões de entrada anômalos reduz-se à tarefa

de avaliar o erro de reconstrução para determinar a qualidade do mapeamento não-linear provido pelo autoassociador. Esse procedimento pode consistir em calcular um limite superior para o erro do reconstrução de todos os vetores ao final do treinamento. Para fins de teste, este limite superior é geralmente diminuído por uma determinada porcentagem. Novos padrões de entrada são classificados subseqüentemente, verificando se seus erros de reconstrução estão acima do limite superior calculado, neste caso eles são considerados novidades ou *outliers*.

A seguir é descrita uma outra arquitetura de rede neural supervisionada multicamada, de ampla utilização em detecção de novidades.

### 3.3.5 Rede de Funções de Base Radial - RBF

A rede de **Funções de Base Radial** (*Radial Basis Function*, RBF) é normalmente concebida como sendo composta de uma camada de entrada, uma só camada intermediária, cujos neurônios normalmente têm funções de ativação do tipo gaussiana, e uma camada de saída cujos neurônios são geralmente lineares. Apesar de haver vários modos de se projetar uma rede RBF, este trabalho utiliza um treinamento em duas etapas. Primeiramente, treina-se a primeira camada, que é formada por uma rede SOM. Em seguida, os parâmetros da segunda camada, formada por neurônios lineares, é calculada pelo método dos **Mínimos Quadrados**.

Assim, após a apresentação de um vetor de entrada $\mathbf{x}$ na iteração $t$, calcula-se a saída do $i$-ésimo neurônio da camada intermediária da seguinte forma

$$v_i(t) = \varphi_i[\mathbf{x}(t)] = \exp\left[-\frac{\|\mathbf{x}(t) - \mathbf{c}_i\|^2}{2\gamma_i^2}\right], \tag{3.43}$$

em que o vetor $\mathbf{c}_i$, mantido constante para o neurônio $i$, define o que se chama de **centro** do $i$-ésimo neurônio, enquanto a constante $\gamma_i > 0$ define a largura (abertura) da função de ativação gaussiana deste neurônio.

De acordo com a Equação (3.43), o neurônio $i$ fornece resposta máxima, i.e. $v_i(t) \approx 1$, para vetores de entrada próximos do seu centro $\mathbf{c}_i$. Desta forma, diz-se que cada neurônio da camada escondida tem seu próprio **campo receptivo** no espaço de entrada, que é uma região centrada em $\mathbf{c}_i$ com tamanho proporcional a $\gamma_i$.

Os passos para utilização de uma rede RBF genérica e a determinação de seus os parâmetros são detalhados a seguir.

**Determinação dos Centros da Rede RBF:** uma maneira usual de se determinar os centros $\mathbf{c}_i$ de um neurônio da camada escondida da rede RBF é através dos vetores

de pesos de uma rede SOM já treinada, ou seja, após treinar a rede SOM apenas com o conjunto de vetores de entrada $\{\mathbf{x}(t)\}$, $t = 1, \ldots, N$, considera-se os centros da rede RBF como sendo vetores de pesos dos neurônios de uma rede SOM, fazendo $\mathbf{c}_i = \mathbf{w}_i$ na Equação (3.43).

**Camada de Saída:** após o projeto da camada escondida, prossegue-se com os cálculos das ativações e saídas dos neurônios da camada de saída. Conforme dito anteriormente, esta é composta de neurônios com função de ativação linear. Deste modo, a saída de um neurônio $k$ na camada de saída é dada por

$$y_k(t) = \sum_{i=0}^{q} m_{ki}(t) v_i(t), \quad k = 1, \ldots, m \tag{3.44}$$

em que $m$ é o número de neurônios de saída. Assumiu-se que $v_0(t) = -1$ e $m_{k0} = \theta_k$.

**Cálculo dos Pesos da Camada de Saída da Rede RBF:** o último passo é o cálculo parâmetros (pesos sinápticos e limiares) dos neurônios da camada de saída da rede RBF. Devido ao uso de neurônios lineares na camada de saída, seus pesos podem ser calculados de uma só vez em bloco (*batch*), ou seja, sem lançar mão de equações recursivas, tais como aquelas usadas pela rede SOM e pela rede MLP.

Pode-se equacionar a relação entre as saídas dos neurônios da camada escondida e dos neurônios de saída da seguinte forma matricial

$$\mathbf{GM} = \mathbf{D}, \tag{3.45}$$

tal que $\mathbf{G}$ é uma matriz de dimensões $N \times q$, definida como

$$\mathbf{G} = \begin{pmatrix} \varphi_1[\mathbf{x}(1)] & \varphi_2[\mathbf{x}(1)] & \varphi_3[\mathbf{x}(1)] & \cdots & \varphi_q[\mathbf{x}(1)] \\ \varphi_1[\mathbf{x}(2)] & \varphi_2[\mathbf{x}(2)] & \varphi_3[\mathbf{x}(2)] & \cdots & \varphi_q[\mathbf{x}(2)] \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \varphi_1[\mathbf{x}(N)] & \varphi_2[\mathbf{x}(N)] & \varphi_3[\mathbf{x}(N)] & \cdots & \varphi_q[\mathbf{x}(N)] \end{pmatrix}, \tag{3.46}$$

e $\mathbf{D} = [\mathbf{d}(1) \;\; \mathbf{d}(2) \;\; \cdots \;\; \mathbf{d}(N)]^T$ é uma matriz de dimensões $N \times m$ cujas linhas são os vetores transpostos de saídas desejadas. Assim, a Equação (3.45) pode ser resolvida pelo método dos mínimos quadrados (HAYKIN, 1999; PRINCIPE et al., 2000), encontrando-se os valores dos pesos da camada de saída por meio da seguinte expressão

$$\mathbf{M} = (\mathbf{G}^T\mathbf{G})^{-1}\mathbf{G}^T\mathbf{D}. \tag{3.47}$$

A seguir são discutidos alguns tópicos de interesse prático para aqueles interessados no uso das redes MLP e RBF em problemas de classificação de padrões.

### 3.3.6 Questões práticas sobre as redes MLP e RBF

A convergência da rede MLP é, em geral, avaliada com base nos valores do erro quadrático médio por época de treinamento, aqui simbolizado por $\epsilon$(época):

$$\epsilon(\text{época}) = \frac{1}{N}\sum_{t=1}^{N}\epsilon(t) = \frac{1}{2N}\sum_{t=1}^{N}\sum_{k=1}^{m}e_k^2(t). \tag{3.48}$$

Por outro lado, quando se utiliza a rede MLP para classificar padrões, o desempenho da mesma é avaliado pela **taxa de acerto na classificação**, definida por

$$P\text{época} = \frac{\text{Número de vetores classificados corretamente}}{\text{Número de total de vetores}} \tag{3.49}$$

O gráfico de $\epsilon$(época) ou $P$(época) pelo número de épocas é chamado de **Curva de Aprendizagem** da rede neural.

Em geral, o treinamento da rede neural é parado quando $\epsilon$(época) (ou $P$(época)) atinge um limite inferior considerado adequado para o problema em questão (por exemplo, $\epsilon(\text{época}) \leq 0{,}001$ ou $P(\text{época}) \approx 0{,}95$), ou quando o número máximo de épocas permitido é alcançado.

Para validar a rede treinada, ou seja, dizer que ela está apta para ser utilizada, é importante testar a sua resposta (saída) para dados de entrada diferentes daqueles utilizados durante o treinamento. Estes novos dados podem ser obtidos através de novas medições, o que nem sempre é viável. Durante o teste, os pesos da rede não são ajustados.

Portanto, na fase de validação, o procedimento mais comum consiste em treinar a rede apenas com uma parte dos dados selecionados **aleatoriamente**, guardando a parte restante para ser usada para testar o desempenho da rede. Assim, ter-se-á dois conjuntos de dados, um para treinamento, de tamanho $N_1 < N$, e outro de tamanho $N_2 = N - N_1$. Em geral, escolhe-se $N_1$ tal que a razão $N_1/N$ esteja na faixa de 0,75 a 0,90, ou seja, se $N_1/N \approx 0{,}75$ tem-se que 75% dos vetores de dados devem ser selecionados aleatoriamente, sem reposição, para serem utilizados durante o treinamento. Os 25% restantes são usados para testar a rede. O valor de $\epsilon$ ou da taxa de acerto na classificação calculados para os dados de teste é chamado de **erro de generalização** da rede, pois testa a capacidade da mesma em "extrapolar" o conhecimento aprendido durante o treinamento para novas situações. É importante ressaltar que, geralmente, o erro de generalização é maior do que o erro de treinamento, pois trata-se de um novo conjunto de dados.

Em geral, os procedimentos de treinamento e teste são repetidos por um número $M \gg 1$ de vezes, a fim de se ter uma noção da variabilidade estatística das taxas de acerto. Para cada bateria de treinamento e teste, os elementos que compõem os conjuntos

de treinamento e teste são selecionados aleatoriamente. O valor final da taxas de acerto é dado então pela média das taxas obtidas para as $M$ baterias. O intervalo de confiança da taxa de acerto também pode ser estimado a partir da amostra obtida para as $M$ baterias de treinamento e teste. O mesmo procedimento é levado a cabo caso se queira ter uma noção do valor médio do erro de generalização.

Em particular, para tarefas de detecção de novidades, uma rede neural pode ser avaliada quanto à quantidades de erros do tipo I (falso positivo) e do tipo II (falso negativo) produzidos. A **Taxa de Falsos Positivos** (TFP) e **Taxa de Falsos Negativos** (TFN) são calculadas, respectivamente, a partir das seguintes expressões:

$$\text{TFP}(\%) = 100 \times \frac{\text{Número de vetores comuns classificados como novidades}}{\text{Número de total de vetores comuns}} \tag{3.50}$$

e

$$\text{TFN}(\%) = 100 \times \frac{\text{Número de vetores novidades classificados como comuns}}{\text{Número de total de vetores novidades}}. \tag{3.51}$$

Por fim, algumas técnicas para acelerar a convergência da rede MLP durante o treinamento são listadas a seguir.

**Taxa de aprendizagem variável:** nas Equações (3.39) e (3.40), de ajuste de pesos sinápticos, é interessante que se use uma taxa de aprendizagem variável no tempo, $\eta(t)$, decaindo até um valor suficientemente baixo com o passar das iterações, em vez de mantê-la fixa por toda a fase de treinamento. Duas opções são dadas a seguir

$$\eta(t) = \eta_0 \left(1 - \frac{t}{t_{max}}\right), \quad \text{Decaimento linear,} \tag{3.52}$$

$$\eta(t) = \frac{\eta_0}{1+t}, \quad \text{Decaimento exponencial,} \tag{3.53}$$

em que $\eta_0$ é o valor inicial da taxa de aprendizagem e $t_{max}$ é o número máximo de iterações, ou seja,

$$t_{max} = N \times \text{Número máximo de épocas.} \tag{3.54}$$

A idéia representada nas duas equações anteriores é começar com um valor alto para $\eta$, dado por $\eta_0 \approx 0{,}5$, e terminar com um valor suficientemente pequeno, da ordem de $\eta \approx 0,01$, a fim de estabilizar o processo de aprendizado.

**Termo de momento:** outra melhoria que se pode fazer nas expressões de ajuste de pesos sinápticos, Equações (3.39) e (3.40), é se usar um termo adicional, chamado **termo de momento**, cujo objetivo é tornar o processo de modificação dos pesos mais estável. Com este termo, Equação (3.39) e Equação (3.40) passam a ser escritas da

forma

$$w_{ij}(t+1) = w_{ij}(t) + \eta\delta_i(t)x_j(t) + \xi\Delta w_{ij}(t-1) \quad (3.55)$$

$$m_{ki}(t+1) = m_{ki}(t) + \eta\delta_k(t)y_i(t) + \xi\Delta m_{ki}(t-1), \quad (3.56)$$

em que $\Delta w_{ij}(t-1) = w_{ij}(t) - w_{ij}(t-1)$ e $\Delta m_{ki}(t-1) = m_{ki}(t) - m_{ki}(t-1)$. A constante $0 < \xi < 1$ é chamada **fator de momento**. Enquanto $\eta$ deve ser mantida em um valor abaixo de 0,5 (e.g. $\eta = 0{,}1$) por questões de estabilidade do aprendizado, o fator de momento assume valores na faixa [0,5 0,9].

É importante destacar a demonstração recente feita por Bhaya & Kaszkurewicz (2004) de que a inclusão do fator de momento na equação recursiva de ajuste dos pesos da rede MLP corresponde a uma versão do método de otimização do gradiente conjugado.

**Função Tangente Hiperbólica:** tem sido demonstrado empiricamente, ou seja, através de simulação computacional que o processo de treinamento da rede MLP converge mais rapidamente quando se utiliza a função de ativação tangente hiperbólica do que quando se usa a função logística. A justificativa para isto está no fato da tangente hiperbólica ser uma função ímpar, ou seja, $\varphi(-u_i) = -\varphi(u_i)$.

**Limites menores que os assintóticos:** é interessante notar que os valores limites 0 e 1 para a função logística, ou ($-1$ e $+1$) para a função tangente hiperbólica são valores assintóticos, ou seja, nunca são alcançados na prática. Assim, ao se tentar forçar a saída rede neural para estes valores assintóticos, os pesos sinápticos, $w_{ij}$ e $m_{ki}$ tendem a assumir valores absolutos muito altos, ou seja, $w_{ij} \to \infty$ e $m_{ki} \to \infty$.

Para evitar este problema, sugere-se elevar de um valor pequeno $\epsilon > 0$ (e.g. $\epsilon = 0{,}02$) o limite inferior de $\varphi(\cdot)$ e diminuir deste mesmo valor o limite superior de $\varphi(\cdot)$. Assim, tem-se a seguinte alteração:

$$-1 \to \epsilon - 1 \quad (3.57)$$

$$0 \to \epsilon \quad (3.58)$$

$$1 \to 1 - \epsilon. \quad (3.59)$$

É comum escolher valores dentro da faixa $\epsilon \in [0,01 - 0,05]$ (HAYKIN, 1999).

## 3.4 Conclusão

Este capítulo apresentou sucintamente, porém de forma auto-contida, as arquiteturas de redes neurais a serem avaliadas nesta dissertação. As seguintes arquiteturas são des-

critas neste capítulo:

- **Redes não-supervisionadas:** *Winner-Take-All* (WTA), *Frequency-Sensitive Competitive Learning* (FSCL), *Self-Organizing Map* (SOM) e *Neural-Gas* (NGA).
- **Redes supervisionadas:** *Multilayer Perceptron* (MLP), *Radial Basis Functions* (RBF) , MLP autoassociador (AAMLP) e Filtro Linear Detector de Novidades.

Estas arquiteturas são a base de uma boa parcela das técnicas neurais utilizadas em problemas de detecção de novidades e, por isso, um conhecimento básico sobre o funcionamento destas redes é de grande valia na compreensão dos métodos de detecção de novidades, tanto os revisados quanto os propostos, presentes neste trabalho.

Uma breve exposição sobre como treinar e testar as arquiteturas neurais em tarefas de classificação de padrões também foi apresentada.

O capítulo seguinte é dedicado à apresentação de uma nova técnica de detecção de novidades usando redes neurais competitivas. A nova proposta pode ser entendida como uma combinação de métodos estatísticos clássicos com redes neurais não-supervisionadas competitivas.

# 4 Detecção de Novidades Usando Redes Neurais Competitivas

Este capítulo propõe um novo método para detecção de novidades usando um teste de limiar duplo inspirado na idéia dos intervalos de confiança da estatística, revisados no Capítulo 2. Compara-se o desempenho do método proposto com aquele de um conhecido método de limiar simples (valor-$p$) numa aplicação de detecção de anomalias em redes de comunicação celular. A escolha do método do valor-$p$ como base de comparação se deve ao fato de que, na literatura sobre detecção de novidades, este método é o mais utilizado para o cálculo de limiares. Nos testes comparativos utilizam-se redes neurais competitivas discutidas no Capítulo 3, devido à sua capacidade de compressão da informação. Entretanto, a aplicação do método proposto pode ser estendida aos demais tipos de redes neurais, como é mostrado no Capítulo 5.

## 4.1 Introdução

No Capítulo 2 mostrou-se como a detecção de novidades pode ser tratada por meio de detecção de *outliers* e descreveram-se os métodos clássicos de detecção univariada e multivariada de *outliers*. Observa-se que simplicidade é a principal vantagem das abordagens para detecção de *outliers* univariados/multivariados descritas no Capítulo 2. Entretanto, há alguns inconvenientes. Os principais são listados abaixo (WEBB, 2002):

- Os limiares de decisão nas Equações (2.2) e (2.7) são calculados com base na suposição de que os dados são oriundos de determinadas distribuições paramétricas, como as gaussianas univariadas/multivariadas.
- Mesmo pequenas diferenças em relação à gaussianidade resultam em pobre desempenho na detecção. Isso é particularmente verdadeiro se os dados $\mathbf{x}$ foram assimetricamente distribuídos em torno de sua média $\bar{\mathbf{x}}$.
- A necessidade de um bom-condicionamento da matriz $\mathbf{C_x}$ na Equação (2.7), especialmente se a dimensão de $\mathbf{x}$ for elevada, uma vez que torna-se um problema

complexo obter sua inversa, $\mathbf{C}_{\mathbf{x}}^{-1}$.

- A presença de *outliers* distorce as estimativas $\bar{\mathbf{x}}$ e $\mathbf{C}_{\mathbf{x}}$, aumentando o número de erros de classificação. Há métodos robustos para calcular estimativas da média e da matriz de covariância (ROUSSEEUW; LEROY, 1996), mas a questão envolvendo a inversa de $\mathbf{C}_{\mathbf{x}}$ e seu elevado custo computacional ainda permanece.

Em uma tentativa de atenuar alguns destes problemas propõe-se nesta dissertação o uso conjunto de métodos de detecção univariada e multivariada de *outliers* similares àqueles discutidos acima, constituindo-se uma abordagem original em detecção de novidades, denominada **intervalos de decisão**.

Intervalos de decisão (ID) globais e locais são construídos a partir da distribuição dos erros da quantização referentes aos vetores de treinamento e suas componentes, respectivamente. Um ID global é usado para avaliar a condição total do sistema em estudo. Se o comportamento anormal for detectado, ID locais são usados componente a componente de modo a encontrar aquelas possivelmente anormais. Os testes são executados utilizando limiares de decisão calculados via percentis sobre os perfis de normalidade global e local. Dessa forma, haverá um intervalo de decisão global e intervalos de decisão locais sempre que existir um conjunto de erros de quantização.

As principais vantagens da abordagem proposta são as seguintes:

- A metodologia é inteiramente **não-paramétrica,** isto é, não se faz nenhuma suposição probabilística *a priori* para a distribuição dos dados para o cálculo dos limiares de decisão.
- Nenhum cálculo explícito de um vetor média amostral nem da matriz de covariância amostral (e sua inversa) são necessários. Regularidades estatísticas locais dentro do conjunto de dados são extraídas automaticamente por meio do aprendizado das redes competitivas e são codificadas em seus vetores do peso (protótipos).
- A exatidão da classificação é aumentada por meio de **um procedimento de teste-duplo** que envolve a aplicação seqüencial de um teste multivariado (ID Global) e univariado (ID Local) para a detecção de *outliers*.

A seguir são descritas estratégias usuais de cálculo de limiares de decisão (simples e duplos) e é descrito o método proposto.

## 4.2 Decisões Baseadas em Intervalos Calculados via Percentis

Como explicado anteriormente, os testes de hipóteses podem ser unilaterais ou bilaterais, e usam respectivamente um ou dois limiares (valores críticos) para definir a região crítica (região de rejeição de $H_0$). Portanto, os perfis de normalidade dos testes unilaterais compreendem a região da distribuição empírica que está à esquerda do limiar. Nos testes bilaterais o perfil de normalidade está compreendido entre os limiares. Os valores críticos são calculados por meio de percentis.

### 4.2.1 Métodos de Limiar Simples

Nestes métodos, calcula-se um limiar máximo $\rho^+$ a partir do qual todos os outros valores são considerados anormais ou desconhecidos:

$$\begin{aligned} \textbf{SE} \quad & \rho^{novo} < \rho^+ \\ \textbf{ENTÃO} \quad & \mathbf{x}^{novo} \text{ é } \textbf{NORMAL} \\ \textbf{SENÃO} \quad & \mathbf{x}^{novo} \text{ é } \textbf{ANORMAL} \end{aligned} \tag{4.1}$$

Estes métodos têm sido bastante utilizados em aplicações práticas de detecção de novidades. Por exemplo, Höglund et al. (2000) treinam uma rede SOM com dados representativos da atividade normal de usuários de uma rede de computadores. O limiar $\rho^+$ é determinado pelo cálculo do valor-$p$ associado à distribuição dos erros de quantização referentes a cada vetor do conjunto de treinamento. Esse procedimento para detecção de novidades pode ser implementado da seguinte forma:

- **Passo 1** – Após a conclusão do treinamento, os erros de quantização associados aos $N$ vetores de treinamento são calculados, gerando o conjunto $\left\{e_q^\mu\right\}_{\mu=1}^N$. Nesta dissertação, este conjunto é chamado **perfil de normalidade**.
- **Passo 2** – O erro de quantização associado a um novo vetor de entrada é calculado, $e_q(\mathbf{x}^{novo}) = e_q^{novo}$.
- **Passo 3** – Definir $H_0$ como: "o vetor $\mathbf{x}^{novo}$ é normal". Fazer $\rho^+ = 100(1-\alpha)$ percentil de $\left\{e_q^\mu\right\}_{\mu=1}^N$, em que $0 < \alpha \leq 1$ é o nível de significância estatística do teste. Um nível de significância $\alpha = 0{,}05$ é usual.
- **Passo 4** – Fazer $\rho^{novo} = e_q^{novo}$.
- **Passo 5** – Se $\rho^{novo} < \rho^+$, então $H_0$ é aceita; caso contrário ela é rejeitada.

- **Passo 6** – Repetir Passos 2-5 para cada novo vetor de entrada.

De acordo com os autores esse método é bastante confiável e apresenta taxas aceitáveis de falsos negativos e falsos positivos, eles concluem que esses erros são causados por variações normais nos perfis de usuários da rede de computador em estudo. Abordagens similares são aplicadas para detecção de novidades em redes de comunicação móvel (LAIHO et al., 2005, 2002), modelagem de séries temporais (GONZALEZ; DASGUPTA, 2002) e monitoramento de máquinas (HARRIS, 1993).

Um outro método de limiar simples baseado na rede SOM é proposto em Tanaka et al. (1995) para detecção de falhas em máquinas elétricas. Este método segue os mesmos passos descritos previamente, exceto que, neste caso, o limiar de novidade é calculado com base na distância do vencedor $i^*$ até seus vizinhos mais próximos $D_{i^*j} = \|\mathbf{w}_{i^*} - \mathbf{w}_j\|$. O limiar de decisão é escolhido como o valor máximo dentre estas distâncias:

$$\rho^+ = \max_{\forall j \in V_1} \{D_{i^*j}\} \tag{4.2}$$

na qual $V_1$ é o conjunto de neurônios na vizinhança imediata do vencedor, ou seja $|i^* - j| = 1$. Assim, se $e_q^{novo} = \rho^{novo} > \rho^+$, então o vetor de entrada carrega informação nova ou anômala, i.e., a hipótese nula deve ser rejeitada.

### 4.2.2 Métodos de Limiar Duplo

Explica-se no Capítulo 2 que os testes de limiar duplo são mais robustos a *outliers* indesejados, uma vez que tais testes encontram *outliers* não somente na região de elevado valor da medida de novidade escolhida, mas também na região na qual esses valores são muito pequenos. No referido capítulo descreveram-se também os métodos de limiar duplo mais comuns.

Nestes métodos, calcula-se um intervalo delimitado pelos dois limiares $[\rho^-,\ \rho^+]$ que é, então, usado para julgar um novo vetor de entrada como normal/anormal por meio de um teste de hipóteses:

$$\begin{aligned} \textbf{SE} \quad & \rho^{novo} \in [\rho^-,\ \rho^+] \\ \textbf{ENTÃO} \quad & \mathbf{x}^{novo} \text{ é } \textbf{NORMAL} \\ \textbf{SENÃO} \quad & \mathbf{x}^{novo} \text{ é } \textbf{ANORMAL} \end{aligned} \tag{4.3}$$

Há diferentes maneiras de se calcular intervalos desta natureza, por exemplo, usando a técnica dos *boxplots*. A desvantagem do uso dos *boxplots* para o cálculo dos limiares de decisão é que eles não têm um valor explícito de confiança associado à região de

normalidade escolhida, ou seja, a região entre os limiares $\rho^-$ e $\rho^+$.

Os intervalos de confiança, brevemente discutidos na Seção 2.4.3, estimam não somente uma região onde é mais provável se encontrar o valor verdadeiro da estatística de teste, mas também atribuem uma medida de confiança para esta estimação, o que é uma vantagem, servindo de motivação para a proposição de um novo método de limiar duplo para a detecção de novidades inspirado nos intervalos de confiança. O novo método é descrito na Seção seguinte.

### 4.2.3 Intervalo Decisão Global

Uma amostra dos erros de quantização (denominada **perfil de normalidade**) é tomada ao final do treinamento de uma rede competitiva, utilizando apenas dados considerados normais, como no **Passo 1** dos métodos de limiar simples descritos na Seção 4.2.1. Os limiares superior e inferior são calculados via percentis a partir do perfil de normalidade.

A motivação para a proposição desse método é que, ao avaliar métodos de detecção de novidades baseados em redes competitivas, observa-se que em conjuntos de dados contaminados por *outliers* não se pode assegurar que pequenos erros de quantização garantem que um dado vetor é conhecido (normal). Uma vez que se percebeu uma boa quantidade de erros do tipo falso negativo. Testes bilaterais excluem regiões nas quais o erro de quantização é, de forma incomum, bastante pequeno. Procura-se, então, um método de limiar duplo não-paramétrico que detectasse *outliers* indesejados na região de baixo erro de quantização, quando se utilizasse redes competitivas (ou outro algoritmo de formação de agrupamentos).

A técnica inspira-se no conceito estatístico clássico dos intervalos de confiança, explicados na Seção 2.4.3, uma vez que busca-se obter uma confiança de $100(1-\alpha)\%$ de encontrar os valores de erro de quantização no intervalo $[\rho^-,\ \rho^+]$. A idéia do método é, então, interpretar $\rho^-$ e $\rho^+$ como os limiares $L_1$ e $L_2$ da Equação (2.4), na qual o parâmetro populacional em estudo é o erro de quantização.

Assim, para um dado nível de significância $\alpha$, busca-se um intervalo no qual se possa certamente encontrar uma percentagem $100(1-\alpha)\%$ (por exemplo, $\alpha = 0,05$) de valores normais para o erro de quantização. Portanto, calculam-se os limites superior e inferior da seguinte forma:

- **Limite Inferior** ($\rho^-$): é dado pelo $[100\frac{\alpha}{2}]$ percentil de $\left\{e_q^\mu\right\}_{\mu=1}^N$.
- **Limite Superior** ($\rho^+$): é dado pelo $[100(1-\frac{\alpha}{2})]$ percentil de $\left\{e_q^\mu\right\}_{\mu=1}^N$.

Desta forma, compõe-se o intervalo $[\rho^-,\ \rho^+]$ e aplica-se o mesmo teste mostrado na Equação (4.3). Para se ter uma idéia da distribuição dos erros da quantização produzidos pelos algoritmos neurais, um perfil de normalidade típico para a rede SOM é mostrado na Figura 4.1. As linhas verticais correspondem aos limites do intervalo de decisão ($\alpha = 0{,}05$).

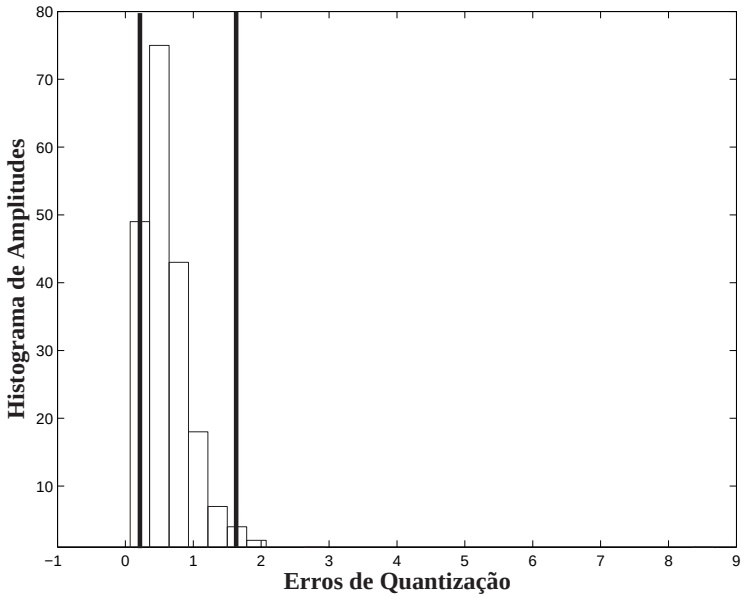


**Figura 4.1:** Perfil de normalidade típico para a rede SOM. Linhas verticais representam o intervalo de decisão global.

Por avaliar o erro de quantização para decidir sobre a novidade de um vetor de entrada, esse método pode ser chamado de **Intervalo de Decisão Global**, pois o EQ é uma medida que concentra a influência global das componentes do vetor de entrada.

### 4.2.4 Intervalo de Decisão Local

Uma vez que uma novidade é detectada pelo ID global, pode ser interessante investigar qual das componentes do vetor de entrada problemático causa uma anomalia no erro de quantização.

Com este propósito, avalia-se o conjunto dos valores absolutos de cada componente do vetor erro de quantização, calculada para todos os vetores de treinamento

$$ABS\left(\mathbf{e}_q(\mu)\right) = \begin{pmatrix} |e_q^1(\mu)| \\ |e_q^2(\mu)| \\ \vdots \\ |e_q^n(\mu)| \end{pmatrix} = \begin{pmatrix} |x_1(\mu) - w_{i^*1}(\mu)| \\ |x_2(\mu) - w_{i^*2}(\mu)| \\ \vdots \\ |x_n(\mu) - w_{i^*n}(\mu)| \end{pmatrix}. \tag{4.4}$$

Desta forma tem-se um perfil de normalidade para cada componente de entrada. O ID local é calculado da mesma forma que o ID global, mas agora tomam-se $N$ observações dos valores absolutos de cada componente do vetor erro de quantização $\mathbf{e}_q$. Assim, para todo o conjunto de valores, $\{\,|e_q^j(\mu)|\,\}_{\mu=1,\ldots,N}^{j=1,\ldots,n}$, calcula-se o intervalo de decisão correspondente $[\,\rho_j^-, \rho_j^+\,]$, no qual $\rho_j^-$ e $\rho_j^+$ são os limites inferior e superior do $j$-ésimo intervalo de decisão local, calculados da seguinte forma:

- **Limite Inferior** ($\rho_j^-$): é dado pelo $[100\frac{\alpha}{2}]$ percentil de $\{\,|e_q^j(\mu)|\,\}_{\mu=1,\ldots,N}^{j=1,\ldots,n}$.
- **Limite Superior** ($\rho_j^+$): é dado pelo $[100(1-\frac{\alpha}{2})]$ percentil de $\{\,|e_q^j(\mu)|\,\}_{\mu=1,\ldots,N}^{j=1,\ldots,n}$.

Portanto, sempre que um novo vetor de entrada $\mathbf{x}^{novo}$ é sinalizado como faltoso pelo detector baseado em intervalos de decisão para erro de quantização, tomam-se os valores absolutos de cada componente $e_q^j(novo)$ do vetor erro de quantização $\mathbf{e_q}^{novo}$ correspondente a $\mathbf{x}^{novo}$ e executa-se o seguinte teste

| | |
|---|---|
| **SE** | $\|e_q^j(novo)\| \in [\,\rho_j^-, \rho_j^+\,]$, |
| **ENTÃO** | $x_j$ é **NORMAL.** |
| **SENÃO** | $x_j$ é uma possível causa do alarme. |

Assim, se a componente $x_j$ está dentro da faixa definida pelo intervalo $[\,\rho_j^-, \rho_j^+\,]$, então ela não contribuiu para a detecção previamente realizada (ID global), caso contrário ela é uma possível causadora da detecção. Se não se encontrar nenhuma componente fora do seu respectivo intervalo, então concluiu-se que a detecção previamente realizada é uma **falsa detecção** (alarme falso) que deverá ser corrigido. Não há necessidade de se utilizar o mesmo nível de significância estatística do ID global.

## 4.3 Uma Aplicação em Sistemas Celulares 3G

As técnicas mostradas neste capítulo são usadas para tratar o problema da detecção de anomalias em sistemas celulares. Para tanto, são utilizados dados gerados por um simulador computacional do sistema celular CDMA2000. Em seguida são mostrados sumariamente as etapas cumpridas para a tarefa de detecção de anomalias. O algoritmo geral utilizado é o seguinte:

1. Selecionar variáveis que descrevam o sistema de interesse. Uma observação destas variáveis forma um vetor de dados $\mathbf{x}(\mu)$, representativo do estado do sistema no instante de tempo $\mu$;

2. Armazenar $N$ vetores de dados, correspondentes a $N$ instâncias (ou casos), que descrevem apenas o funcionamento **normal** ou **esperado** do sistema (**classificação binária**);

3. Treinar a rede neural com os $N$ vetores obtidos;

4. Após o término do treinamento, determinar a medida de novidade (estatística de teste) $z$ a ser extraída da rede neural treinada, por exemplo o erro de quantização para as redes competitivas;

5. Calcular um valor amostral de $z$ para cada vetor de treinamento $\mathbf{x}(\mu)$, para obter uma amostra da distribuição empírica da estatística de teste $(z_1, z_2, \ldots, z_N)$. A partir dessa amostra, calcular um **perfil de normalidade** para o sistema;

6. Para classificar um novo vetor de dados $\mathbf{x}^{novo}$ como **normal** ou **anormal**, calcula-se a medida de novidade $z^{novo}$ associada a este vetor.

7. Teste de Hipóteses: critério de decisão que permite classificar $z^{novo}$ com base nos valores críticos (limiares) da medida de novidade que delimitam o perfil de normalidade.

Nos testes, a **Hipótese Nula** ($H_0$) é definida da seguinte forma:

- $H_0$: O vetor de entrada $\mathbf{x}^{novo}$ reflete atividade **conhecida**.

na qual o termo **conhecido** indica que o vetor $\mathbf{x}^{novo}$ representa comportamento normal, se estamos lidando com problemas de classificação binária. Se temos um problema de classificação multi-classes, o termo **conhecido** sugere que o vetor de entrada pertence a uma das classes conhecidas *a priori*.

A chamada **hipótese alternativa**, denotada por $H_1$, é obviamente, dada por:

- $H_1$: O vetor de entrada reflete atividade **desconhecida**.

de forma que, nesse caso, o vetor de entrada carrega informação nova, o que, pelo modo como é definido neste trabalho, é indicativo de comportamento anormal do sistema em estudo.

A Figura 4.2 mostra a maneira como o teste duplo (ID Global + ID Local) é aplicado na detecção de falhas em um sistema celular 3G. Pode-se perceber a principal inovação do método de detecção de novidades proposto nesta disseetação, o uso de duas fases de teste. Outros trabalhos na área usam apenas abordagens similares ao teste ID global (MARKOU; SINGH, 2003a, 2003b). A principal vantagem do uso de dois testes é conseguir detectar falsos alarmes.

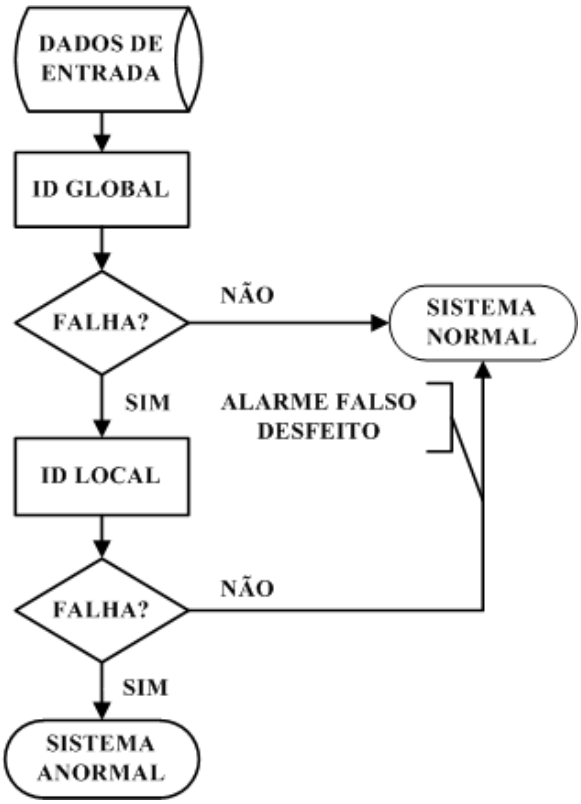


**Figura 4.2:** Fluxograma de teste duplo para a detecção de anomalias.

### 4.3.1 Dados Experimentais

Para a detecção de anomalias é necessário, inicialmente, a definição dos elementos da rede celular a serem monitorados, ou quais fases de operação estão em estudo. A partir dessa decisão é que podem ser definidas as variáveis que vão compor os vetores de treinamento. Cada variável é um dos indicadores-chave de desempenho (*Key Performance Indicator – KPI*) que descrevem o comportamento operacional do sistema num dado instante[1].

$$\mathbf{x}(\mu) = [x_1(\mu) \;\; x_2(\mu) \;\; \cdots \;\; x_n(\mu)]^T \tag{4.5}$$

$$= [KPI_1(\mu) \;\; KPI_2(\mu) \;\; \cdots \;\; KPI_n(\mu)]^T \tag{4.6}$$

A Equação (4.5) mostra como são formados os vetores de treinamento para as redes neurais, composto por $n$ KPI's descritivas do estado da rede celular. Nos testes realizados a seguir os vetores de dados têm quatro atributos, correspondendo às seguintes KPI's:

[1]KPI's são um conjunto de medidas que sumariza o comportamento da rede celular de interesse e pode ser usado para a regulação e especificação do sistema celular.

- Número de usuários em serviço;
- Vazão de dados total no enlace direto ($kB/s$);
- Razão da interferência total pelo ruído térmico (*Noise Rise*) ($dBm$);
- Interferência devido a outras células ($dBm$).

### 4.3.2 Configuração das Simulações

Os dados referentes ao funcionamento de uma rede celular são gerados por um simulador estático do sistema CDMA2000[2]. O ambiente de rede celular de terceira geração[3] (3G) usado para simulações do sistema é macrocelular, com dois anéis de células interferentes em torno de uma célula central, num total de 19 células. Outras configurações são possíveis, com 1, 7 ou 37 células. Todas as estações rádio-base (ERB) usam antenas onidirecionais situadas a uma altura de 30 metros do nível do solo. O modelo de propagação de rádio-freqüência (RF) é o Okumura-Hata (HATA, 1980; OKUMURA et al., 1968) clássico com freqüência de portadora igual a $900MHz$. Fenômenos como desvanecimento multipercurso e sombreamento são incluídos no modelo do canal de comunicação. A cada usuário móvel são atribuídos serviços de voz ou de dados com diferentes taxas de transmissão. O controle de potência é feito pela avaliação dos níveis de potência da portadora e carga total da célula. Parâmetros de qualidade, tais como $E_b/N_t^{alvo}$ e o nível máximo de ruído (*Noise Rise*) são ajustados em $5dB$ e $6dB$, respectivamente. O número típico de usuários móveis iniciais é 60, mas estes podem ser removidos do sistema mediante a ação do algoritmo de controle de potência.

Devido ao elevado número de parâmetros de entrada que podem ser ajustados dentro do simulador (da ordem de 30[4]), alguns cenários de simulação específicos são selecionados, focalizando na análise do comportamento de tráfego e interferência da rede CDMA2000. Para cada iteração da simulação, um conjunto de de KPIs é armazenado e usado para o treinamento das redes neurais e para testes dos algoritmos. Cada conjunto de dados correspondente a um determinado cenário da rede é formado por 500 vetores de estado, gerados por 500 iterações independentes da ferramenta de simulação estática. Destes vetores, aproximadamente 80% são aleatoriamente selecionados para o treinamento e os 20% restantes são usados para testar a rede neural.

Para um dado valor de um parâmetro de treinamento (por exemplo o número dos neurônios), a rede neural é treinado 100 vezes com pesos iniciais diferentes. Em cada

[2]Maiores detalhes do simulador estão no Apêndice B.

[3]Sistemas celulares 3G foram concebidos para fornecer mobilidade global e uma larga faixa de serviços incluindo telefonia, mensagens de texto, Internet e transmissão de dados.

[4]Estes parâmetros, bem como os valores utilizdos, estão discriminados na Seção B.1.

**Tabela 4.1:** Taxas médias (%) de alarmes falsos (AF) e limiares (ID Global e valor-$p$).

| | **ID Global** | | **Valor-$p$** | |
|---|---|---|---|---|
| **Rede Neural** | $[\rho^-\ \rho^+]$ | AF (%) | $\rho^+$ | AF (%) |
| **WTA** | [0,37 1,53] | 12,4 | 0,47 | 17,9 |
| **FSCL** | [0,05 2,27] | 6,8 | 1,80 | 11,1 |
| **NGA** | [0,07 3,19] | 6,2 | 2,13 | 7,3 |
| **SOM** | [0,03 3,52] | 4,9 | 2,53 | 5,6 |

iteração de treinamento, os vetores de estado são selecionados aleatoriamente para compor os conjuntos de treinamento e teste. A ordem de apresentação dos vetores do estado para cada época do treinamento também é mudada aleatoriamente. O valor final da taxa média de alarmes falsos é calculado para 100 iterações independentes de testes. Treinamento e testes independentes são necessários para evitar estimativas tendenciosas para a taxa de erro.

### 4.3.3 Resultados

Esta Seção apresenta os resultados dos testes de avaliação das redes neurais na tarefa de detecção de falhas usando o método proposto neste Capítulo. Cada conjunto de simulações tem por objetivo observar a influência de um determinado parâmetro de treinamento, como o número de neurônios, no desempenho da rede competitiva.

Por simplicidade, para cada conjunto de simulações avaliando a variação de um dado parâmetro, serão mostrados os resultados obtidos por apenas uma rede competitiva. Desempenhos similares são observados para as demais.

O primeiro conjunto de simulações avalia o desempenho das redes neurais competitivas descritas no Capítulo 3, quanto à ocorrência de alarmes falsos. Para esta finalidade, o cenário da rede celular escolhido corresponde a 100 estações móveis tentando inicialmente conectar a 7 ERBs, para os quais não se considera efeitos de desvanecimento. Somente os serviços da voz são permitidos. Os resultados (em porcentagem) estão organizados na Tabela 4.1, na qual são também mostrados os intervalos de decisão encontrados para o nível de confiança avaliado (95%, ou seja, $\alpha = 0{,}05$).

Pode-se observar na Tabela 4.1, que a principal diferença entre as redes está no limiar superior $\rho^+$, quanto melhor o desempenho da rede, maior o valor deste limiar. A explicação está no fato de que as redes que desempenham uma melhor quantização vetorial dos dados, o que resulta numa distribuição de erros de quantização mais próximos de zero. Deste modo, o limiar superior a $100(1-\frac{\alpha}{2})\%$ dos erros de quantização só é encontrado entre os maiores valores de EQ.

As taxas médias de erro são calculadas para 100 iterações independentes de treinamento/teste das redes neurais. Para todas elas, o número de neurônios e o número de épocas de treinamento são ajustados em 20 e 50, respectivamente, e os valores iniciais e finais para a taxa de aprendizagem das redes competitivas são ajustados a $\eta_0 = 0,9$ e a $\eta_T = 10^{-5}$. Para a FSCL somente, o parâmetro $q$ na Equação (3.6) é ajustado em 5. Para a NGA, somente, ajustam-se $\lambda_0 = 5$ e $\lambda_T = 0,0001$.

As redes de SOM/NGA apresentam melhor desempenho que as redes WTA/FSCL. A diferença principal entre eles é que as primeiras possuem a propriedade de preservação de topologia aliada às habilidades de formação agrupamentos de dados (*clustering*), que são inerentes às regras do aprendizado competitivo simples. Esta propriedade é incorporada nas regras de aprendizagem das redes SOM/NGA pela inserção do fator de vizinhança, alterando os pesos dos neurônios da rede conforme suas distâncias ao neurônio vencedor. Agindo como conexões laterais, o fator de vizinhança transforma a regra de aprendizagem competitiva simples, Equação (3.3), em um tipo de regra de aprendizagem de Hebbian como nas Equações (3.8) e (3.14). Sabe-se bem que as regras de aprendizagem hebbianas podem aprender estatísticas de segunda ordem a partir da distribuição de dados de entrada, enquanto que as regras de aprendizagem competitivo simples aprendem somente estatísticas de primeira ordem (KOHONEN, 2001).

O segundo conjunto de simulações avalia a sensibilidade das redes neurais a mudanças em seus parâmetros de treinamento. O objetivo é compreender como o número dos neurônios, o número de épocas de treinamento e o tamanho do conjunto do treinamento afetam o desempenho das redes neurais. Os resultados são mostrados nas Figuras 4.3, 4.5 e 4.7, respectivamente.

Em cada caso, compara-se a abordagem de intervalos de decisão com a abordagem de limiar simples, como aquela utilizada por Laiho et al. (2002). O cenário da rede celular escolhido corresponde a 120 estações móveis está tentando inicialmente conectar a 7 ERBs, desta vez considerando efeitos de desvanecimento e sombreamento. Serviços de voz e de dados são permitidos.

Nas Figuras 4.3 e 4.4, o número dos neurônios é variado de 1 a 100, e cada iteração de treinamento dura 50 épocas. O comportamento pouco usual apresentado na Figura 4.3 é explicado pela Figura 4.4. Uma vez que o aumento do número de neurônios causa uma diminuição global dos EQ, os limites do ID Global se aproximam, aumentando a região de exclusão. Desta forma há um número maior de alarmes e, conseqüentemente, mais alarmes falsos são reportados.

Nas figuras 4.5 e 4.6, o número de épocas varia de 1 a 100, enquanto o número de neurônios é fixado em 20.

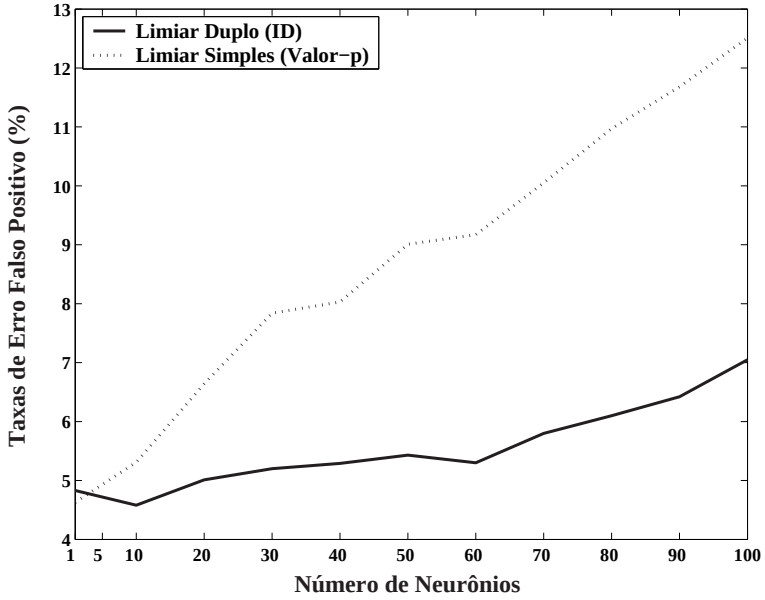


**Figura 4.3:** Evolução da taxa de alarmes falsos com o número de neurônios da rede FSCL.

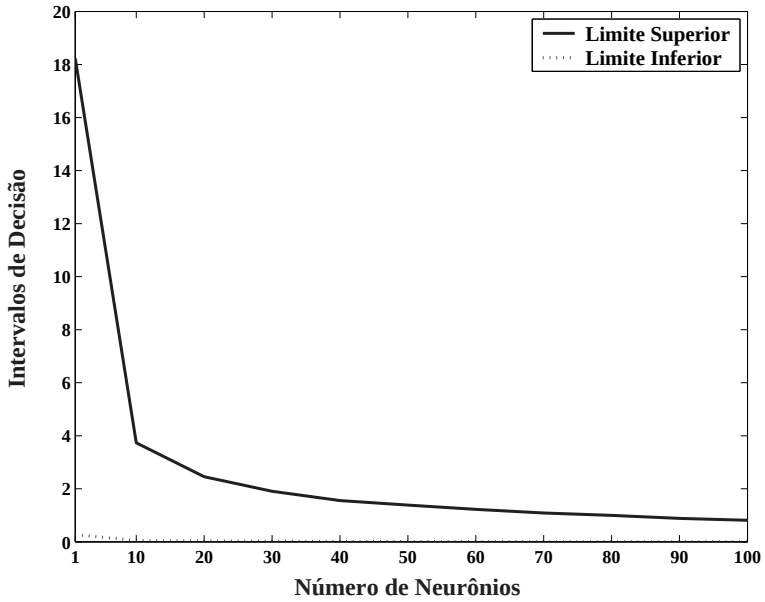


**Figura 4.4:** Limites inferior e superior do intervalo de decisão para a rede FSCL.

Nas figuras 4.7 e 4.8, o número dos neurônios e o número de épocas de treinamento são fixados em 20 e 50, respectivamente, quanto ao tamanho do conjunto do treinamento (isto é número dos vetores usados para o treinamento) é variado de 10% a 90% do total de vetores disponíveis.

Em uma simulação de anomalias reais, inserindo-se ruído com intensidade (desvio-padrão) variável nos vetores de teste normais, pode-se calcular as taxas de ausência de alarme (falso negativo). A Figura 4.9 mostra os resultados obtidos.

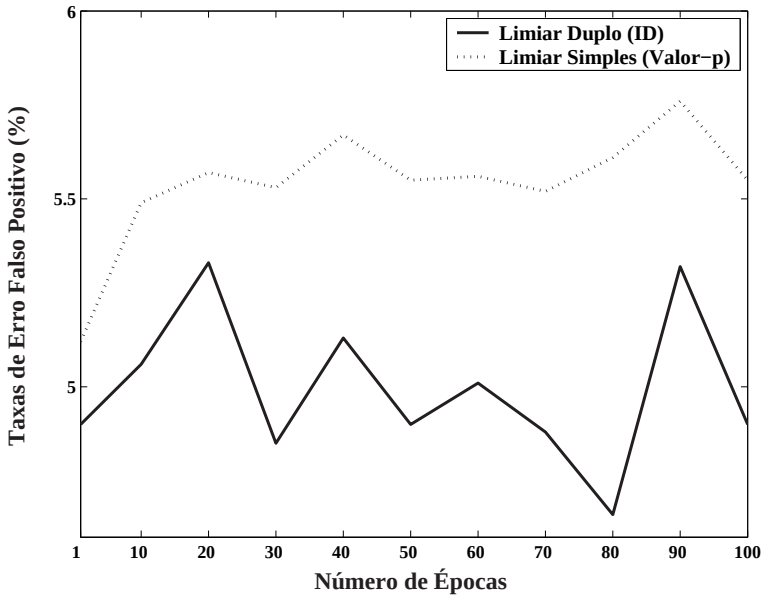


**Figura 4.5:** Evolução da taxa de alarmes falsos com o número de épocas de treinamento para a rede SOM.

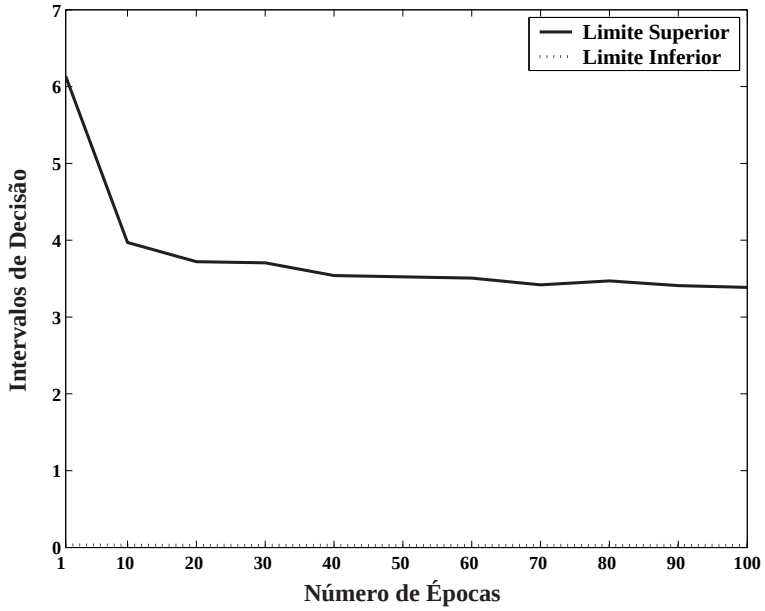


**Figura 4.6:** Limites inferior e superior do intervalo de decisão para a rede SOM.

Para uma rede SOM treinada, avalia-se o desempenho da detecção via *ID Global* a detecção via valor-$p$ e via combinação das detecções ID Global e ID Local sobre um mesmo conjunto de dados (mesmo perfil de normalidade). Para o ID global utiliza-se $\alpha = 0{,}05$ e para o ID local utiliza-se $\alpha = 0{,}01$. As taxas de ausências de alarmes das três abordagens em função do desvio-padrão do ruído usado para contaminar dados normais, tornando-os *outliers*, estão mostradas na Figura 4.9. Para cada valor de desvio-padrão repetiu-se os procedimentos de treinamento/teste 100 vezes independentes, tomando-se após isso os

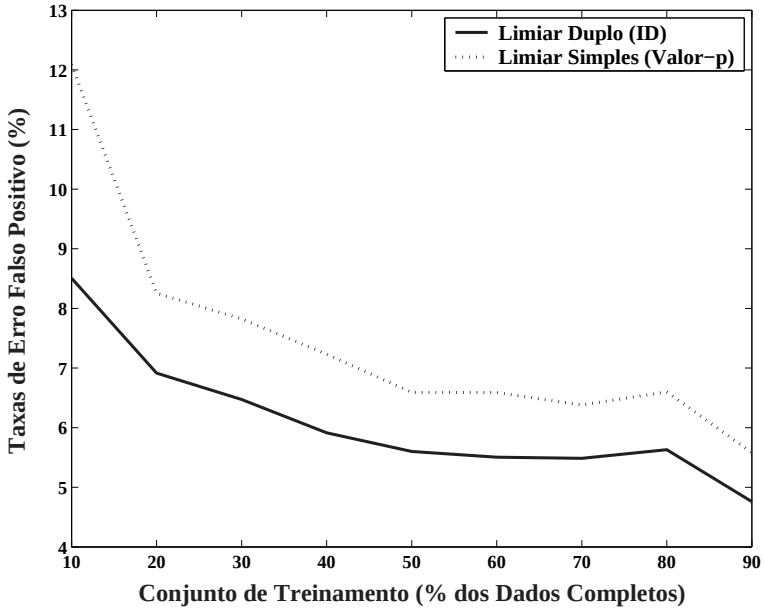


**Figura 4.7:** Evolução da taxa de alarmes falsos com o número de épocas de treinamento para a rede NGA.

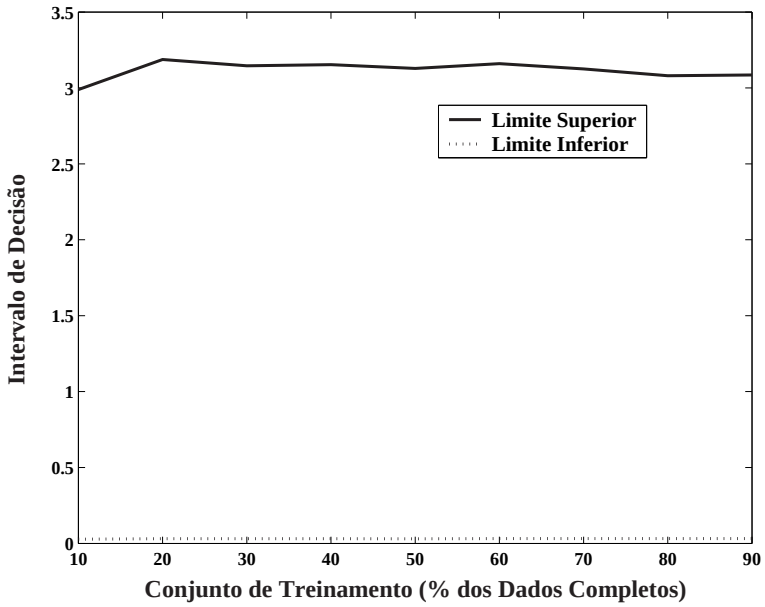


**Figura 4.8:** Limites inferior e superior do intervalo de decisão para a rede NGA.

valores médios das taxas de erro obtidas.

A aplicação conjunta dos testes ID Global e ID Local, conforme o fluxograma da Figura 4.2, apresenta a menor taxa de ausência de alarmes. Isso indica que o uso do teste duplo (ID Global + Local) mostra-se mais preciso e robusto a ruído que o método de limiar simples valor-$p$. Pode-se observar na Figura 4.9 que apenas se a magnitude (desvio-padrão) do ruído for suficientemente grande (ou seja, os padrões anormais são

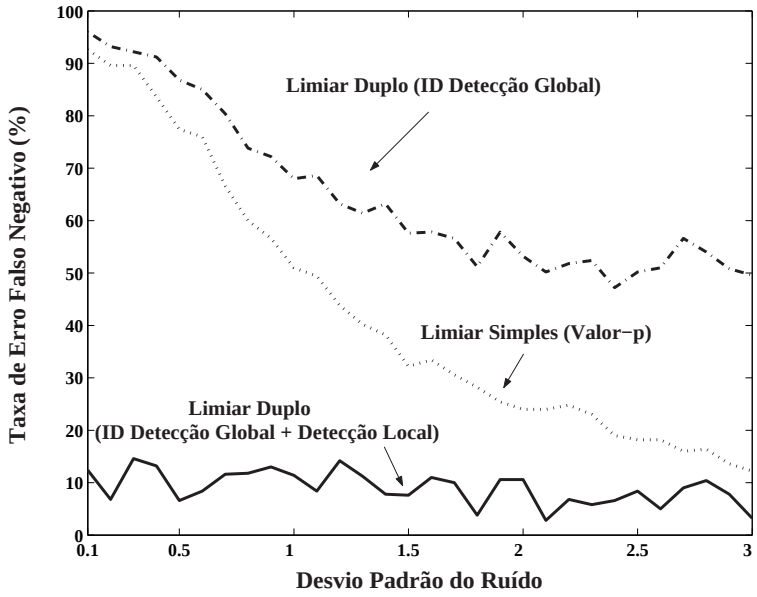


**Figura 4.9:** Limites inferior e superior do intervalo de decisão para a rede NGA.

facilmente distinguíveis) é que o método de limiar simples (valor-$p$) apresenta taxas de erro similares àquelas do método duplo (ID Global + Local).

## 4.4 Discussão

Os resultados obtidos pela abordagem proposta neste capítulo dão margem para a discussão das questões seguintes.

**Condições para se realizar comparações efetivas:** sempre que aplicações de RNAs à análise de sistemas complexos devem ser relatadas, os autores devem comparar algoritmos diferentes a fim ter uma visão mais precisa acerca da diferença entre seus desempenhos. Isto é necessário porque a escolha da rede neural que melhor se ajusta a uma situação real depende de diversos fatores.

Entretanto, não se tem ainda um método geral o suficiente para comparação entre os diferentes métodos de detecção de novidades baseados em RNAs, principalmente quando se quer comparar os métodos supervisionados e os não-supervisionados.

Uma dificuldade para se encontrar tal método geral está na diversidade de modelos neurais e, por conseguinte, na diversidade de parâmetros inerentes a cada rede que podem ser usadas como medida de novidade. Alguns métodos de cálculo de limiares são de aplicação mais clara em determinados tipos de rede que em outras.

O método proposto neste Capítulo tem aplicação apenas em redes competitivas, já que utiliza uma medida particular a essas redes, o erro de quantização. Entretanto, é possível

conceber certos ajustes que tornem este método aplicável às redes supervisionadas. É possível calcular intervalos de decisão de uma medida escalar extraída de uma rede MLP, por exemplo, a própria saída da rede (rede com 1 neurônio de saída) ou a norma do vetor de saída (rede com mais de 1 neurônio de saída).

Outro ponto que merece atenção é a estratégia de classificação. A literatura especializada não especifica como se deve proceder para comparar numa mesma massa de dados um classificador binário e um classificador multicategórico. Uma solução seria encontrar, ou criar, uma massa de dados que contivesse em abundância exemplos da classe normal (exemplos positivos) e da classe anormal (exemplos negativos) e, a partir desse conjunto de dados, testar o desempenho dos classificadores que usam dados de apenas uma classe e classificadores que usam dados das duas classes.

**Controle dos tipos de erro:** como indicado na Seção 2.3, deve-se avaliar as habilidades das redes neurais com respeito a dois tipos de erros: alarme falso (erro Tipo I) e ausência de alarme (erro Tipo II). Para sistemas celulares, erros Tipo II têm custos operacionais elevados porque podem resultar na perda de equipamentos caros. Este tipo de erro é particularmente difícil de tratar porque uma anomalia pode não causar nenhum dano no início de sua ocorrência, mas, com o passar do tempo, torna-se mais séria ao ponto que a rede neural possa detectá-lo. Porém, quando a anomalia é detectada pode ser demasiado tarde. Assim, um modelo neural com alta sensibilidade teria melhor desempenho nestas situações. Entretanto, pequenas variações no comportamento esperado do sistema não são sempre indicativas de problemas reais. Podem ser causados, por exemplo, por ruído aleatório inserido nas medidas ou por flutuações transientes da tensão da ERB devido a fatores externos (por exemplo, relâmpagos durante uma tempestade). Um modelo de alta sensibilidade relataria alarmes falsos muito freqüentemente, até o ponto em que os operadores não teriam mais confiança sobre as decisões sistema, podendo eventualmente se recusar a acreditar que um problema real está ocorrendo.

Um sistema **ideal** de detecção de anomalias seria aquele que indicasse anomalias somente quando um problema real está em curso ou em vias de acontecer. Um sistema **possível** de detecção de anomalias sempre relatará um determinado número de alarmes falsos e não responderá em determinadas situações defeituosas. O objetivo final do projetista é manter as probabilidades da ocorrência destes erros em níveis aceitáveis.

Problemas similares são encontrados em outros sistemas de detecção de novidades, que são usualmente avaliados pelo número de falsos alarmes e ausências de alarmes que eles produzem. Um sistema ideal deveria ter $\alpha = 0$ e $\beta = 0$, mas isso não é possível na prática. Assim, resta tentar gerenciar as probabilidades de erro $\alpha$ e $\beta$ baseados nas conseqüências (e.g danos pessoais, altos custos, quebra de máquinas etc.) para o sistema em estudo.

Por exemplo, reportar falsos alarmes muito freqüentemente, ao ponto dos operadores do sistema perderem a confiança nos alarmes e passarem a ignorá-los. Em testes médicos, a ausência de alarme (falso negativo) deixa pacientes e médicos incorretamente seguros de que não existe uma doença que na verdade existe.

A dificuldade é que, para qualquer tamanho fixo de amostra $N$, um decréscimo em $\alpha$ causará um aumento em $\beta$. Inversamente, um aumento em $\alpha$ causa um decréscimo em $\beta$. Para diminuir ambos $\alpha$ e $\beta$ para níveis aceitáveis, deve-se aumentar o tamanho $N$ da amostra ($\beta \propto \frac{1}{N}$). Também, para qualquer $\alpha$ fixo, um aumento em $N$ causará uma redução em $\beta$, i.e., um número maior de observações reduz a probabilidade de não rejeitar a hipótese nula quando ela é falsa.

Duas soluções seriam aplicáveis: aumento do número de neurônios da rede (arquiteturas crescentes) ou aumento artificial do número de observações. Usualmente, em detectores de novidades baseados em redes neurais, se há um aumento no número de neurônios para causar uma diminuição em $\alpha$ e $\beta$, A desvantagem é que os custos computacionais também poderiam aumentariam rapidamente. Isso pode ser problemático se o sistema de detecção de novidades precisa trabalhar em tempo real (*online*), como em redes de computadores ou aplicativos para a detecção de mensagens indesejadas. Mesmo para aplicações *offline*, o grande esforço computacional requer máquinas de elevado potencial, o que causaria uma elevação nos custos com equipamentos (*hardware*). Isto leva à avaliação da segunda solução, a criação de observações artificiais, que podem ser obtidas de forma paramétrica, via estimação da função densidade de probabilidade, ou de forma não-paramétrica, via, por exemplo, reamostragem a partir da amostra original.

No Capítulo seguinte, serão desenvolvidas novas abordagens de forma a obter resultados e conclusões relevantes com respeito ãs duas questões levantadas acima.

Um outro ponto passível de maiores esclarecimentos diz respeito à sistemática de aplicação do teste duplo de detecção (ID Global + ID Local) proposto neste Capítulo.

A Figura 4.2 mostrou o fluxograma empregado nesta dissertação para a realização do teste duplo. Pode-se, entretanto, conceber um outro sistema similar de detecção de anomalias, mostrado na Figura 4.10, no qual ambos os testes são sempre realizados, e não apenas quando o teste global indica falha.

Essa abordagem, apesar de ter sido considerada, não foi adotada neste trabalho por necessitar de maiores esclarecimentos sobre questões importantes acerca de seu funcionamento. Uma das questões-chave é: o que deve ser feito quando há discordância entre os dois testes? Qual estará correto? Para respondê-la corretamente, será necessário escolher qual dos testes deverá ser levado em consideração. Normalmente esta ecolha recairá sobre o teste global, por ser o mais utilizado na literatura, desta forma será o mesmo que

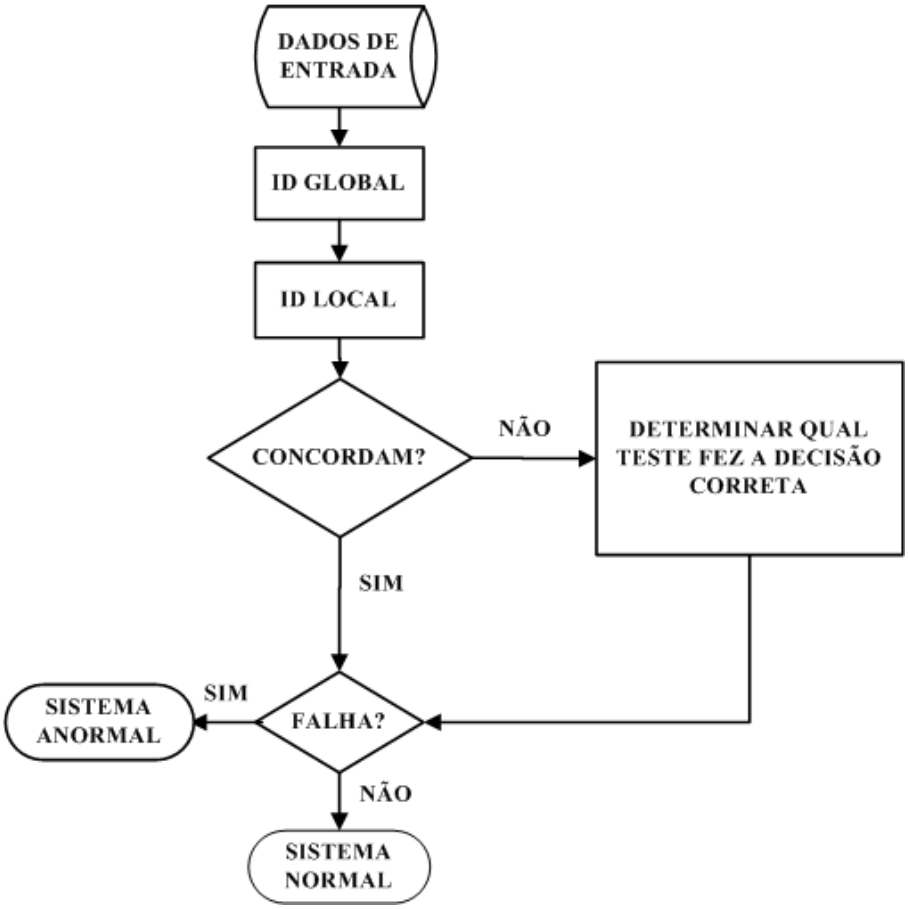


**Figura 4.10:** Uma outra abordagem possível de teste duplo para a detecção de anomalias.

empregar o fluxograma da Figura 4.2.

Outra opção, possivelmente mais acurada e mais custosa, será projetar um método para determinar qual dos testes (global ou local) tem maior probabilidade de estar correto naquele momento de operação. A concepção de tal sistema de decisão está fora do escopo do presente trabalho, mas poderá vir a ser um dos seus mais interessantes desdobramentos.

## 4.5 Conclusão

Neste capítulo propôs-se um teste bilateral, através da seleção de dois limiares (superior e inferior), definindo uma região de variação aceitável para todos os erros de quantização considerados normais. A necessidade de um teste bilateral vem de um problema existente quando se usa um único limiar (superior) na análise de redes celulares, dependendo dos KPIs escolhidos, valores excessivamente baixos de uma determinada variável podem também ser indicativos de uma anomalia no sistema. Em outras aplicações, igualmente, situações deste tipo podem indicar novidades. Assim, é necessário definir também

um limiar inferior para que se possa fazer um exame destes casos. A abordagem proposta foi baseada no conhecido método estatístico dos intervalos de confiança.

O método proposto age em duas fases: uma global e outra local. A detecção global é baseada na medida escalar erro de quantização, a local é baseada nas componente do vetor erro de quantização.

Conforme mostrado na Seção 4.3.3, a utilização do método ID Global, mostrou melhor desempenho quanto às taxas de erro falso positivo que o método de limiar simples valor-$p$. Quanto às taxas de erro falso negativo, a conjunção dos testes ID Global e ID Local apresentou resultados melhores que o teste do valor-$p$, conforme ilustra a Figura 4.9.

Quanto às redes competitivas utilizadas, as redes SOM e NGA apresentaram melhor desempenho, embora as demais, mais simples, não tenham apresentado desempenho muito inferior. Desta forma é possível aplicar-se as redes competitivas WTA e FSCL em certas aplicações nas quais a velocidade e o baixo esforço computacional sejam mais importantes que um elevado desempenho nos testes de novidade.

O capítulo seguinte traz a expansão do método proposto neste capítulo, de aplicação, até então, limitada às redes neurais competitivas, permitindo sua aplicação em um número maior de redes neurais artificiais, incluindo as redes supervisionadas mais conhecidas MLP e RBF. Esta generalização é estendida a outros métodos de detecção de novidades já disponíveis na literatura utilizados em conjunção com redes competitivas. Desta forma, será proposta uma metodologia geral o suficiente para permitir a comparação de diferentes redes neurais e diferentes abordagens de cálculo de limiares para a detecção de novidades.

Para tratar do problema do aumento do tamanho $N$ da amostra melhorando a confiabilidade estatística dos resultados obtidos, será utilizada a técnica de *bootstrap* (EFRON; TIBSHIRANI, 1993) que permite aumentar o número de observações da variável de interesse por meio de reamostragem.

# 5 Detecção Neural de Novidades: uma Abordagem Unificadora

Este capítulo propõe uma metodologia geral para comparação de diferentes métodos para detecção de novidades usando redes neurais artificiais.

Os métodos propostos são testados computacionalmente numa aplicação em engenharia biomédica.

## 5.1 Introdução

Há um considerável número de trabalhos que usam redes neurais artificiais para a detecção de novidades (HODGE; AUSTIN, 2004; MARKOU; SINGH, 2003b), porém poucos têm sido propostos explorando plataformas de comparação, nem fornecem resultados claros indicando, por exemplo, qual técnica funciona melhor para qual tipo de dados, ou qual tem seu desempenho menos degradado pela presença de *outliers* desconhecidos. Em geral, as técnicas atualmente disponíveis variam segundo os pontos abaixo:

- **Estratégia de Classificação:** pode-se utilizar uma estratégia de **classificação binária**, na qual se treina o classificador com exemplos de uma única classe, chamados exemplos **positivos** (SCHOLKOPF et al., 2000), ou uma estratégia de **classificação multivalente** na qual se treina o classificador com dados de todas as classes em questão (VASCONCELOS et al., 1995).

- **Tipo de Rede Neural:** esta escolha tem algum grau de relação com o item acima. Pode-se utilizar métodos que utilizam redes supervisionadas e redes não-supervisionadas. Vale destacar que cada rede fornecerá uma **medida de novidade** particular, que deverá ser escolhida para fazer a decisão.

- **Tipo de Teste:** a partir da modelagem dos dados usando redes neurais, calculam-se limiares para decidir sobre a novidade de um vetor de dados, com base na medida de novidade gerada por ele. Pode-se utilizar testes de **limiar simples** ou de **limiar duplo**.

- **Regra para Cálculo do(s) Limiar(es) de Decisão:** pode-se utilizar regras heurísticas, bem como métodos bem fundamentados estatisticamente.

Esta última questão, envolvendo o cálculo de limiares, é uma das mais delicadas Singh & Markou (2004), uma vez que mesmo havendo concordância com relação aos três primeiros pontos acima, ainda assim pode-se divergir bastante sobre o modo de se calcular os limiares.

A motivação para se propor uma metodologia geral para o cálculo de limiares é a observação de que muitos sistemas neurais de detecção de novidades, especialmente aqueles baseadas nas redes MLP e RBF, têm seus limiares calculados heuristicamente, sem princípios claramente indicados. Um exemplo é a regra *Winner-Take-All* (WTA) Li et al. (2002) para classificadores clássicos baseados nas redes MLP/RBF segundo a qual a saída assume o valor 1 se a ativação do respectivo neurônio for superior a 0,5 e, caso contrário, assume o valor 0.

Por outro lado, a maioria dos métodos de cálculo de limiar utilizados em detectores de novidade baseados em redes competitivas como a rede SOM Barreto et al. (2004), Höglund et al. (2000), Muñoz & Muruzábal (1998) é mais bem fundamentada em termos estatísticos. Esta dissertação sustenta que os métodos de cálculo de limiares de decisão utilizados em redes competitivas podem igualmente ser usadas por detectores de novidades baseados nas redes supervisionadas (como MLP e RBF). Para tanto, propõe-se uma abordagem unificada para o cálculo de tais limiares.

Conforme indicado no Capítulo 4, um outro problema encontrado quanto ao cálculo dos limiares de decisão diz respeito à confiabilidade estatística. Afirmou-se, então, que pode-se diminuir os erros Tipo I e Tipo II aumentando-se o tamanho da amostra de dados, o que nem sempre é realizável (por exemplo, devido a questões físicas ou financeiras). Um outro meio de se resolver esta questão é inserir o uso de alguma técnica de reamostragem na fase de cálculo dos limiares. Desta forma eles são calculados com base num maior número de dados, o que lhes provê de uma maior confiabilidade estatística.

Tendo em vista os pontos acima, os objetivos específicos deste capítulo são:

1. Propor uma metodologia não-paramétrica para o cálculo de limiares de decisão válida para sistemas de detecção de novidades baseados em redes neurais supervisionadas e redes não-supervisionadas;

2. Propor o uso da reamostragem *bootstrap*[1], como parte integrante da metodologia supra-citada, para melhorar a confiabilidade estatística no cálculo dos limiares de

[1]Não há ainda uma tradução técnica formal para a língua portuguesa, trabalhos escritos em português têm usado o termo em inglês com destaque em itálico ou entre aspas. O termo em inglês é mantido neste texto.

decisão;

3. Comparar os diferentes métodos para determinação dos limiares de decisão usando diferentes redes neurais;

4. Usar a nova metodologia para avaliar a sensibilidade dos classificadores à proporção relativa entre dados positivos e negativos.

## 5.2 Reamostragem *Bootstrap*

*Bootstrap* é uma técnica de reamostragem de dados introduzida por Efron (1979). Uma amostra *bootstrap* é obtida aleatoriamente por $M$ reamostragens, com reposição, da amostra original. Neste capítulo, novas amostras de dados são geradas usando *bootstrap* não-paramétrico, tornando desnecessário fazer suposições de que os dados pertencem a uma distribuição paramétrica específica (por exemplo, distribuição normal multivariada). O método de reamostragem *bootstrap* é abordada ainda em (DiCICCIO; EFRON, 1996; EFRON; TIBSHIRANI, 1993).

Em diversas aplicações em que se busca calcular intervalos de confiança, supõe-se freqüentemente que os dados são observações independentes de uma distribuição normal, seja recorrendo ao teorema do limite central (TRIOLA, 1999) ou a alguma outra justificativa teórica, e os intervalos são construídos usando a distribuição $t$ de Student. Essa abordagem pode apresentar problemas se:

- os dados são oriundos de alguma distribuição de dados que não seja a normal,
- não se está certo sobre a distribuição (talvez seja mesmo uma distribuição desconhecida) ou o processo gerador dos dados.

No primeiro caso, pode-se procurar por resultados teóricos sobre a maneira construir intervalos da confiança das distribuições estatísticas não normais. No segundo caso tais resultados não existem (ou raramente existem), então, na prática, geralmente supõe-se a normalidade dos dados e constrói-se intervalos de confiança a partir dessa suposição. Usar a reamostragem de *bootstrap* é uma alternativa atrativa para resolver esse problema.

A técnica de *bootstrap* é usada para obter uma descrição das propriedades amostrais dos estimadores empíricos usando a própria amostra original, em vez de outros resultados teóricos (por exemplo, métodos paramétricos). O único requisito é que a amostra seja representativa da realidade do processo em estudo.

Se, por exemplo, o interesse é estimar um intervalo da confiança para uma estatística de teste $\nu$, começando com uma amostra de tamanho $m$, o algoritmo é o seguinte:

**Algoritmo Geral de Reamostragem** *Bootstrap*

1. Extrair uma amostra nova de tamanho $m$ com reposição da amostra original.
2. Calcular o parâmetro em estudo para a nova amostra: $\nu_1$.
3. Repetir os passos (**i**) e (**ii**) um número $M$ de vezes.
4. Traçar a distribuição desses $M$ valores amostrais $(\nu_1, \nu_2, \ldots, \nu_M)$.
5. Construir o intervalo de confiança para $\nu$ usando percentis, a partir dessa distribuição construída.

Variações deste algoritmo existem, como quão grande cada amostra deve ser (o valor de $m$) e quantas novas amostras devem ser geradas (o valor de $M$). Usar *bootstrap* para obter intervalos de confiança (ou outras estatísticas de interesse) está ganhando popularidade, porque não requer nenhum conhecimento prévio sobre os os dados subjacentes ao processo. A maioria de pacotes estatísticos permitem que se use *bootstrap*.

## 5.3 Cálculo de Limiares via *Bootstrap*

No Capítulo 4 diversos testes de novidade são mostrados usando limiares simples ou limiares duplos (intervalos de decisão). Os limiares são calculados usando a própria amostra original da medida de novidade utilizada (por exemplo, erro de quantização) via percentis (valor-$p$, intervalo de decisão, *boxplot*).

Neste capítulo, os métodos são usados da mesma forma, a única diferença é que, em vez da amostra original, um conjunto de amostras *bootstrap* geradas a partir dos dados originais será utilizada para o cálculo dos limiares.

A motivação para esse procedimento, que já explicada anteriormente, é a preocupação com a robustez dos testes de hipóteses, uma vez que se pode reduzir as probabilidades de erro Tipo I e erro Tipo II a partir do aumento do tamanho da amostra utilizada (TRIOLA, 1999).

Por exemplo, para calcular os limites do intervalo de decisão proposto no Capítulo 4 usando *bootstrap*, um conjunto de $M$ amostras *bootstrap* é criado *com reposição* a partir da amostra original de $N (N \ll M)$ erros de quantização $(e_1, e_2, \ldots, e_N)$, em que cada elemento da amostra original tem igual probabilidade de ser escolhido durante a reamostragem. A seguir, os limites inferior e superior do intervalo são computados como anteriormente, via percentis.

Intervalos de confiança podem ser calculados via amostras *bootstrap* sem que seja necessário fazer qualquer suposição sobre a distribuição de dados original, bastando pra isso que o número $M$ de amostras *bootstrap* seja suficientemente grande, por exemplo $M > 1000$ (REICH; BARAI, 1999; DiCICCIO; EFRON, 1996; EFRON; TIBSHIRANI, 1993).

Pode-se também usar *bootstrap* com o método de *boxplot* para determinar o intervalo $[\rho^-,\ \rho^+]$ definido na Equação 4.3. Como é mostrado nas simulações computacionais, essa técnica usando *bootstrap* e *boxplot* revelou-se uma das abordagens mais robustas em detecção de novidades.

Vale a pena notar que este uso da técnica *boxplot* para a detecção de novidades é semelhante àquela proposta por (MUÑOZ; MURUZáBAL, 1998). Entretanto, há duas importantes diferenças: $(i)$ neste trabalho introduziu-se uma modificação na qual o intervalo $[\rho^-,\ \rho^+]$ é calculado a partir de um conjunto de $M$ amostras *bootstrap* $(e_1^b, e_2^b, \ldots, e_M^b)$, enquanto que em (MUÑOZ; MURUZáBAL, 1998) esse intervalo é calculado diretamente dos erros de quantização gerados por um conjunto de treinamento "limpo", i.e., dos quais são previamente removidos os *outliers* conhecidos. $(ii)$ Para encontrar e remover os *outliers*, o método de Muñoz & Muruzábal (1998) necessita do cálculo adicional da matriz de distâncias inter-neurônios (*Median Interneuron Distance - MID*), essa matriz é definida como aquela cujas entradas $m_{ij}$ é a média da distância euclideana entre o vetor de pesos $\mathbf{w}_i$ e todos os $L$ neurônios vizinhos a ele, também procedia-se um mapeamento de (SAMMON JR., 1969)[2], isso torna o método quase inviável para aplicações *online* devido à excessiva carga computacional requerida. Esse excessivo custo computacional fica em desacordo com um dos princípios estabelecidos no Capítulo 2, o princípio da complexidade computacional.

## 5.4 Uma Abordagem Geral de Comparação

Propõe-se, então, uma abordagem unificada que permite comparar, sob base comum, os diferentes sistemas de detecção de novidades usando redes neurais. Esta abordagem pode ser descrita pelas seguintes etapas:

- **Passo 1:** Definir uma variável de saída da rede neural, $z_t$, a ser avaliada nos testes de novidade. É importante enfatizar que $z_t$ deve refletir a variabilidade estatística do conjunto de dados de treinamento. Para as redes discutidas nesta dissertação, as possibilidades são as seguintes:

[2] É um mapeamento não-linear que mapeia um conjunto de vetores de entrada num plano tentando preservar aproximadamente as distâncias relativas entre os vetores de entrada. Ele é vastamente usado para visualizar uma rede SOM pelo mapeamento dos valores dos vetores de peso em um plano. O mapeamento de Sammon pode ser aplicado diretamente a conjunto de dados, mas é computacionalmente muito custoso.

  - **SOM**: o erro de quantização, $z_t = e_q^{(t)}$ definido na Equação (3.17), é a escolha usual.
  - **MLP/RBF**: neste caso, há duas possibilidades.
    * Redes com um neurônio de saída: pode-se usar a própria saída da rede, i.e., $z_t = y(t)$.
    * Redes com mais de um neurônio de saída: pode-se usar a norma euclidiana da diferença entre a saída desejada, $\mathbf{d}(t)$, e a saída real da rede, $\mathbf{y}(t)$. Então, tem-se $z_t = \|\mathbf{d}(t) - \mathbf{y}(t)\|$. Para o AAMLP, pode-se tomar o erro de reconstrução, $z_t = e_r^{(t)}$ definido na Equação (3.42).

- **Passo 2:** Depois do treinamento da rede neural, calcular os valores de $z_t$ correspondentes a cada vetor do conjunto de treinamento, $\mathcal{Z} = (z_1, z_2, \ldots, z_N)$.

- **Passo 3:** Gerar uma amostra *bootstrap* $\mathcal{Z}^b$ tomada **com reposição** da amostra original $(z_1, z_2, \ldots, z_N)$, na qual cada valor $z_i$ tem igual probabilidade de ser reamostrado.

- **Passo 4:** Calcular o limiar de decisão para os testes de detecção de novidades usando o conjunto de amostras *bootstrap* $(\mathcal{Z}_1^b, \mathcal{Z}_2^b, \ldots, \mathcal{Z}_M^b)$. Neste caso, há duas possibilidades:
  - Testes de limiar simples: como o método do valor-$p$, descrito na Seção 4.2.1 ou o método de Tanaka, descrito na Equação (4.2).
  - Testes de limiar duplo: como o método ID (BARRETO et al., 2004) ou via *boxplot*.

O **Passo 4** introduz uma inovação desta dissertação em relação a outros trabalhos no campo de detecção de novidades, no que diz respeito ao cálculo de limiares, pela utilização da reamostragem *bootstrap*. O uso de reamostragem, conforme dito anteriormente, busca aumentar a confiabilidade estatística dos resultados obtidos. Outras características favoráveis da abordagem proposta são listadas abaixo:

- **confiabilidade -** é uma abordagem bem fundamentada estatisticamente e o uso de reamostragem *bootstrap* permite a geração de um grande número amostras, melhorando as estimativas dos limiares de decisão. Além disso, se é adotado o método ID, os limiares calculados corresponderão exatamente aos limites dos intervalos de confiança para a variável de saída $z_t$.

- **não-paramétrico -** nenhuma suposição sobre as propriedades estatísticas da variável $z_t$ é feita em qualquer estágio do procedimento.

- **generalidade -** permite a comparação de sistemas de detecção de novidades baseados em redes supervisionadas e não-supervisionadas sob uma base comum.
- **simplicidade -** o método é intuitivo e fácil de se aplicar.

Como é mostrado nas simulações da Seção (5.5), uma das conclusões principais extraída da comparação dos métodos neurais sob a metodologia proposta é que treinar as redes com *outliers* (exemplos negativos), quando estes estão disponíveis em número suficiente, não é a única forma de se obter desempenhos satisfatórios, segundo tem sido sugerido por alguns autores.

## 5.5 Uma Aplicação em Engenharia Biomédica

Nesta Seção, o desempenho dos métodos neurais de detecção de novidades é avaliado através de simulação computacional usando um banco de dados de câncer de mama Wolberg & Mangasarian (1990) disponibilizado para fins de pesquisa no *UCI – Machine Learning Repository* Blake & Merz (1998), um repositório de dados para aplicações em aprendizagem de máquinas. Esta escolha se deve ao fato de as aplicações biomédicas demandarem alta precisão, devido a fatores envolvendo vidas humanas.

Falsos positivos e falsos negativos em diagnósticos médicos têm implicações diferentes para o indivíduo que está sendo analisado, embora ambos devam ser reduzidos. Considere um teste de detecção de câncer executado sob a hipótese de que a pessoa é saudável (i.e. comportamento normal ou esperado). Se um câncer real não é detectado (falso negativo), o mais provável é que o indivíduo volte a sua casa e esqueça a saúde por um tempo, pelo menos até a visita médica seguinte. Este é um problema sério, uma vez que a detecção de um tumor maligno nos estágios iniciais de seu desenvolvimento pode vir a ser um fator crucial para o sucesso do tratamento. Se um falso câncer é detectado (falso positivo), o indivíduo provavelmente fará investigações adicionais sobre a doença e descobrirá finalmente que o diagnóstico precedente estava errado. Nesta caso, além dos custos adicionais para os novos exames, a pessoa é exposta a um estresse psicológico indesejável enquanto espera pelos resultados finais, muito embora ele não corra risco de vida pelo erro de diagnóstico. Assim, é importante avaliar os métodos de detecção apresentados nas seções anteriores quanto às taxas de falsos positivos e falsos negativos.

### 5.5.1 Dados Experimentais

O conjunto de dados utilizado consiste em 699 vetores de dimensão 9, cujos atributos $x_i$, $(i = 1, 2, \ldots, 9)$ são os seguintes:

- espessura do nódulo
- uniformidade do tamanho da célula
- uniformidade da forma da célula
- adesão marginal
- tamanho da célula epitelial simples
- núcleos descobertos
- nível de cromatina nuclear
- núcleos normais
- mitoses;

Todos os atributos têm valores dentro do intervalo $[1-10]$, a maneira como esses valores são atribuídos por especialistas da área médica pode ser encontrada em (DUTRA et al., 2004). Os vetores de dados são re-escalados através de um procedimento de normalização severa (*hard normalization*)[3] para o intervalo $[0-1]$.

Foram eliminados 16 exemplos que continham atributos sem valor definido (*NOT A NUMBER* – NaN). Dos 683 vetores restantes, 444 correspondem a tumores benignos e 239 a malignos.

Dos 444 vetores "normais", 355 deles (aproximadamente 80%) são selecionados para o treinamento das redes neurais. Dos 89 vetores normais restantes usados para os testes de validação, 30 são substituídos por vetores anormais, escolhidos aleatoriamente do conjunto de 239 vetores anormais.

A inclusão de vetores anormais no conjunto de testes é necessária para se avaliar as taxas de erro falso negativo (erro Tipo II). Se apenas exemplos de vetores normais compusessem o conjunto de teste, poderia-se somente estimar as taxas de erro falso positivo (erro Tipo I). Este procedimento é repetido em 100 iterações de simulação, e as taxas médias de erro são calculadas no final.

### 5.5.2 Configurações das Redes Neurais e das Simulações

Para detectores de novidades baseados na rede SOM, a variável de saída $z_t$ é o erro da quantização. Para detectores baseados nas redes MLP e RBF utiliza-se um neurônio na

[3] Detalhes sobre esses métodos de pré-processamento estão no Apêndice A.

camada de saída e, portanto, $z_t = y(t)$, a própria saída da rede. Para o AAMLP utiliza-se o erro de reconstrução.

Para todos os algoritmos neurais os limiares de decisão são determinados a partir da amostra de *bootstrap* para $z_t$ usando os seguintes métodos: valor-$p$, *boxplot* e ID. Adicionalmente, para detectores SOM, analisa-se também o método de Tanaka usado para calcular limiares de decisão.

Todos os testes são executados usando uma rede SOM unidimensional. As redes MLP e GMLP consistiram em uma única camada escondida de neurônios treinados com o algoritmo padrão de retropropagação do erro (*error backpropagation*) com fator de momento para acelerar a convergência. Uma função de ativação do tipo logística é adotada para todos os neurônios da rede MLP e uma função de ativação do tipo gaussiana é adotada para os neurônios intermediários da rede GMLP.

A rede RBF utilizada consistiu em uma primeira camada de funções de base gaussianas cujos centros $\mathbf{c}_i$ são calculados por uma rede SOM.

As redes não-supervisionadas e supervisionadas são avaliadas segundo a abordagem proposta de forma a verificar seus desempenhos em detecção de novidades em face a fatores como presença de *outliers* e sensibilidade a parâmetros de treinamento: número de neurônios, número de épocas de treinamento e tamanho do conjunto de dados de treinamento.

### 5.5.3 Resultados

Os resultados obtidos são apresentados abaixo separadamente para as redes não-supervisionadas e para a rede SOM.

#### 5.5.3.1 Rede SOM

O primeiro conjunto de simulações compara a habilidade da detecção de novidades dos diferentes métodos que usam a rede SOM.

As taxas de erro falso negativo obtidas para os detectores SOM em função do número dos neurônios são mostradas na Figura 5.1. Cada modelo neural é treinado por 100 épocas.

O segundo conjunto de simulações avalia a sensibilidade dos detectores SOM a mudanças no número de épocas de treinamento, como mostrado na Figura 5.3. Os parâmetros de treinamento usados são os mesmos do primeiro conjunto de simulações, exceto pelo número dos neurônios, que é fixado em 40. O desempenho geral mantém-se como na

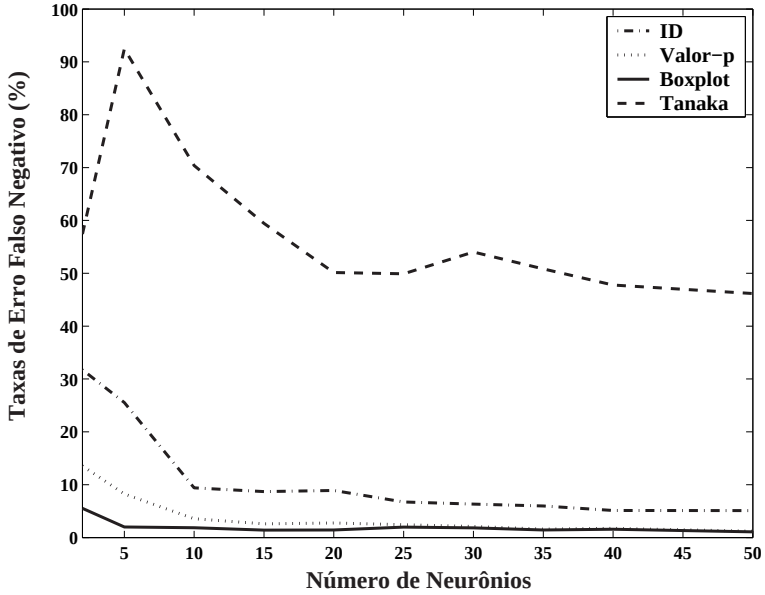


**Figura 5.1:** Taxas médias de erro falso negativo (%) em função do aumento do número de neurônios da rede SOM.

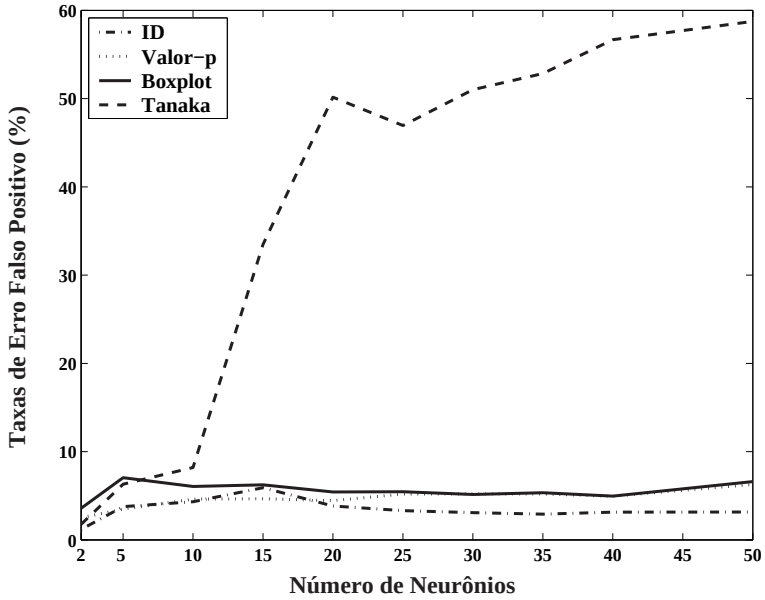


**Figura 5.2:** Taxas médias de erro falso positivo (%) em função do aumento do número de neurônios da rede SOM.

Figura 5.1, com o par (SOM, *boxplot*) conseguindo as menores taxas de erro falso negativo.

Com testes de limiar-duplo, pode-se detectar *outliers* tanto nas regiões de elevado erro de quantização (EQ) quanto em regiões de EQ baixo. Como discutido no Capítulo 4, este tipo de *outlier* (*outliers* desconhecidos) pode ser resultado de erros na fase de rotulação dos dados. Se *outliers* desconhecidos estiverem presentes no conjunto de treinamento, alguns neurônios podem ser atraídos para esses padrões espúrios, de modo que, no futuro,

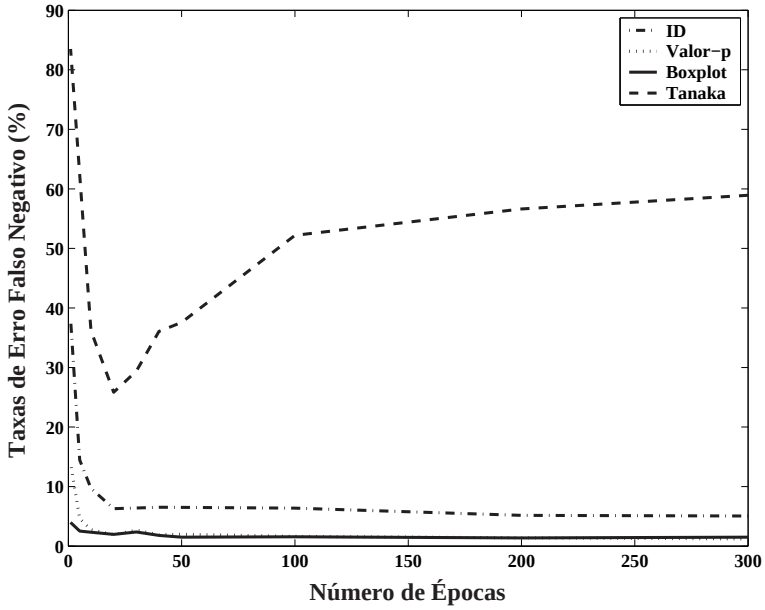


**Figura 5.3:** Taxas médias de erro falso negativo (%) em função do aumento do número de épocas de treinamento da rede SOM.

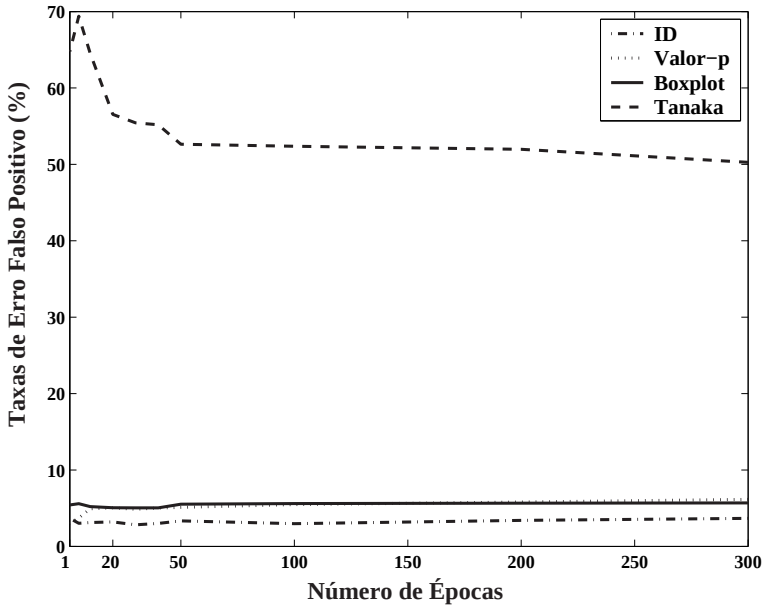


**Figura 5.4:** Taxas médias de erro falso positivo (%) em função do aumento do número de épocas de treinamento da rede SOM.

alguns *outliers* ativarão esses neurônios e o valor para o EQ provavelmente será baixo. Somente métodos de detecção de novidades baseados em limiar duplo, tais como *boxplot* ou ID, podem detectar *outliers* neste caso.

O par (SOM, ID), que na teoria poderia também detectar *outliers* na região de baixo EQ, obteve um desempenho somente melhor do que o par (SOM, Tanaka). Isto pode

ser devido ao fato que a grande maioria de *outliers* desconhecidos está na região de EQ elevado, provavelmente devido à baixa ocorrência de *outliers* desconhecidos (tais como dados erradamente rotulados como normais) no conjunto de dados de treinamento, o que implicitamente revela a boa qualidade do conjunto de dados escolhido.

Quanto à variação das taxas de erro falso positivo dos detectores SOM com o número de neurônios e com o número de épocas de treinamento, mostrada nas figuras 5.2 e 5.4, observa-se um melhor desempenho do método do ID, o que é concordante com os resultados obtidos no Capítulo 4.

Pode-se notar que o desempenho do par (SOM, Tanaka) piora com o aumento do número de épocas de treinamento. Isto ocorre por causa da própria natureza do teste de Tanaka. Uma vez que a rede SOM tem mais tempo para convergir, ela cobre melhor a distribuição dos dados. Então, observa-se que o erro de quantização tende a diminuir, enquanto que o limiar de novidade $\rho^+$, que é computado segundo a Equação (4.2), tende a se estabilizar num valor constante, devido à melhor distribuição dos centróides sobre a massa de dados. Assim, enquanto a rede consegue uma melhor representação dos dados, torna-se cada vez mais raro observar na fase de teses $e_q^{novo} > \rho^+$, e, portanto, raramente se anuncia uma novidade, mesmo quando o vetor de dados apresentado é verdadeiramente novo. A Figura 5.5 mostra o comportamento anteriormente explicado. O método de Tanaka, portanto, contradiz o senso comum que diz que quanto melhor a representação dos dados, melhor é o resultado apresentado pela rede neural.

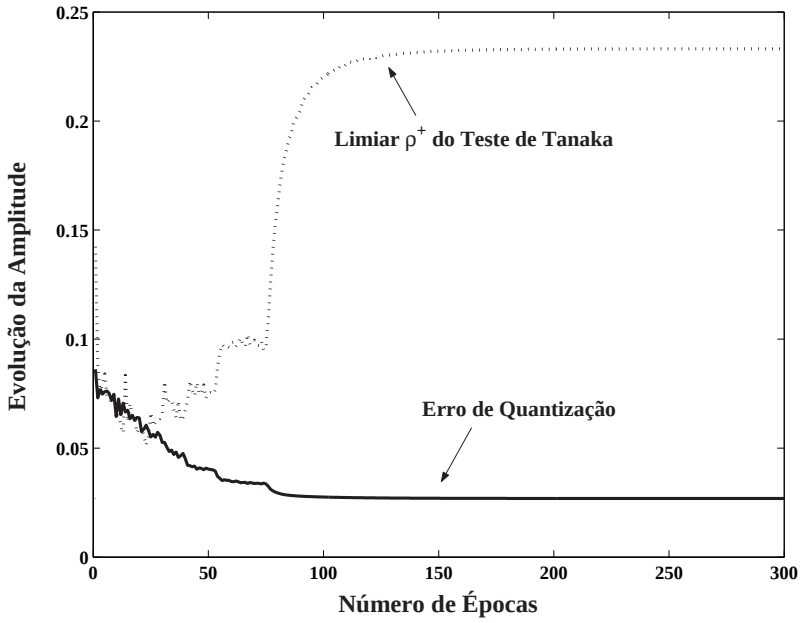


**Figura 5.5:** Evolução dos valores médios do erro de quantização e do limiar do teste de Tanaka em função do aumento do número de épocas de treinamento da rede SOM.

O terceiro conjunto de simulações compara a exatidão de detectores SOM com res-

peito ao tamanho do conjunto de treinamento, em função da percentagem do total de dados disponíveis. A finalidade deste teste é indicar qual método requer menos dados para uma boa precisão. Na Figura 5.6 observa-se que cada um dos métodos para o cálculo de limiares mostrou melhor desempenho em diferentes tamanhos do conjunto de treinamento. Quanto aos dois melhores métodos, *boxplot* e valor-$p$, não se pode dizer que houve mudanças relevantes em seus desempenhos a partir do tamanho do conjunto de treinamento ser correspondente a 60% do total de dados disponíveis. Para esses testes, o número dos neurônios e o número de épocas de treinamento são ajustados em 40 e 100, respectivamente. Para simplificar a Figura 5.6, o teste de Tanaka é omitido por ter apresentado desempenho demasiadamente baixo.

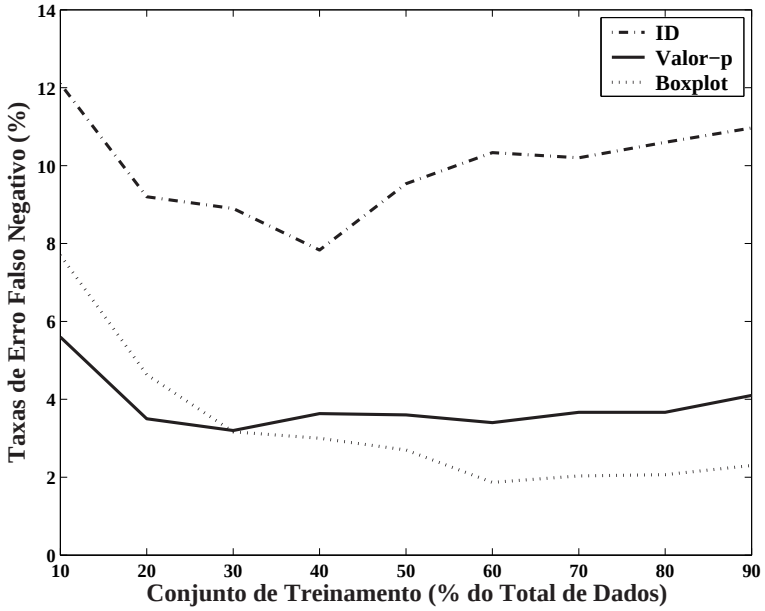


**Figura 5.6:** Taxas médias de erro falso negativo (%) em função do tamanho do conjunto de treinamento.

#### 5.5.3.2 Redes Supervisionadas

Os mesmos testes descritos anteriormente para detectores SOM são repetidos aqui para os métodos supervisionados (MLP, AAMLP, GMLP e RBF).

O primeiro conjunto de simulações avalia a taxa de erro falso negativo em função do número de neurônios escondidos. Para esses testes, cada rede MLP é treinada por 1000 épocas somente com vetores de dados normais (**classificação binária**). A taxa de aprendizagem e o fator de momento são ajustados respectivamente em 0,35 e 0,5.

Para maior clareza, os resultados são mostrados somente para os métodos de determinação de limiares de decisão valor-$p$ (Figura 5.7) e *boxplot* (Figura 5.8), pois eles apresentam melhor desempenho, respectivamente pares (MLP, valor-$p$) e (RBF,*boxplot*).

Essas figuras ilustram também que alguns métodos para cálculo de limiares de decisão são inviáveis em conjunção com determinadas redes neurais supervisionadas, por exemplo, os pares (RBF,valor-$p$) e (MLP,*boxplot*).

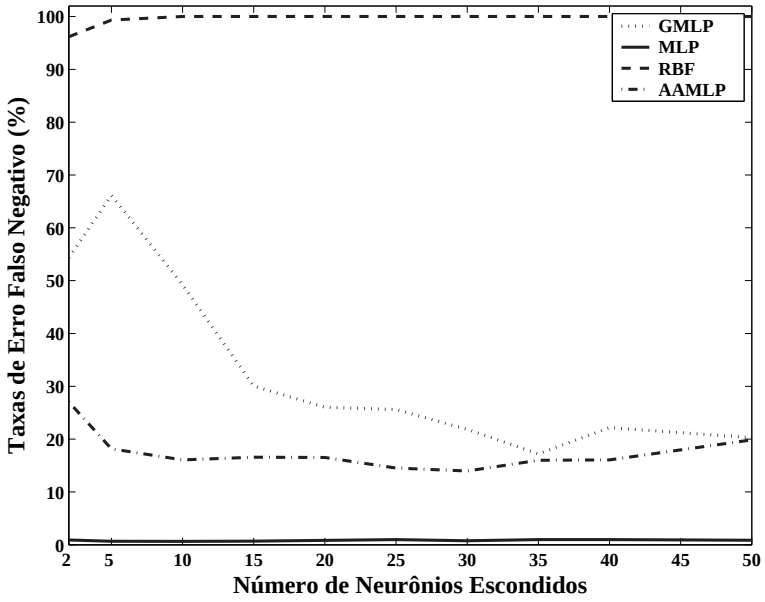


**Figura 5.7:** Taxas médias de erro falso negativo (%) em função do número de neurônios na camada escondida (redes supervisionadas), usando limiar de decisão calculado pelo método do valor-$p$.

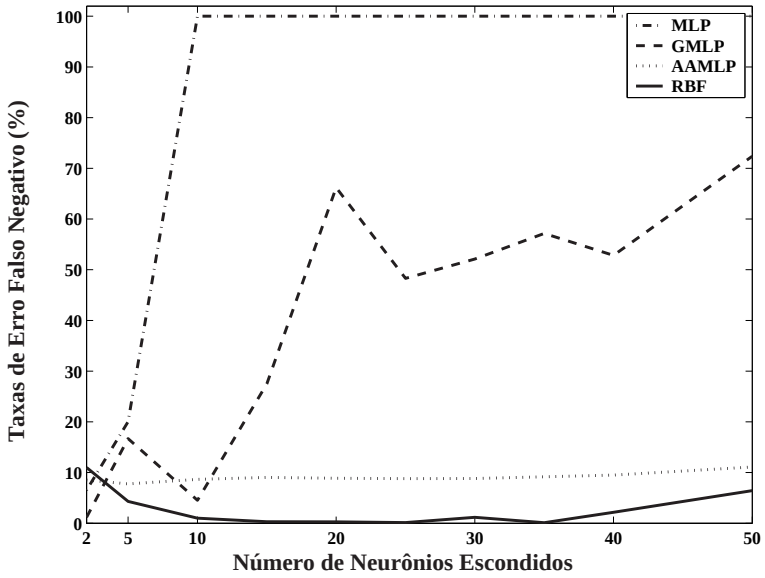


**Figura 5.8:** Taxas médias de erro falso negativo (%) em função do número de neurônios na camada escondida (redes supervisionadas), usando limiares de decisão calculados pelo método de *boxplot*.

Para ilustrar como a presença de *outliers* nos dados de treinamento tem influência no desempenho dos detectores de novidades supervisionados e não-supervisionados, treinam-

se os pares (SOM, *boxplot*) e (MLP, valor-$p$) usando dados que contém um determinado número de *outliers* falsificados (*fake outliers*), isto é, vetores de dados originalmente anormais que são rotulados intencionalmente como sendo normais, numa simulação de erros de rotulação. O resultado é mostrado na Figura 5.9. Para fins de comparação, treina-se também um classificador MLP padrão para um problema de 2 classes (normal/anormal), usando para decisão a regra WTA. O par (MLP, regra WTA) é treinado usando os rótulos verdadeiros dos vetores de dados, ou seja, nesse caso não houve falsificação de rótulos.

Nesse teste, é necessário esclarecer o papel dos *outliers* em cada um dos modelos de classificação abordados. Para os classificadores bivalentes (SOM, *boxplot*) e (MLP, valor-$p$) a presença de *outliers* é indesejada e deve piorar o desempenho desses sistemas, uma vez que eles tendem a generalizar os dados discrepantes como dados normais. Já para o classificador multivalente (MLP, regra WTA), a presença de *outliers*, que funcionam como exemplos negativos, juntamente com os dados normais, que funcionam como exemplos positivos, tendem a tornar o treinamento desse classificador mais efetivo, melhorando o posicionamento da fronteira de decisão entre as duas classes (normal/anormal).

A Figura 5.9 mostra como as taxas de erro falso negativo variam com o número relativo (porcentagem) de *outliers* presentes no conjunto de treinamento. Como esperado, o desempenho dos **classificadores bivalentes**, (SOM, *boxplot*) e (MLP, valor-$p$), deteriora-se com a presença dos *outliers*, enquanto que o desempenho dos **classificadores multivalentes** melhora. Isto ocorre porque os classificadores binários aprendem erroneamente a considerar *outliers* como vetores de dados normais, diminuindo sua sensibilidade aos verdadeiros *outliers*, ou seja, às novidades. Para o classificador multivalente (MLP, regra WTA), a sensibilidade à novidade aumenta, visto que o classificador aprende a separar melhor o que é normal do que é anormal. Vale destacar que o desempenho do classificador MLP para duas classes melhora somente quando mais de 30% dos padrões de treinamento são exemplos negativos (anormais). Entretanto, geralmente não é possível (ou não é economicamente viável) obter um número tão elevado de vetores anormais.

Ainda em relação à Figura 5.9 nota-se que o desempenho dos métodos binários, (SOM, *boxplot*) e (MLP, valor-$p$), é melhor do que o do classificador multivalente, (MLP, regra WTA), quando a porcentagem de *outliers* é inferior a 10% do conjunto total de dados de treinamento.

Finalmente, a Tabela 5.1 apresenta os melhores resultados obtidos para o conjunto de dados utilizado neste artigo. Em termos de erro falso negativo, o melhor desempenho global é obtido pelo par (RBF, *boxplot*). Convém advertir que o resultado mostrado para o par (MLP, regra WTA) é para um conjunto de treinamento equilibrado, que contém 50% de vetores anormais e 50% de vetores normais, sendo mais propriamente um classificador que um detector de novidades.

**Tabela 5.1:** Melhores desempenhos (%) para a tarefa de detecção de novidades.

| | Falsos Negativos | | Falsos Positivos | |
|---|---|---|---|---|
| **MODELO / CLASSIFICADOR** | **Média** | **Variância** | **Média** | **Variância** |
| **(RBF,** *boxplot***) / Bivalente** | 0.1 | 0.1 | 9.9 | 11.8 |
| **(MLP, valor-*p*) / Bivalente** | 0.6 | 0.5 | 3.4 | 3.5 |
| **(SOM,** *boxplot***) / Bivalente** | 2.0 | 1.0 | 3.7 | 2.9 |
| **(OLAM,** *boxplot***) / Bivalente** | 3.3 | 41.0 | 5.2 | 11.9 |
| **(MLP, regra WTA) / Multivalente** | 0.9 | 0.9 | 3.5 | 3.2 |

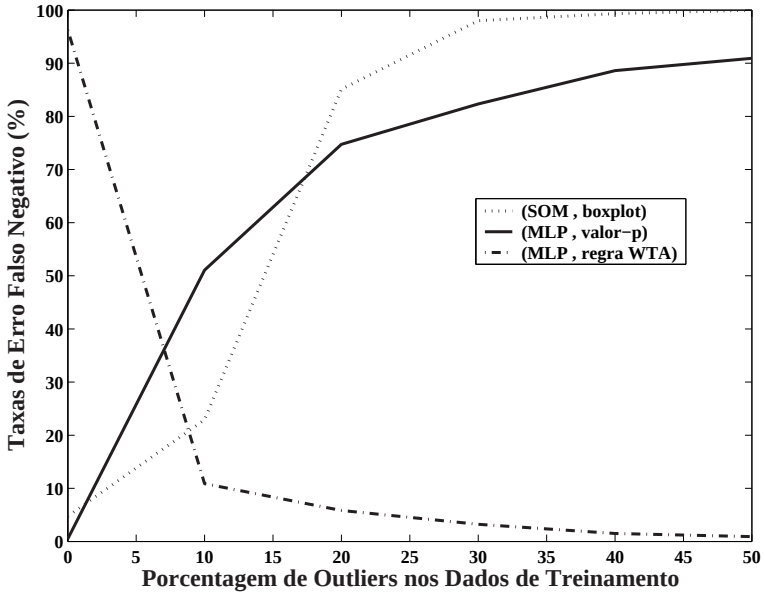


**Figura 5.9:** Taxas médias de erro falso negativo (%) em função do número de *outliers* presentes nos dados de treinamento.

Uma informação importante da Tabela 5.1 é o tipo de classificador que foi construído por cada um dos modelos neurais. Convém ressaltar que os classificadores bivalentes são treinados apenas com dados de uma única categoria (exemplos positivos), enquanto que os classificadores multivalentes são treinados com dados de todas as categorias em questão (exemplos positivos e exemplos negetivos).

## 5.6 Conclusão

Este capítulo propôs uma abordagem não-paramétrica para a comparação do desempenho de diferentes redes neurais artificiais aplicadas à tarefa de detecção de novidades. Esta abordagem permite avaliar as propriedades computacionais de ambos os tipos de rede, supervisionadas e não-supervisionadas. Também permite testar a efetividade de diferentes técnicas de cálculo de limiares de decisão, quando usadas em conjunção com

diferentes algoritmos de redes neurais, tais como SOM, MLP e RBF.

Uma aplicação de detecção de tumores malignos em mamografias trouxe a validação prática para os métodos, permitindo realizar uma comparação entre diferentes métodos de detecção de novidades baseados em redes neurais (supervisionadas e não supervisionadas) e diferentes estratégias de cálculo de limiares de decisão.

O cenário apresentado pelos resultados obtidos indicou importantes conclusões sobre as importantes questões levantadas no início deste capítulo:

- **Quanto à estratégia de classificação:** um dos principais resultados obtidos vem da observação de que, na falta de uma boa quantidade de exemplos negativos (pelo menos 30% do total de dados disponíveis) o desempenho dos classificadores bivalentes é melhor que o desempenho de classificadores multivalentes. Como tal quantidade de exemplos negativos é raramente possível, conclui-se que se deve optar por métodos de detecção de novidades baseados em **classificação binária**, ou seja, aqueles que usam exemplos de uma única classe (exemplos positivos ou normais).

- **Quanto ao tipo de rede neural:** as redes supervisionadas apresentam melhor desempenho. No entanto, em aplicações nas quais a velocidade de treinamento seja mais importante que a exatidão dos testes de novidade (por exemplo, aplicações em tempo real), pode-se utilizar a rede SOM, que obteve taxas aceitáveis para ambos os erros, falso positivo e falso negativo. O método baseado no Filtro Linear de Novidades (OLAM) também pode ser utilizado em certos casos, nos quais a velocidade seja mais crítica que o desempenho.

- **Quanto ao tipo de teste:** há três evidências que levam a concluir-se pela indicação do uso dos testes bilaterais. A primeira é que dos três melhores desempenhos mostrados na Tabela 5.1, dois são obtidos com o método *boxplot*. A segunda é que o método do intervalo de decisão mostrou as menores taxas de erro falso positivo. A terceira, mais conceitual, é o fato de que testes bilaterais são os únicos capazes de detectar novidades mesmo quando *outliers* desconhecidos são usados no treinamento, diminuindo, portanto, o efeito ocultação discutido no Capítulo 2.

- **Quanto à regra para cálculo do(s) limiar(es) de decisão:** a regra heurística WTA para classificadores multivalentes mostrou-se utilizável apenas em condições muito específicas. A regra heurística de Tanaka mostra-se ineficaz, tendo possivelmente desempenho aceitável apenas em condições bastante favoráveis. Portanto, conclui-se ser preferível a utilização de métodos com maior fundamentação teórica, como os três estudados neste capítulo, a saber, *boxplot*, valor-$p$ e intervalo de decisão.

O capítulo seguinte conclui esta dissertação realçando todos os principais resultados e conclusões finais. Futuros desdobramentos e ampliações possíveis ao escopo deste trabalho são também sugeridas.

# 6 Conclusões e Trabalhos Futuros

Esta dissertação tratou da detecção de novidades, um ramo do reconhecimento de padrões que tem por objetivo reportar, através de análise e modelagem de uma massa de dados, aquelas observações, antigas ou futuras, que apresentam algum desvio em relação à grande maioria do restante da massa de dados.

A exemplo de outros ramos da mineração de dados e do reconhecimento de padrões, redes neurais artificiais têm sido empregadas com sucesso em tarefas de detecção de novidades e afins (MARKOU; SINGH, 2003b), com destque para a utilização de redes neurais supervisionadas, principalmente as redes MLP e RBF.

Entretanto, um importante ramo das redes neurais artificiais não-supervisionadas, chamadas redes competitivas, apresentam características que podem ser úteis em aplicações de detecção de novidades, como a compressão de dados, através da construção de uma representação compacta dos dados de entrada, chamado *mapeamento de características*, e o uso de uma medida chamada *erro de quantização*, inerente aos modelos neurais competitivos, para controle da qualidade desse mapeamento.

Tendo em vista tais qualidades das redes competitivas, esta dissertação propôs um novo método não-paramétrico para a detecção de novidades inspirado no conceito estatístico de intervalos de confiança. Esse método, chamado *Intervalos de Decisão – ID*, consiste em um teste duplo, um multivariado (ID global) e outro univariado (ID local) que permite uma maior confiabilidade nos alarmes de novidade gerados pelo método.

Uma aplicação de detecção de anomalias em sistemas celulares de terceira geração foi utilizada para testar o método proposto, que superou em desempenho os métodos de limiar simples, principalmente quando se observou o número de alarmes falsos gerados. Dentre as redes competitivas analisadas, a rede SOM apresentou os melhores resultados, sendo, por isso, mantida para as simulações do capítulo seguinte. Entretanto, um resultado importante observado nas simulações foi que modelos competitivos simples, tais como os WTA e o FSCL, podem produzir resultados equivalentes em tarefas da detecção de novidades.

Em seguida, ao buscar realizar testes comparativos mais abrangentes, permitindo a comparação entre diferentes paradigmas neurais e de cálculo de limiares de decisão, em diferentes domínios de aplicação, foram encontradas algumas lacunas na literatura, pois poucos trabalhos têm sido propostos explorando plataformas de comparação, nem fornecem resultados claros indicando, por exemplo, qual técnica funciona melhor para qual tipo de dados, ou qual tem seu desempenho menos degradado pela presença de *outliers* desconhecidos nos dados de treinamento.

Dessa forma, este trabalho culminou na elaboração de uma abordagem unificadora para o cálculo/escolha de limiares de decisão, tanto para redes neurais supervisionadas como não-supervisionadas. Essa abordagem consiste na generalização não somente do método proposto dos intervalos de decisão para o uso em redes supervisionadas, mas também generalizar o uso de outras técnicas para o cálculo de limiares de decisão antes restritas a redes competitivas.

Uma comparação dos diferentes sistemas de detecção de novidades baseados em redes neurais artificiais, utilizando diferentes técnicas de cálculo de limiares de decisão, foi realizada em uma aplicação em Engenharia Biomédica.

A abordagem unificadora permitiu extrair importantes conclusões, esclarecendo pontos que permaneciam obscuros. Elas estão detalhadas no Capítulo 5, mas vale destacar as seguintes:

- Redes competitivas podem ter desempenho tão bom quanto redes supervisionadas de maior prestígio, como a rede MLP;
- O uso, no treinamento supervisionado, de exemplos negativos juntamente com exemplos positivos melhora o desempenho do detector de novidades (classificador multivalente). Entretanto se a quantidade de exemplos negativos não é suficiente (pelo menos 30% dos dados de treinamento) pode ser melhor utilizar apenas os exemplos positivos e treinar um classificador neural bivalente;
- Testes que utilizam limiares duplos apresentaram melhor desempenho que aqueles que utilizam apenas um limiar de decisão.

Portanto, as principais contribuições deste trabalho podem ser sumarizadas em dois pontos:

- O método dos intervalos de decisão, que consiste em dois testes (um global e um local) ambos de limiar duplo, é inovador ao propor duas etapas de decisão, e mostrou desempenho satisfatório, sobretudo na redução de falsos alarmes;

- A abordagem unificadora proposta, que é um primeiro passo no sentido de permitir comparações mais efetivas e conclusivas entre diferentes algoritmos e estratégias de detecção de novidades. Permite, de maneira mais objetiva, decidir que método é mais indicado para um determinado cenário de dados.

Trabalhos futuros explorarão os métodos robustos da estatística aos algoritmos neurais estudados nesta dissertação, a fim de diminuir a sensibilidade dos algoritmos à presença de *outliers*. Uma desenvolvimento natural do trabalho atual envolve sua aplicação em tarefas de detecção preditiva de anomalias. Nesse caso, anomalias devem ser detectadas previamente para permitir que medidas preventivas sejam tomadas. Séries temporais de KPIs, por exemplo, seriam monitoradas pela rede neural a fim detectar tendências problemáticas no sistema celular. Para gerar dados da série de tempo, requer-se um simulador inteiramente dinâmico da rede celular (LAIHO et al., 2001). Para extrair informação dos dados dinâmicos deve-se usar versões temporais das redes neurais competitivas.

Num modo mais geral, pode-se estender a utilização dos intervalos de decisão para detecção de novidades em séries temporais (e.g. séries financeiras). Para tanto, diferentes arquiteturas de redes neurais recorrentes, como a rede de Elman ou a SOM recursiva, usando diferentes métodos de cálculo de limiares de decisão. Uma aplicação de interesse é a detecção de intrusos em redes de computadores (*Intrusion Detection System – IDS*), na qual comportamentos anômalos são detectados baseados na análise temporal do tráfego de dados numa rede de computadores.

# *APÊNDICE A* – Pré-processamento de Dados

Quando se utiliza RNAs para análise de dados, seja em aplicações de classificação (como detecção de novidades) ou de predição (como predição de séries temporais) é importante ter cuidado com os dados que serão usados para o treinamento das redes, de forma a maximizar o desempenho das mesmas.

Este apêndice discute algumas questões relevantes quanto à preparação dos dados quando se usa RNAs. Algumas estratégias-padrão de preparo dos dados e métodos de pré-processamento que foram utilizados neste trabalho serão discutidos.

## A.1 Seleção das Variáveis de Entrada

A seleção dos dados pode ser uma tarefa difícil. O poder de uma RNA está associado à qualidade dos dados usados para treiná-la. Quanto mais informação nos dados, mais informação será aprendida pela rede. Se entradas de dados importantes faltarem, o efeito no desempenho de rede neural pode ser significativo.

Levar a cabo uma aplicação funcional usando redes neurais pode ser ainda mais difícil sem uma compreensão adequada do domínio do problema, uma vez que normalmente muitas variáveis são disponíveis para se montar o vetor de características. Com a respeito à seleção das variáveis de entrada, deve-se ter em mente alguns pontos importantes:

- **Escolher as variáveis mais relevantes:** aquelas de maior impacto no tipo de experimento que se quer realizar;
- **Escolher as variáveis que trazem maior informação:** são, geralmente, aquelas de maior variabilidade;
- **Evitar relações triviais entre as variáveis:** uma variável que pode ser derivada de outras variáveis já presentes no vetor de dados, devem ser evitadas, uma vez que

aumentam o custo computacional do sistema, sem contribuição relevante em termos de informação.

Observar os princípios acima é essencial, mas não suficiente. Mesmo após a correta seleção de variáveis, ainda restam problemas como:

- **Diferentes escalas de valores para cada componente:** usualmente, há componentes que representam grandezas diversas, tendo suas próprias escalas de valores. Isso pode causar uma polarização indesejada nos valores dos pesos da rede neural, pela importância indevida que algumas componentes terão, simplesmente por estarem representadas numa escala de valores mais elevada;
- **Presença de ruído:** o ruído pode ter origem variada, desde imprecisões nos sensores de medição, até interferência eletromagnética nos equipamentos. Ruído é especialmente nocivo quando trabalha-se com sinais de pequena amplitude.
- **Presença de *outliers*:** há duas importantes razões para se descobrir a presença de observações discrepantes no conjunto de dados. A primeira é que a descoberta de *outliers* permite sua remoção, uma vez que, usualmente, essas observações levam a erros na modelagem. A segunda é que *outliers* podem ser uma importante fonte de informação, uma vez que podem indicar mudanças na dinâmica do processo que gerou os dados (WEBB, 2002).

Para minimizar ou corrigir esses problemas existem técnicas amplamente empregadas que serão descritas a seguir.

## A.2 Pré-processamento dos Dados de Entrada

Uma vez que os dados de entrada mais apropriados foram selecionados, eles devem ser pré-processados. Sem isso, a rede neural poderá não produzir resultados adequados. Em geral, as decisões feitas nesta fase do desenvolvimento são críticas para o desempenho de uma rede. O conhecimento do domínio é importante no momento de escolher o método pré-processamento adequado, que pode aumentar a habilidade da rede neural em aprender melhor a associação entre entradas e saídas (no caso supervisionado), ou as principais características estatísticas subjacentes à massa de dados (no caso não-supervisionado).

### A.2.1 Normalização

A normalização é um dos métodos de pré-processamento mais utilizados. A normalização é uma transformação executada, componente a componente, para distribuir

uniformemente os dados e colocá-los em uma escala aceitável para a rede neural. O objetivo é assegurar-se de que a distribuição estatística dos valores para cada entrada e saída seja aproximadamente uniforme. Além disso, garantir que os valores de cada componente serão colocados numa mesma escala de valores.

Três técnicas de normalização dos dados serão descritas. Devido a peculiaridades tanto dos dados como do tipo de RNA a ser usada, diferentes aplicações podem ter melhores resultados com uma ou outra normalização, portanto, é interessante sempre que possível, fazer testes comparativos usando os diferentes tipos de normalização.

- **Normalização suave** - As distribuições das componentes individuais , $x_j$, $j = 1, 2, \ldots, m$ são normalizadas para terem média zero e variância unitária

$$x_j^{new} = \frac{x_j - \bar{x}_j}{\sigma_j} \tag{A.1}$$

em que

$$\bar{x}_j = \frac{1}{m}\sum_{j=1}^{m} x_j \quad \text{and} \quad \sigma_j = \sqrt{\frac{1}{m-1}\sum_{j=1}^{m}(x_j - \bar{x_j})^2} \tag{A.2}$$

- **Normalização severa**- As componentes $x_j$ são re-escaladas para o intervalo $[0; 1]$:

$$x_j^{new} = \frac{x_j - \min(x_j)}{\max(x_j) - \min(x_j)} \tag{A.3}$$

em que $\max(x_j)$ e $\min(x_j)$ são os valores máximo e mínimo de $x_j$, respectivamente.

- **Descorrelação** - Os vetores de dados $\mathbf{x}$ são transformados em novos vetores $\mathbf{v}$, cujas componentes são descorrelacionadas e suas variâncias são unitárias. Em outras palavras, a matriz de covariância é a matriz identidade $E\{\mathbf{v}\mathbf{v}^T\} = \mathbf{I}$. Esse procedimento é, usualmente, feito através da decomposição pelos autovalores (*Eigenvalue Decomposition – EVD*) da matriz de covariância $\mathbf{R}_x = E\{\mathbf{x}\mathbf{x}^T\}$ dos dados originais (WEBB, 2002; PRINCIPE et al., 2000).

Não se pode indicar um desses métodos como o melhor, uma vez que seu desempenho depende da aplicação ou da própria natureza dos dados em análise. Para ilustrar, como o pré-processamento pode influir no desempenho de um classsificador baseado em RNAs, utilizaram-se os três métodos de normalização descritos acima nos dados de câncer de mama usados no Capítulo 5 e verificou-se sua influência nas taxas de erro obtidas. Para este conjunto de dados, em particular, a *normalização suave* e a *descorrelação* tiveram desempenho pior do que a *normalização severa*, como pode ser visto na Figura A.1, na qual o número dos neurônios para o par (SOM, *boxplot*) é variado.

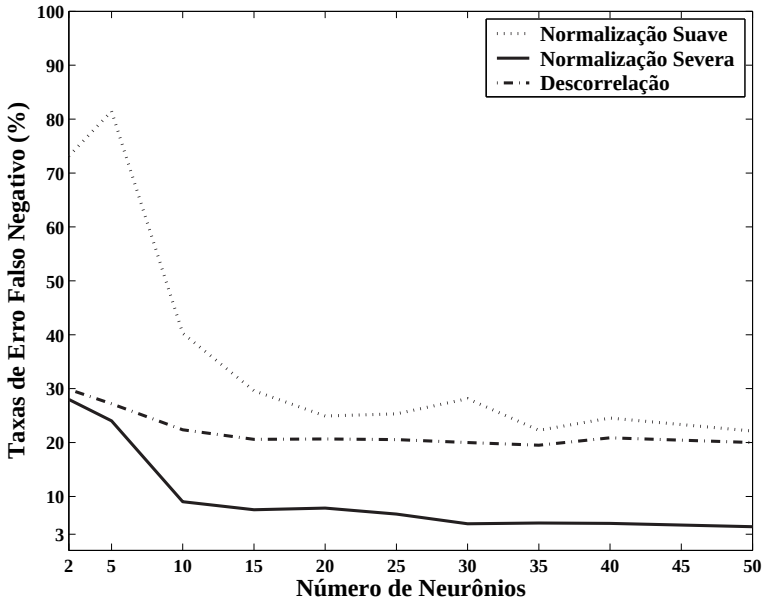


**Figura A.1:** Taxas médias de erro falso negativo (%) obtidas pelo par (SOM, *boxplot*) para os três métodos de pré-processamento estudados, variando-se o número de neurônios da rede SOM.

Portanto, estes resultados confirmam que diferentes métodos de pré-processamento dos dados podem resultar em diferentes taxas de erro. Desta forma, sugere-se que um certo número desses métodos deve ser testado sempre que possível.

### A.2.2 Redução de Ruído

O uso de redes competitivas por si só já reduz o efeito do ruído, uma vez que os centróides convergem para valores médios de subgrupos dos dados.

Se outras RNAs foram usadas ou se uma filtragem de ruído mais severa for necessária pode-se usar ferramentas matemáticas adequadas, como usar médias móveis simples ou exponenciais. Outras técnicas mais avançadas da redução de ruído são a Transformada Rápida de Fourier (FFT) e o Periodograma de Welch.

Uma vez que a arquitetura de rede foi selecionada e as entradas escolhidas e pré-processadas, a rede neural está pronta para ser treinada.

## A.3 Remoção de Outliers

A remoção de *outliers* é feita através do uso prévio de algum método de detecção de *outliers*, que foram objeto de estudo do Capítulo 2 desta dissertação.

Especificamente para a detecção de novidades, a remoção de *outliers* é desejável sempre que se usa uma estratégia de *classificação binária* e a manutenção de *outliers* devidamente rotulados é desejável em problemas que usam a estratégia de *classificação multi-classes*.

Dentro desta questão, Ypma & Duin (1997) comentam sobre a não disponibilidade de amostras que descrevam exatamente as anomalias (ou estados anormais) dos diversos sistemas de interesse e reivindicam que a melhor solução deve ser construir exatamente uma representação da operação normal desses sistemas e detectar anomalias (comportamento anormal) a partir de desvios desta representação de normalidade. Scholkopf et al. (2000) aborda a detecção de novidades usando o princípio de Vapnik que diz para nunca resolver um problema que seja mais geral do que aquele que se está realmente interessado. Se o interesse é somente detectar novos eventos, não é sempre necessário estimar um modelo de toda a densidade dos dados.

Como indicado mais cedo nesta dissertação, uma abordagem comum à detecção de novidades (para alguns autores a abordagem genuína) é tratar a detecção de novidades como um problema de classificação binária, em que se está interessado em construir uma boa representação para uma única classe e definir, então, um método para testar se novos vetores de dados são membros desta classe. Nesse método, o conjunto de dados de treinamento deve ser idealmente livre de *outliers*, inclusive dos desconhecidos. Os defensores desse ponto de vista sustentam que um sistema de detecção de novidades pode ter seu desempenho melhorado se associado com algum mecanismo de "limpeza" de dados (*data cleaning*) baseada na remoção de *outliers*.

### A.3.1 Um Novo Método de Remoção de *Outliers*

O método proposto no Capítulo 4 desta dissertação pode ser utilizado para a limpeza automática dos dados, removendo os vetores anômalos (*outliers* indesejáveis) do conjunto de treinamento, permitindo o retreinamento do modelo neural com um conjunto de dados limpo. O procedimento de limpeza proposto dos dados é detalhado em seguida:

- **Etapa 1:** escolher um modelo neural e calcular os limiares de decisão de acordo com a metodologia geral descrita na Seção (5.4).
- **Etapa 2:** aplicar testes de novidade usando os próprios vetores de dados de treinamento. Obviamente, para uma rede bem treinada, somente alguns destes vetores serão considerados como sendo novos.
- **Etapa 3:** excluir aqueles "vetores anormais" do conjunto de treinamento original e treinar novamente a rede com o novo (limpo) conjunto de dados.

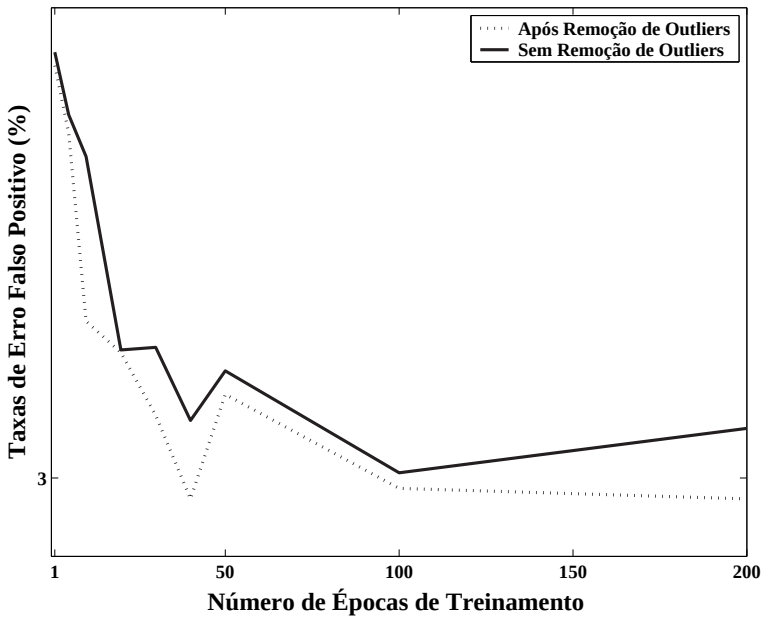


**Figura A.2:** Taxas médias de erro falso negativo (%) para o par (SOM, *boxplot*) treinado com o conjunto de dados original e com um conjunto de dados "limpos" variando o número de épocas de treinamento.

Para ilustrar o método proposto e a importância da remoção de *outliers*, fez-se uma simulação com o conjunto de dados de mamografia utilizado no Capítulo 5. Observou-se que todos os métodos de detecção de novidades apresentaram uma redução considerável em suas taxas de erro falso negativo após a aplicação do procedimento de limpeza dos dados proposto nesta dissertação, como mostrado na Figura A.2 para o par (SOM, *boxplot*). O número de neurônios foi ajustado em 40, e o número de épocas de treinamento variou de 1 a 200. Vale a pena notar que o treinamento usando um conjunto de dados "limpos" rendeu o melhor desempenho de um detector SOM, obtendo uma taxa de falsos negativos abaixo de 3%.

# *APÊNDICE B* – Simulação de Sistema Celular CDMA2000

Os dados utilizados no Capítulo 4 foram gerados por uma ferramenta de simulação baseada na tecnologia de transmissão de rádio CDMA200 1xRTT (PHYSICAL..., 2002). Em instantes diferentes de tempo, o simulador faz um exame instantâneo do sistema, de tal maneira que nenhuma correlação existe entre os parâmetros do sistema em instantes de tempo diferentes (BANKS et al., 2000; LAW; KELTON, 2000).

Supondo um grande número de serviços de dados e de voz, o simulador utilizado é capaz de executar análise da capacidade do sistema, indicando quantos usuários (assinantes) podem ser suportados em diferentes cenários de simulação. Para esta finalidade, faz-se um exame dos recursos de rádio disponíveis, tais como a interferência, os códigos de Walsh, a potência de transmissão e o ambiente de propagação. Em cada instante são geradas informações sobre o estado de sistema tais como condições de tráfego, carga suportada, interferência e *handoff*. Os parâmetros da entrada são divididos em cinco classes, de acordo com sua influência no comportamento do sistema.

Os usuários são distribuídos uniformemente sobre uma área retangular. Pode-se escolher o número de células interferentes de zero a três, que conduz a uma região que contem 1, 7, 19 ou 37 células respectivamente. Não há nenhuma correlação entre as posições do usuário em diferentes instantes de tempo. O modelo da propagação leva em conta as contribuições de três fenômenos diferentes: perda de percurso, desvanecimento rápido e sombreamento. O modelo de perda percurso é o Okumura–Hata clássico (HATA, 1980; OKUMURA et al., 1968), ligeiramente modificado. Supõe-se um desvanecimento rápido que segue a distribuição de Rayleigh. O sombreamento é composto por uma combinação da unidade móvel (UM) e da estação rádio-base (ERB), ambas são modeladas por distribuições do tipo log-normal.

O simulador fornece até cinco tipos diferentes de serviços aos usuários CDMA2000, com taxas de 9600, 19200, 38400, 76800 e 153600 *bps*. Cada serviço é caracterizado por uma demanda e uma aplicação específicas de tráfego. É também possível ajustar a quantidade de unidades móveis usando cada serviço.

Os cenários de simulação podem ser ajustados escolhendo alguns parâmetros da entrada que serão mantidos fixos durante todo um conjunto de simulações, quando outros parâmetros terão seus valores variados. Cada cenário corresponde a uma variação em um único parâmetro, a fim de tornar mais simples uma análise posterior. É possível especificar os cenários de tráfego nos quais os serviços de todos os usuários são apenas de voz ou nos quais há ambos os serviços: transmissão de voz e de dados.

Além da análise da carga, outros resultados podem fornecer uma compreensão melhor sobre o desempenho da rede CDMA2000, tal como a vazão média de dados (*throughput*) no enlace direto ou no reverso, no nível total de ruído no sistema (*Noise Rise*), na análise de interferência intra/inter-células e nos mapas de cobertura. Além disso, estes resultados podem ser usados para derivar, de forma direta, a taxa de erro de *bit* (BER – *Bit Error Rate*) e a taxa de erro de frame (FER – *Frame Error Rate*). Os resultados gerados pelo simulador são estatisticamente compatíveis com dados reais de redes celulares.

## B.1 Ajuste de Parâmetros do Simulador

Para a geração de dados, um conjunto de parâmetros principais e secundários devem ser ajustados. As especificações técnicas referentes aos diferentes parâmetros de simulação podem ser encontrados em (PHYSICAL..., 2002).

Os parâmetros são divididos em quatro categorias principais:

- parâmetros gerais;
- parâmetros das ERBs;
- parâmetros do canal de transmissão;
- parâmetros das unidades móveis.

Um ajuste usual para estes parâmetros nas simulações realizadas é mostrada a seguir.

- **PARÂMETROS GERAIS**
  - Raio celular = 2 km
  - Quantidade de Unidades Móveis = 60
  - Quantidade de ERBs = 7
  - Iterações do Controle de Potência = 50
  - Total de rodadas independentes = 500

- Nível de ruído máximo = 6 $dB$
- Limiar de $\frac{E_c}{I_o}(T_ADD)$ = -13 $dB$
- Tipo de célula = onidirecional
- Cofiguração de rádio = RC3

- **PARÂMETROS DAS ERBs**
  - Potência maxima da ERB = 20 W (os parâmetros percentuais seguintes são relativos a este)
  - Potência do canal piloto = 15%
  - Potência do canal de tráfego = 5%
  - Potência dos canais de controle = 9%
  - Figura de ruído do receptor da ERB = 5 $dB$
  - Perda nos cabos e conectores da ERB = 2 $dB$
  - Ganho de diversidade de recepção = 1 $dB$

- **PARÂMETROS DO CANAL DE TRANSMISSÃO**
  - *Fast Fading* = Habilitado
  - *Shadow Fading* = Habilitado
  - Desvio-Padrão do *Shadow Fading* = 8 $dB$
  - Distância de Descorrelação devido a proximidade geográfica = 0,02 Km
  - Modelo de propagação = Okumura-Hata
  - Freqüência da portadora = 850 $MHz$
  - Altura da antena da ERB em relação ao solo = 30 m

- **PARÂMETROS DAS UNIDADES MÓVEIS - UM**
  - Potência maxima de transmissao da UM = 630,9573 mW
  - Figura de ruido do receptor da UM = 9 $dB$
  - Ganho da antena de transmissao da UM = 0 $dBi$
  - Perda nos cabos e conectores da UM = 1 $dB$

# *Referências*

ADROVER, J. Minimax bias-robust estimation of the dispersion matrix of multivariate distributions. *Annals of Statistics*, v. 26, p. 2301–2320, 1998.

AHALT, S. et al. Competitive learning algorithms for vector quantization. *Neural Newtowks*, v. 3, n. 3, p. 277–290, 1990.

ATKINSON, A. Fast very robust methods for the detection of multiple outliers. *Journal of the American Statistical Association*, v. 89, p. 1329–1339, 1994.

AUGUSTEIJN, M. F.; FOLKERT, B. A. Neural network classification and novelty detection. *International Journal of Remote Sensing*, v. 23, n. 14, p. 2891–2902, 2002.

BANKS, J. et al. *Discrete-Event System Simulation*. 3rd. ed. Upper Saddle River, NJ: Prentice-Hall, 2000.

BARRETO, G. A. et al. Competitive neural networks for fault detection and diagnosis in 3G cellular systems. *Lecture Notes in Computer Science*, v. 3124, p. 207–313, 2004.

BHAYA, A.; KASZKUREWICZ, E. Steepest descent with momentum for quadratic functions is a version of the conjugate gradient method. *Neural Networks*, v. 17, n. 1, p. 65–71, 2004.

BLAKE, C. L.; MERZ, C. J. UCI repository of machine learning databases. University of California, Irvine, Dept. of Information and Computer Sciences, http://www.ics.uci.edu/∼mlearn/MLRepository.html. 1998.

CARPENTER, G. A.; RUBIN, M.; STREILEIN, W. W. ARTMAP-FD: familiarity discrimination applied to radar target recognition. In: *Proceedings of the IEEE International Conference on Neural Networks*. [S.l.: s.n.], 1997. v. 3, p. 1459–1464.

DAO, V.; VEMURI, V. Computer network intrusion detection: A comparison of neural networks methods. *Differential Equations and Dynamical Systems, (Special Issue on Neural Networks, Part-2)*, v. 10, n. 1 & 2, 2002.

DAWSON, M. R. W.; SCHOPFLOCHER, D. P. Modifying the generalized delta rule to train networks of nonmonotonic processors for pattern classification. *Connection Science*, v. 4, n. 1, p. 19–31, 1992.

DiCICCIO, T. J.; EFRON, B. Bootstrap confidence intervals. *Statistical Science*, v. 11, n. 3, p. 189–228, 1996.

DUTRA, A. et al. c-erbB-2 expression and nuclear pleomorphism in canine mammary tumors. *Brazilian Journal of Medical and Biological Research*, v. 37, n. 11, p. 1673–1681, 2004.

EFRON, B. Bootstrap methods: Another look at jackknife. *Ann. Statist.*, v. 7, p. 1–26, 1979.

EFRON, B.; TIBSHIRANI, R. J. *An Introduction to the Bootstrap*. [S.l.]: Chapman & Hall, 1993.

GONZALEZ, F.; DASGUPTA, D. Neuro-immune and self-organizing map approaches to anomaly detection: A comparison. In: *Proceedings of the First International Conference on Artificial Immune Systems*. Canterbury, UK: [s.n.], 2002. p. 203–211.

GRAY, R. M. Vector quantization. *IEEE Acoustics, Speech, and Signal Processing Magazine*, v. 1, n. 2, p. 4–29, 1984.

GUH, R. S. et al. On-line control chart pattern detection and discrimination: A neural network approach. *Artificial Intelligence in Engineering*, v. 13, n. 4, p. 413–425, 1999.

HARDIN, J.; ROCKE, D. Outlier detection in the multiple cluster setting using the minimum covariance determinant estimator. *Computational Statistics and Data Analysis*, v. 44, p. 625–638, 2004.

HARRIS, T. A Kohonen SOM based machine health monitoring system which enables diagnosis of faults not seen in the training set. In: *Proceedings of the International Joint Conference on Neural Networks, (IJCNN'93)*. [S.l.: s.n.], 1993. v. 1, p. 947–950.

HATA, M. Empirical formula for propagation loss in land mobile radio services. *IEEE Transactions on Vehicular Technology*, v. 29, n. 3, p. 317–325, 1980.

HAWKINS, D.; BRADU, D.; KASS, G. Location of several outliers in multiple regression data using elemental sets. *Technometrics*, v. 26, p. 197–208, 1984.

HAYKIN, S. *Neural Networks: A Comprehensive Foundation*. 2nd. ed. [S.l.]: Prentice-Hall, 1999.

HERTZ, J.; KROGH, A.; PALMER, R. G. *Introduction to the theory of neural computation*. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 1991.

HODGE, V. J.; AUSTIN, J. A survey of outlier detection methodologies. *Artificial Intelligence Review*, v. 22, n. 2, p. 85–126, 2004.

HÖGLUND, A. J.; HÄTÖNEN, K.; SORVARI, A. S. A computer host-based user anomaly detection system using the self-organizing map. In: *Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks (IJCNN'00)*. Como, Italy: [s.n.], 2000. v. 5, p. 411–416.

HOWARD, A. *Álgebra Linear com Aplicações*. 8a. ed. [S.l.]: Editora Companhia Bookman, 2001.

JAPKOWICZ, N.; MYERS, C.; GLUCK, M. A novelty detection approach to classification. In: *Proceedings of the 14th International Joint Conference on Artificial Intelligence (IJCAI'95)*. [S.l.: s.n.], 1995. p. 518–523.

KNORR, E.; NG, R. T.; TUCAKOV, V. Distance-based outliers: Algorithms and applications. *VLDB Journal*, v. 8, n. 3, p. 237–253, 2000.

KOHONEN, T. *Self-Organization and Associative Memory*. 3rd. ed. [S.l.]: Springer-Verlag, 1989.

KOHONEN, T. *Self-Organizing Maps*. 3rd. ed. [S.l.]: Springer-Verlag, 2001.

KOHONEN, T.; OJA, E. Fast adaptive formation of orthogonalizing filters and associative memory in recurrent networks of neuron-like elements. *Biological Cybernetics*, v. 25, p. 85–95, 1976.

KOSKO, B. *Neural Networks and Fuzzy Sytems: A Dynamical Systems Approach to Machine Intelligence*. [S.l.]: Prentice-Hall, 1992.

LAIHO, J.; KYLVäJä, M.; HöGLUND, A. Utilisation of advanced analysis methods in UMTS networks. In: *Proceedings of the IEEE Vehicular Technology Conference (VTS/spring)*. Birmingham, Alabama: [s.n.], 2002. p. 726–730.

LAIHO, J. et al. Advanced analysis methods for 3G cellular networks. *IEEE Transactions on Wireless Communications*, v. 4, n. 3, p. 930–942, 2005.

LAIHO, J. et al. Verification of WCDMA radio nework planning prediction methods with fully dynamic network simulator. In: *Proceedings of the IEEE Vehicular Technology Conference (VTS/fall)*. [S.l.: s.n.], 2001. v. 1, p. 526–530.

LAW, A. M.; KELTON, W. D. *Simulation Modeling and Analysis*. 3rd. ed. New York: McGraw-Hill, 2000.

LAWRENCE, S. et al. Neural network classification and unequal prior class probabilities. In: ORR, G.; Müller, K.-R.; CARUANA, R. (Ed.). *Neural Networks: Tricks of the Trade*. [S.l.]: Springer Verlag, 1998, (Lecture Notes in Computer Science, v. 1524). p. 299–314.

Le Cun, Y. et al. Handwritten digit recognition with a back-propagation network. In: TOURETZKY, D. S. (Ed.). *Advances in Neural Information Processing Systems*. [S.l.]: Morgan Kaufmann, 1990. v. 2, p. 396–404.

LI, Y.; PONT, M. J.; JONES, N. B. Improving the performance of radial basis function classifiers in condition monitoring and fault diagnosis applications where 'unknown' faults may occur. *Pattern Recognition Letters*, v. 23, n. 5, p. 569–577, 2002.

MALSBURG, C. v. Self-organization in the brain. In: ARBIB, M. (Ed.). *The Handbook of Brain Theory and Neural Networks*. 2nd. ed. Cambridge, MA: MIT Press, 2003. p. 1002–1005.

MARKOU, M.; SINGH, S. Novelty detection: A review – Part 1: Statistical approaches. *Signal Processing*, v. 83, n. 12, p. 2481–2497, 2003.

MARKOU, M.; SINGH, S. Novelty detection: A review – Part 2: Neural network based approaches. *Signal Processing*, v. 83, n. 12, p. 2499–2521, 2003.

MARSLAND, S. Novelty detection in learning systems. *Neural Computing Surveys*, v. 3, p. 157–195, 2003.

MARSLAND, S.; NEHMZOW, U.; SHAPIRO, J. Novelty detection for robot neotaxis. In: *Proceedings of the 2nd International ICSC Symposium on Neural Computation*. [S.l.: s.n.], 2000. p. 554–559.

MARSLAND, S.; SHAPIRO, J.; NEHMZOW, U. A self-organising network that grows when required. *Neural Networks*, v. 15, n. 8–9, p. 1041–1058, 2002.

MARTINETZ, T. M.; SCHULTEN, K. J. A 'neural-gas' network learns topologies. *Artificial Neural Networks*, Amsterdan, p. 397–402, 1991.

MUÑOZ, A.; MURUZáBAL, J. Self-organising maps for outlier detection. *Neurocomputing*, v. 18, p. 33–60, 1998.

OKUMURA, T.; OHMORI, E.; FUKUDA, K. Field strength and its variability in VHF and UHF land mobile service. *Review of the Electrical Communications Laboratory*, v. 16, n. 9–10, p. 825–873, 1968.

PETSCHE, T. et al. A neural network autoassociator for induction motor failure prediction. In: TOURETZKY, D.; MOZER, M.; HASSELMO, M. (Ed.). *Advances in Neural Information Processing Systems*. [S.l.]: MIT Press, 1996. v. 8, p. 924–930.

PHYSICAL Layer Standard for CDMA2000 Spread Spectrum Systems. [S.l.]: 3GPP2, 2002. Release C.

PRINCIPE, J. C.; EULIANO, N. R.; LEFEBVRE, W. C. *Neural and Adaptive Systems: Fundamentals through Simulations*. [S.l.]: John Wiley & Sons, 2000.

RAIVIO, K.; SIMULA, O.; LAIHO, J. Neural analysis of mobile radio access network. In: *Proceedings of the IEEE International Conference on Data Mining (ICDM)*. San Jose, California: [s.n.], 2001. p. 457–464.

REICH, Y.; BARAI, S. V. Evaluating machine learning models for engineering problems. *Artificial Intelligence in Engineering*, v. 13, p. 257–272, 1999.

ROCKE, D.; WOODRUFF, D. Identification of outliers in multivariate data. *Journal of the American Statistical Association*, v. 91, p. 1047–1061, 1996.

ROSE, C. J.; TAYLOR, C. J. A generative statistical model of mammographic appearance. In: RUECKERT, D.; YANG, J. H. andG. Z. (Ed.). *Proceedings of the 2004 Medical Image Understanding and Analysis (MUIA'04)*. [S.l.: s.n.], 2004. p. 89–92.

ROUSSEEUW, P.; LEROY, A. *Robust Regression and Outlier Detection*. 3rd. ed. [S.l.]: John Wiley & Sons, 1996.

SAMMON JR., J. A nonlinear mapping for data structure analysis. *IEEE Transactions on Computers*, C-18, p. 401–409, 1969.

SCHOLKOPF, B. et al. Support vector method for novelty detection. In: SOLLA, S. A.; LEEN, T. K.; MÜLLER, K.-R. (Ed.). *Advances in Neural Information Processing Systems*. Cambridge, MA: MIT Press, 2000. v. 12, p. 582–588.

SINGH, S.; MARKOU, M. An approach to novelty detection applied to the classification of image regions. *IEEE Transactions on Knowledge and Data Engineering*, v. 16, n. 4, p. 396–407, 2004.

SPIEGEL, M. R. *Estatística*. 2a.. ed. Rio de Janeiro – RJ: McGraw-Hill, 1984.

TANAKA, M. et al. Application of Kohonen's self-organizing network to the diagnosis system for rotating machinery. In: *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics (SMC'95)*. [S.l.: s.n.], 1995. v. 5, p. 4039–4044.

TAX, D.; DUIN, R. Outlier detection using classifier instability. In: Amin, A. et al. (Ed.). *Advances in Pattern Recognition, Lecture notes in Computer Science*. Berlin: Springer, 1998. v. 1451, p. 593–601.

TRIOLA, M. F. *Introdução à Estatística*. 7a.. ed. Rio de Janeiro – RJ: Editora LTC, 1999.

TUKEY, J. *Exploratory Data Analysis*. [S.l.]: Addison-Wesley, 1977.

VASCONCELOS, G. C.; FAIRHURST, M. C.; BISSET, D. L. Investigating feedforward neural networks with respect to the rejection of spurious patterns. *Pattern Recognition Letters*, v. 16, p. 207–212, 1995.

WEBB, A. *Statistical Pattern Recognition*. 2nd. ed. [S.l.]: John Wiley & Sons, 2002.

WOLBERG, W. H.; MANGASARIAN, O. L. Multisurface method of pattern separation for medical diagnosis applied to breast cytology. *Proceedings of the National Academy of Sciences (U.S.A.)*, v. 87, p. 9193–9196, 1990.

WORDEN, K.; MANSON, G.; FIELLER, N. Damage detection using outlier analysis. *Journal of Sound and Vibration*, v. 229, n. 3, p. 647–667, 2000.

YPMA, A.; DUIN, R. P. W. Novelty detection using self-organising maps. In: KASABOV, N. et al. (Ed.). *Progress in Connectionist-Based Information Systems*. [S.l.]: Springer-Verlag, 1997. v. 2, p. 1322–1325.

ZHANG, Z. et al. HIDE: A hierarchical network intrusion detection system using statistical preprocessing and neural network classification. In: *Proceedings of the IEEE Workshop on Information Assurance and Security*. [S.l.: s.n.], 2001. p. 85–90.