Universidade Federal do Rio Grande do Norte
Centro de Tecnologia
Programa de Pós-Graduação em
Engenharia Elétrica e Computação

Uma Arquitetura para Análise de Agrupamentos sobre Bases de Dados Distribuídas

Flavius da Luz e Gorgônio

Orientador: Prof. DSc. José Alfredo Ferreira Costa

Tese de Doutorado apresentada à coordenação do programa de Pós-Graduação em Engenharia Elétrica e Computação da Universidade Federal do Rio Grande do Norte como parte dos requisitos necessários para obtenção do grau de Doutor em Ciências no domínio da Engenharia Elétrica

Área de Concentração:
Redes Neurais / Processamento Inteligente da Informação

Natal / RN
Março de 2009

Universidade Federal do Rio Grande do Norte - UFRN
Sistema de Bibliotecas - SISBI
Catalogação de Publicação na Fonte. UFRN - Biblioteca Central Zila Mamede

Gorgônio, Flavius da Luz e.
Uma arquitetura para análise de agrupamentos sobre bases de dados distribuídas / Flavius da Luz e Gorgonio. - 2019.
155f.: il.

Tese (Doutorado)-Universidade Federal do Rio Grande do Norte, Centro de Tecnologia, Programa de Pós-Graduação em Engenharia Elétrica e Computação, Natal, 2019.
Orientador: Dr. José Alfredo Ferreira Costa.

1. Análise de agrupamentos distribuída - Tese. 2. Comitês de agrupamento - Tese. 3. k-Médias - Tese. 4. Mapas auto-organizáveis - Tese. I. Costa, José Alfredo Ferreira. II. Título.

RN/UF/BCZM CDU 004.65

Elaborado por Raimundo Muniz de Oliveira - CRB-15/429

Uma Arquitetura para Análise de Agrupamentos sobre Bases de Dados Distribuídas

Flavius da Luz e Gorgônio

Tese de Doutorado aprovada em 06 de Março de 2009 pela banca examinadora composta pelos seguintes membros:

Prof. DSc. Guilherme de Alencar Barreto (avaliador externo)	DETI/UFC
Prof. PhD. Paulo Jorge Leitão Adeodato (avaliador externo)	CIN/UFPE
Profª. DSc. Anne Magaly de Paula Canuto	DIMAP/UFRN
Prof. DSc. Allan de Medeiros Martins	DEE/UFRN
Prof. DSc. José Alfredo Ferreira Costa (orientador)	DEE/UFRN

À minha esposa e filhos,
pela compreensão nos momentos
de ausência.

‘‘When you use information from one source,
it’s plagiarism;
When you use information from many,
it’s information fusion’’

Belur V. Dasarathy

Agradecimentos

À minha esposa, *Maria Lionete Costa Gorgônio*, pelo apoio constante em todas as etapas do meu trabalho, sem a ajuda da qual, eu não haveria conseguido atingir meus objetivos. Aos meus filhos, *Arthur*, *Júlia* e *Eduardo*, que mesmo sem a compreensão dos motivos, tiveram que suportar tantos momentos de ausência em uma das fases mais importantes de suas vidas.

Aos meus pais, *Flávio de Araújo Gorgônio* e *Dulce Helena da Luz Gorgônio* (*in memorian*) e avós paternos e maternos, pelos inúmeros ensinamentos e lições de vida que me foram transmitidos.

Ao meu orientador, *Prof. Dr. José Alfredo Ferreira Costa*, tanto pelo constante incentivo que me foi dado, quanto pelas cobranças quando foram necessárias e, principalmente, pela paciência e pela confiança em mim depositada durante todo o período da pesquisa.

À *Prof^{a}. Dr^{a}. Anne Magaly de Paula Canuto* (DIMAP/UFRN), pelas críticas e valorosas contribuições dadas ao trabalho, tanto durante a etapa de qualificação quanto na fase final de escrita da tese.

Aos colegas do Laboratório de Sistemas Adaptativos, *Sérgio Santos*, *Gustavo Souza*, *Sheyla Farias*, *Cláudio Medeiros*, *Carlos Heitor*, *Raquel Escarcina*, *Jakeline Silva*, *Bruno Andrade* e *Ricardo Oliveira*, pelo companheirismo durante o período em que passamos juntos.

À *UFRN* e à *CAPES*, pela oportunidade de realizar o curso e pelo suporte financeiro, através do financiamento da minha pesquisa.

Resumo

Mineração de dados pode ser definida como um conjunto de técnicas para a extração de conhecimento e procura de padrões úteis e previamente desconhecidos em grandes volumes de dados multidimensionais. Algoritmos tradicionais de análise de agrupamento, como mapas auto-organizáveis e K-médias têm sido largamente utilizados como ferramentas de mineração de dados, com o objetivo de permitir a visualização de dados de elevada dimensionalidade, auxiliando na identificação de agrupamentos de dados com características semelhantes. No entanto, algoritmos tradicionais para análise de dados podem não ser eficientes para algumas aplicações atuais por não considerarem a existência de dados armazenados de forma distribuída. Assim, uma crescente tendência de minerar dados armazenados de forma distribuída tem motivado o surgimento de métodos que permitem analisar cada uma das bases de dados isoladamente e combinar os resultados parciais para obter um resultado final. Este trabalho apresenta uma arquitetura para análise de agrupamentos em bases de dados distribuídas, a partir da utilização de algoritmos tradicionais, que reduz sensivelmente a quantidade de dados transferidos entre as unidades remotas e a unidade central. A arquitetura é composta por uma estratégia, baseada em quantização vetorial, que possibilita extrair um conjunto de representantes a partir de partições horizontais e/ou verticais da base de dados, a fim de se obter visões parciais dos agrupamentos existentes em cada um dos conjuntos de dados locais. Posteriormente, os representantes de cada unidade local são enviados à unidade central, que efetua a combinação dos resultados parciais através de um processo de agrupamento sobre os representantes dos dados. Os resultados experimentais obtidos com a utilização da arquitetura proposta sobre diferentes conjuntos de dados demonstram que essa estratégia consegue obter resultados com mesma eficácia que os obtidos com as técnicas de mineração de dados convencionais, onde todas as bases de dados são transferidas para uma unidade central durante a etapa de pré-processamento.

Palavras-chaves: análise de agrupamentos distribuída; comitês de agrupamento; *k*-médias; mapas auto-organizáveis.

Abstract

Data mining can be defined as a set of techniques for knowledge extraction and search of useful and previously unknown patterns in large multidimensional databases. Clustering is the process of discovering data clusters within high-dimensional databases, based on similarities, with a minimal knowledge of their structure. Distributed data clustering is a recent approach to deal with distributed databases, since traditional clustering algorithms require centering all databases in a single dataset. Moreover, current privacy requirements in distributed databases demand algorithms with the ability to process clustering securely. Thus, an increasing need of methods to mining data stored in a distributed way has motivated the development of algorithms to analyze each database separately and to combine the partial results to get a final result. This thesis presents a framework for cluster analysis in distributed databases using traditional algorithms, as K-means and self-organizing maps. This approach reduces significantly the amount of data transferred between remote units and the central unit. The framework includes a strategy, based on vectorial quantization, that extract a representatives subset, in order to get partial views of the existing clusters in each horizontal and/or vertical partitions of the database. Later, the representatives of each local unit are sent to the central unit, which carry out a combination of the partial results applying a clustering algorithm over all representative subsets. The experimental results with different datasets show that the framework proposed obtains results very close and with effectiveness comparable to conventional data mining techniques, where all the databases are transferred to a central unit in the pre-processing stage.

Keywords: distributed data clustering, cluster ensembles; *k*-means; self-organizing maps.

Sumário

Lista de Figuras

Lista de Tabelas

Lista de Símbolos e Abreviaturas

bmu	*best match unit*, unidade de uma rede neural que mais se assemelha ao elemento do conjunto de entrada que está sendo apresentado
codebook	conjunto de centróides que representam um conjunto de dados
codeword	elemento de um *codebook*
KDD	*Knowledge Discovery in Databases*, ou descoberta de conhecimento em banco de dados
qe	*quantization error*, erro de quantização médio
SOM	*self-organizing map*, ou mapa auto-organizável
te	*topological error*, erro topológico de um mapa auto-organizável
σ	desvio padrão associado a um conjunto de valores medidos em um experimento
$\sigma_{inicial}$	valor inicial do raio de vizinhança a ser atualizado em um SOM
σ_{final}	valor final do raio de vizinhança a ser atualizado em um SOM
C_i	i-ésima partição de um conjunto de dados dada por $C = \{C_1, C_2, ..., C_k\}$
$\boldsymbol{x}_i$	i-ésimo elemento de um conjunto de dados dado por $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$
$\boldsymbol{w}_i$	i-ésimo elemento de um *codebook* dado por $\mathbf{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k\}$
d_{ij}	$d(\boldsymbol{x}_i, \boldsymbol{x}_j)$, distância do elemento i ao elemento j
N, n	cardinalidade (número de elementos) do conjunto de entrada
p	dimensionalidade (número de atributos) do conjunto de entrada
k	número de classes ou de agrupamentos de um conjunto
UCI	*University of California, Irvine*, mantenedora de um repositório de bases de dados para aprendizado de máquina

Capítulo 1
Introdução

A história da humanidade sempre foi marcada por constantes mudanças. Heráclito (540 – 470 a.C.), filósofo grego, universalizou esse conceito ao afirmar que "*tudo muda, exceto a própria mudança*". Entretanto, em tempos recentes, alguns setores como tecnologia e negócios têm enfrentado transformações cada vez mais radicais e em um ritmo muito mais intenso.

A globalização da economia, os avanços tecnológicos e a popularização dos computadores e da Internet são algumas das modificações ocorridas no cenário econômico mundial que alteraram definitivamente a forma de atuação das empresas no mundo dos negócios. Tais mudanças encurtaram distâncias, acirraram a concorrência, colocaram frente a frente empresas grandes e pequenas, reais e virtuais, locais e remotas.

Como em qualquer revolução, a denominada *revolução da informação* trouxe consigo uma série de consequências. Talvez a mais óbvia delas seja a crescente necessidade de se coletar, manter e processar um volume cada vez maior de dados dentro das organizações. Com isso, há uma crescente demanda por mecanismos que possam auxiliar na automatização desses processos. Ao conjunto desses mecanismos, incluindo *hardware*, *software* e outros dispositivos de apoio ao processamento da informação, denomina-se Tecnologia da Informação (TI).

Esse capítulo busca mostrar a importância e necessidade das empresas terem que lidar com grandes volumes de dados como ferramenta de apoio em seus processos de tomadas de decisão. É apresentado um resgate histórico dos mecanismos administrativos utilizados desde a revolução industrial até os dias atuais, concluindo com um conjunto de perspectivas e necessidades que motivaram esta pesquisa. São apresentados, ainda, a problemática abordada, os objetivos da pesquisa e as principais contribuições obtidas pelo trabalho.

1.1 Contexto Histórico

Após a revolução industrial, iniciada em meados do século XVIII, a tecnologia passou a influenciar de forma determinante o funcionamento da sociedade em todo o mundo [Chiavenato, 2004]. Assim como a produção manual e artesanal deu lugar às máquinas e motores, o surgimento de novos meios de comunicação e transporte provocou

mudanças com grande impacto no processo produtivo nos níveis político, econômico e social.

A revolução industrial foi o ponto de partida para o surgimento de empresas e organizações e para o desenvolvimento de diversos aparatos tecnológicos, tais como a máquina de escrever, o telefone e a calculadora. Entretanto, foi o surgimento do computador digital que possibilitou às empresas a automação e a automatização em larga escala de suas atividades, sem o qual não seria possível gerenciar a diversidade de produtos, processos, serviços, clientes, fornecedores e pessoal existentes na atualidade [Chiavenato, 2004].

Apesar da criação do computador ter sido por volta de 1945, coincidindo com fim da Segunda Guerra Mundial, esses equipamentos levaram algum tempo para chegar às empresas. Os primeiros *mainframes*, que vieram logo em seguida, eram equipamentos extremamente caros e frágeis, além de exigirem enorme esforço para programação e uso, sendo utilizados em escala comercial somente em instituições financeiras e empresas de grande porte.

O surgimento de novas linguagens de programação como o FORTRAN e o COBOL, mais fáceis de utilizar que a programação usual da época, em código de máquina [Sebesta, 2003], e os avanços na área de *hardware*, com a invenção dos transistores e circuitos integrados foram decisivos na criação de máquinas menores e mais robustas [Brookshear, 2000]. Porém, somente com o surgimento do computador pessoal, ao final da década de 1970, esses equipamentos chegaram definitivamente ao mercado.

Antes da popularização da Informática, as empresas controlavam o seu fluxo de dados e informações através de processos burocráticos, apoiadas por métodos e técnicas provenientes de diversas áreas, como a Teoria Matemática da Informação, a Teoria dos Jogos, a Teoria Geral dos Sistemas e a Cibernética [Chiavenato, 2004]. Nesse período, as empresas se utilizavam de técnicas de arquivamento e recuperação de informações em grandes arquivos manuais, onde era necessário um profissional especializado para organizar os dados, registrá-los, classificá-los, catalogá-los e recuperá-los quando necessário [Dias, 2006].

Além de exigir um grande esforço humano para manter os dados atualizados e organizados, nem sempre estes métodos eram eficientes na recuperação das informações necessárias à tomada de decisões em um tempo hábil. Assim, a crescente necessidade das organizações em processar suas informações, à medida que os seus volumes de dados aumentavam, impulsionou o desenvolvimento do computador digital, dos aplicativos comerciais e de muitas das ferramentas de automação comercial disponíveis nos dias atuais.

Atualmente, os computadores estão tão interligados ao mundo dos negócios que é praticamente impossível a sua dissociação. Com os computadores, as empresas puderam lidar com um volume cada vez maior de informações. Entretanto, foi o surgimento da

Internet que impulsionou definitivamente a sociedade para uma nova revolução, a tão comentada *revolução da informação*.

A principal diferença no mundo pós-Internet é a velocidade com que dados e informações circulam de um lado a outro do planeta. Nunca houve tanta troca de informação em tão pouco tempo. Isso acontece principalmente porque, diferente de outros bens, a informação não possui nenhum valor quando é armazenada. O valor da informação não está em mantê-la, mas em utilizá-la de forma estratégica [Fayyad *et al*., 1996b].

Dentro deste contexto, a informação passou a ter um valor gerencial muito importante, sendo considerada como um fator determinante para a obtenção do sucesso. Moresi (2000) afirma que, dentro de uma sociedade globalizada como a atual, a informação não é apenas mais um recurso, mas o recurso-chave de competitividade efetiva, diferencial de mercado e lucratividade nessa nova sociedade. O capital financeiro, mesmo mantendo sua importância relativa no mundo dos negócios, é cada vez mais dependente do capital intelectual, baseado no conhecimento [Chiavenato, 2004].

No entanto, mesmo com a presença cada vez maior dos computadores nos mais diversos setores das empresas atuais, nem sempre o nível de informatização dessas empresas é proporcional aos investimentos efetuados. Uma falha comum, ainda existente em muitas empresas, é encontrar administradores que investem demasiadamente em *hardware*, mas economizam em *software* e capacitação de pessoal. Se por um lado essas empresas possuem tecnologia necessária para armazenar e processar os dados, por outro lado esse potencial é extremamente desperdiçado por não haver bons *softwares* disponíveis ou por não possuírem pessoal devidamente capacitado a operá-los de forma adequada.

Portanto, um dos maiores desafios para os profissionais de TI atuais continua sendo convencer empresários e administradores a aceitar que a informação possui o mesmo valor que outros recursos da empresa e que os investimentos na área de TI, particularmente em *software* e treinamento de pessoal, devem ter o mesmo grau de importância que os investimentos em máquinas, equipamentos e instalações físicas.

1.2 Os Sistemas de Informação

A tecnologia é um recurso de fundamental importância em um mercado extremamente competitivo como o atual, em que se buscam alternativas que possam se transformar em vantagens perante a concorrência. O diferencial competitivo de uma empresa depende da sua capacidade de oferecer soluções para as demandas da sociedade [Garcia *et al*., 2003]. Atualmente, a forma mais eficiente de se conhecer estas demandas é utilizando os recursos da tecnologia da informação.

De acordo com Mañas (2005), um sistema de informação (SI) é um conjunto formado pelas pessoas, estrutura organizacional, *hardware*, *software*, procedimentos e métodos que permitem à empresa dispor das informações que necessita, no tempo

desejado, tanto para o seu funcionamento atual, quanto para a sua evolução na busca de novos mercados.

Os sistemas de informação são responsáveis pela coleta dos dados de entrada, pelo armazenamento desses dados em dispositivos adequados e pela sua manipulação e posterior disseminação, seja na forma bruta (dados) ou devidamente processados (informação), a fim de atender a uma necessidade específica.

O primeiro passo para se obter êxito em um processo de informatização é a implantação de um sistema de informação que possa gerenciar o fluxo de dados e informações dentro da empresa. Os sistemas de informação permitem consolidar os dados colhidos no nível operacional, apresentando-os na forma de planilhas, relatórios e gráficos, visando atender às necessidades dos níveis gerenciais. Entretanto, sistemas de informação são incapazes de produzir conhecimento, o que sugere a demanda de ferramentas específicas para esta finalidade.

1.3 Dados, Informação e Conhecimento

Toda a base para a tomada de decisão dentro das organizações inicia-se no nível operacional, através da coleta e armazenamento dos dados referentes às transações realizadas. Entretanto, o simples armazenamento dos dados não garante sua utilidade para a empresa, sendo necessário que estes sejam processados e analisados a fim de contribuírem em processos de tomada de decisão.

Alguns autores defendem que os níveis superiores de uma empresa necessitam de informações qualitativas, com alto valor agregado, que permitam ter uma visão global da situação, enquanto que os níveis inferiores necessitam de informação quantitativa, de baixo valor agregado, que possibilitam o desempenho de atividades corriqueiras [Nakagawa, 1987] [Moresi, 2000]. Entretanto, embora essa visão simplista e segmentada possa parecer adequada ao desempenho das tarefas administrativas mais comuns, ela é cada vez menos adequada aos dias atuais. Independentemente da posição hierárquica ou do cargo que ocupa, os executivos necessitam de informações que ofereçam suporte à tomada de decisão, porém essas demandas podem ser bastante diferentes em função do contexto em que são necessárias.

À medida que flui dentro das empresas e organizações, a informação passa por vários níveis hierárquicos. De acordo com o porte da empresa ou organização e do seu nível de desenvolvimento tecnológico, pode haver mais ou menos níveis dentro dessa hierarquia. Cada nível da hierarquia corresponde a diferentes necessidades e requer um conjunto de processamentos necessários para transformar os valores recebidos em informações úteis para a tomada de decisões.

Urdaneta (1992) divide a informação em quatro níveis hierárquicos: dados, informação, conhecimento e entendimento. Em uma classificação mais recente, Rezende (2003) apresenta seis níveis: dados, informação, conhecimento, compreensão, análise e síntese, onde os três últimos são uma visão mais detalhada do último nível da classificação

proposta por Urdaneta. Entretanto, a maioria dos autores opta por uma classificação mais simples, baseada na existência de três níveis hierárquicos, mais comumente explorados dentro de uma organização: dados, informação e conhecimento [Setzer, 1999] [Valentim, 2002]. A Figura 1.1 apresenta a hierarquia existente entre dados, informação e conhecimento e os seus níveis de atuação dentro das organizações.

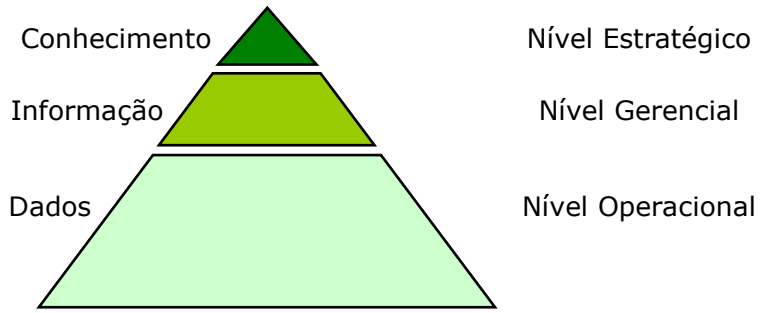


Figura 1.1: Dados, informação e conhecimento.

Um *dado* é um registro a respeito de um evento ou de uma ocorrência. Analisado de forma isolada, um dado tem pouco ou nenhum significado. Quando um conjunto de dados é combinado e analisado dentro de um contexto, é possível extrair *informações* acerca do seu significado. O *conhecimento* é a análise estratégica das informações obtidas, com a finalidade de auxiliar no processo de tomada de decisões.

Suponha a realização de uma pesquisa de mercado com a finalidade de medir a aceitação de um determinado produto em uma população. A idade, o sexo ou a renda do entrevistado são exemplos de *dados* obtidos na pesquisa. A tabulação dos dados e sua posterior apresentação na forma de tabelas e gráficos são exemplos de *informações* que podem ser obtidas a partir do processamento dos dados. É possível, por exemplo, constatar que a maioria dos entrevistados do sexo feminino respondeu SIM na questão 4 ou identificar a quantidade de homens, com idade entre 20 e 25 anos que ainda não conhecem o produto investigado.

Nesse contexto, o *conhecimento* poderia ser extraído a partir da aplicação de algoritmos de mineração de dados sobre os resultados da pesquisa, a partir dos quais se identificasse, supostamente, a existência de um subgrupo de entrevistados, formado por mulheres casadas, que trabalham fora e possuem idade entre 25 e 32 anos, cujo percentual de aceitação do produto é superior a 93%. Nesse caso, esse subgrupo poderia representar uma excelente oportunidade de mercado para a empresa e mereceria ser avaliado com mais atenção. Infelizmente, informações preciosas como esta, nem sempre são fáceis de serem obtidas e dificilmente estariam disponíveis através de um sistema de informação.

Como as organizações são dinâmicas e o processo de coleta e armazenamento de dados é constante, tanto o processamento das informações como o conhecimento extraído

durante o processo são voláteis, pois estão em constante mutação. Assim, a inclusão de novos registros ou até uma modificação efetuada na base de dados pode influenciar diretamente nos resultados obtidos durante a análise dos dados.

Um erro comum em processos de informatização é acreditar que o simples acúmulo de grandes volumes de dados a partir da implantação de um sistema de informação garantirá conhecimento para a empresa. Da mesma forma que um grande volume de dados precisa ser processado para produzir informação, é necessária a utilização de métodos e técnicas específicas para que a informação seja analisada e interpretada antes de se transformar em conhecimento, que constitui a matéria-prima para a tomada de decisões [Castro *et al*., 1999].

1.4 Descoberta de Conhecimento em Bases de Dados

A popularização dos computadores e dos sistemas de informação ocorrida nas últimas décadas foram alguns dos fatores que contribuíram para o aumento exponencial do volume de dados armazenados nas empresas nos últimos anos. Outros fatores incluem a facilidade e o baixo custo dos atuais dispositivos de armazenamento de dados e a ampla projeção que as redes de computadores, particularmente a Internet, obtiveram nos últimos anos, favorecendo a transferência dos dados entre computadores.

O acúmulo de grandes volumes de dados requer a utilização de métodos e técnicas eficientes que permitam processá-los. Analisar e visualizar grandes volumes de dados na forma de registros, descritos por vários atributos e armazenados em um banco de dados é uma tarefa não-trivial, tanto em função do grande número de registros normalmente existentes nesses bancos de dados, como pela grande quantidade de informações presentes em cada registro [Costa, 1999].

O uso de métodos estatísticos tem sido a maneira tradicionalmente utilizada para realizar essa tarefa. Até o início da década de 1990, os métodos tradicionais de análise e interpretação de dados utilizados na tentativa de extrair conhecimento eram baseados na seleção e aplicação de técnicas estatísticas guiados por um especialista [Fayyad *et al*., 1996a]. As facilidades decorrentes do armazenamento e transferência de dados dentro das organizações e a ampla difusão dos computadores nos mais variados setores das empresas motivou pesquisadores a procurar alternativas que permitissem processar esses dados de forma mais fácil, com a finalidade de obter alguma vantagem competitiva [Shaw *et al.*, 2001].

O processo de descoberta de conhecimento em bancos de dados (*Knowledge Discovery in Database* – KDD) pode ser definido como um processo de várias etapas, não trivial, interativo e iterativo, para identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados [Frawley *et al*., 1992] [Fayyad *et al*., 1996a]. Na Figura 1.2, adaptada de Fayyad *et al*. (1996a), estão ilustradas as principais etapas do processo de KDD, que serão descritas com mais detalhes no Capítulo 2.

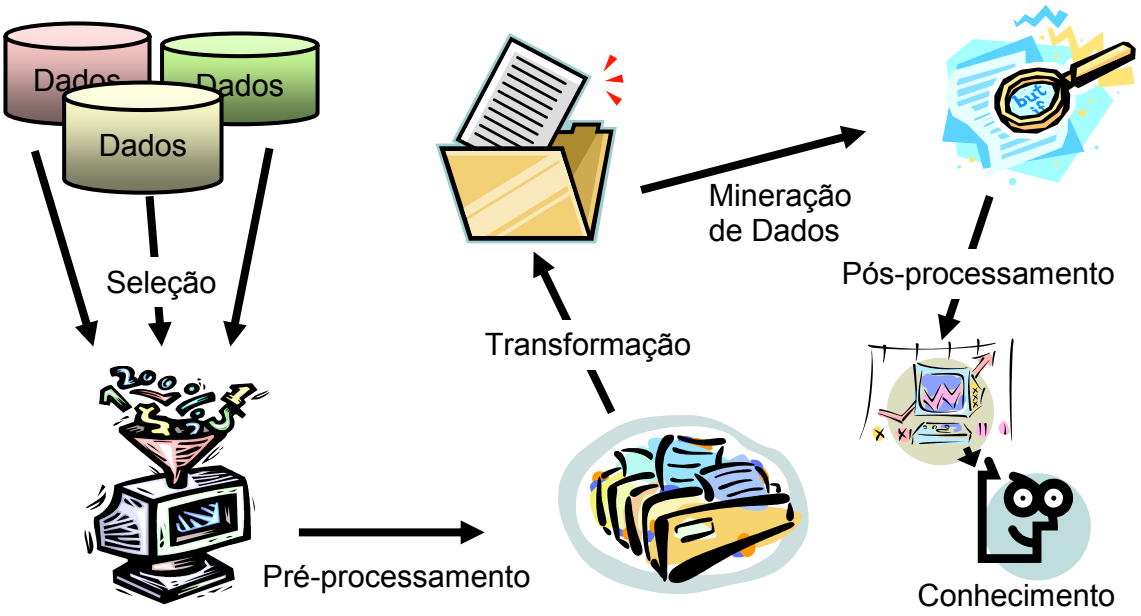


Figura 1.2: Ilustração do processo de KDD.

Por se tratar de um processo não trivial, KDD inclui um conjunto de métodos e técnicas que, uma vez combinados, possibilita extrair informações que na maioria das vezes estão ocultas em imensas quantidades de dados. A aplicação desses métodos e técnicas às bases de dados caracteriza uma das etapas do processo de KDD, conhecida como mineração de dados (*data mining*).

1.5 Mineração de Dados

Mineração de dados, apesar de ser muito confundida com o próprio processo de KDD, é definida como sendo uma das etapas desse processo, que consiste na aplicação de algoritmos de análise e descoberta de padrões sobre os dados pré-processados, a fim de produzir conhecimento [Goldschmidt e Passos, 2005]. A finalidade da etapa de mineração de dados é descobrir relações entre os dados que não são visíveis através da utilização de consultas tradicionais em SQL, nem com o auxílio de técnicas OLAP (*On-Line Analytical Processing*), ferramentas tradicionais da área de banco de dados [Fayyad *et al*., 1996b].

A mineração de dados pode ser vista como resultado de uma evolução natural da tecnologia da informação, em uma sequência que inclui o desenvolvimento de mecanismos de coleta e criação de bancos de dados (década de 1960), o desenvolvimento de sistemas de gerenciamento de bancos de dados – SGBD (década de 1970), a evolução dos modelos de representação de dados (década de 1980), o surgimento de mecanismos para armazenamento de grandes volumes de dados e para análise e compreensão dos

dados (década de 1990) e, finalmente, o aparecimento de uma nova geração de sistemas de informação (a partir do ano 2000) [Han e Kamber, 2006].

Os principais objetivos da etapa de mineração de dados são, basicamente, predição e descrição. A predição busca utilizar os dados disponíveis para criar um modelo que permita predizer sobre dados novos ou sobre variáveis não conhecidas, enquanto que a descrição busca explorar os dados disponíveis a fim de descobrir padrões que expliquem relações existentes entre os dados que não são facilmente percebíveis [Tan *et al*., 2006]. Cada um desses objetivos pode ser atingido de diversas maneiras, dependendo do resultado que se deseja obter. Por exemplo, em tarefas de descrição, uma das técnicas mais utilizadas é a análise de agrupamentos; enquanto que em tarefas de predição, a técnica de classificação obtém maior destaque. A Figura 1.3 divide a mineração de dados em duas tarefas: predição e descrição, apresentando as principais técnicas associadas a cada uma delas.

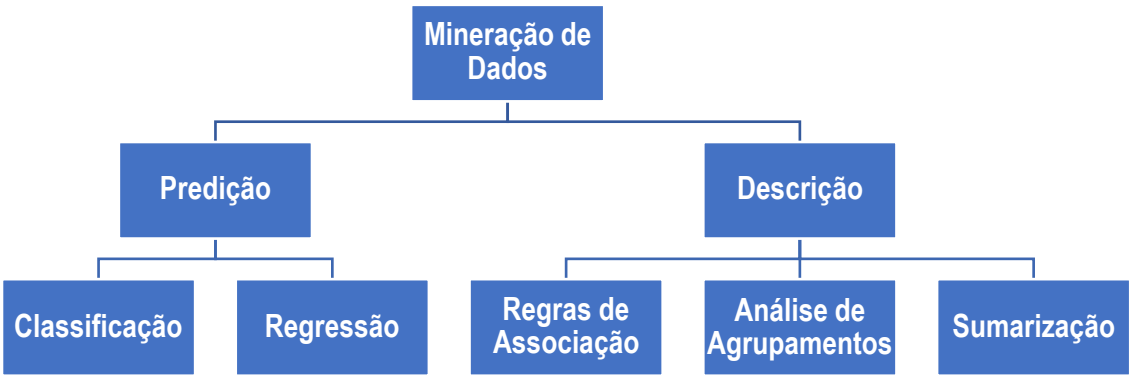


Figura 1.3: Tarefas associadas à mineração de dados.

A utilização de técnicas de inteligência computacional na análise de dados do setor de negócios é uma das principais aplicações para a mineração de dados, sendo amplamente conhecida por *business intelligence*. Um exemplo de atividade de predição ligada à área comercial é a utilização de algoritmos de classificação para identificar e prever fraudes em operações de comércio eletrônico [Coelho *et al*., 2006]. Em relação à atividade de descrição, pode-se citar a utilização de técnicas de análise de agrupamento sobre um determinado público-alvo, com o objetivo de conhecê-lo melhor e identificar suas necessidades [Paço, 2002] [Paço, 2005].

Análise de agrupamentos é uma técnica estatística analítica que pode ser definida como o processo de divisão de um conjunto de dados em grupos de itens similares, onde cada agrupamento consiste de itens semelhantes entre si e diferentes dos itens dos outros grupos. Aplicações para a análise de agrupamentos podem ser identificadas nas mais diversas áreas do conhecimento humano, incluindo o reconhecimento de padrões,

compressão de dados, mineração de dados, segmentação de mercado e redução de dimensionalidade.

1.6 Necessidades Atuais no Processamento de Informações

Em maio de 2006, a Sociedade Brasileira de Computação (SBC) realizou um seminário para discutir os "Grandes Desafios de Pesquisa em Computação no Brasil para o período 2006 a 2016". O evento reuniu renomados pesquisadores brasileiros da área de Computação e contou com o apoio de importantes órgãos, como a Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) e a Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP).

O objetivo do evento foi discutir a pesquisa na área e gerar um documento que apontasse os cinco maiores desafios para a Computação no Brasil para a década seguinte [Carvalho *et al*., 2006]. O primeiro desafio eleito pelo comitê foi a ***gestão da informação em grandes volumes de dados multimídia distribuídos***, demonstrando a importância estratégica da área, tanto no setor comercial quanto no meio acadêmico.

É certo que a informatização é um processo irreversível dentro das empresas, a produção de dados cresce de forma exponencial e a Internet é o meio físico capaz de transformar o mundo em uma enorme base de dados, atualizada em tempo real, por milhares de pessoas. Assim, o comitê justificou que a escolha desse tópico foi impulsionada pela necessidade de buscar soluções para processar volumes cada vez maiores de dados, que possuam escalabilidade e que garantam a segurança e privacidade dos dados analisados.

Um exemplo dessa necessidade é o crescente número de empresas atuais que têm buscado suporte competitivo através da participação em organizações corporativas. Tais organizações são caracterizadas pela existência da aglomeração de um número significativo de empresas que atuam em torno de uma atividade principal e incluem os arranjos produtivos locais, aglomerados de empresas, redes corporativas, cooperativas e franquias [Kotler e Keller, 2006].

Na medida em que essas empresas se agrupam para vencer novos desafios, o seu conhecimento particular acerca do mercado precisa ser compartilhado entre todos, a fim de aumentar o conhecimento do grupo sobre a área de negócio [Thomazi, 2006]. No entanto, nenhuma empresa deseja disponibilizar seus dados sobre clientes e transações comerciais para outras empresas, tanto pela necessidade de manter sigilo comercial quanto por questões relativas à legislação.

Assim, essa necessidade de compartilhamento de informações cria uma série de problemas relativos à segurança e privacidade dos dados, sendo objeto de pesquisa em áreas como mineração de dados distribuída, mineração de dados com preservação da privacidade e análise de agrupamento sobre dados distribuídos, onde a segurança e confidencialidade devem ser mantidas durante todo o processo.

1.7 Problemática

A democratização dos computadores e a sua interligação através de redes locais e da Internet não trouxe apenas vantagens, mas introduziu grandes desafios em relação à tarefa de análise de dados. À medida que o acúmulo de grandes massas de dados tornou-se financeiramente viável e as promessas de benefícios proporcionados pela KDD foram se tornando mais concretas, empresas e organizações passaram a coletar e armazenar inúmeras transações em bases de dados cada vez maiores.

Além disso, a descentralização sofrida pela informática nas últimas décadas impulsionou não apenas a proliferação de bases de dados, mas foi determinante no processo de organização e distribuição dessas bases de dados em múltiplas unidades de armazenamento. Dessa forma, são poucas as empresas e organizações atuais que mantêm seus dados centralizados em um único local.

Algoritmos de mineração de dados, normalmente operam sobre bases de dados centralizadas. Por isso, quando há a necessidade de analisar e processar tais dados em busca de informação e conhecimento, se faz necessária uma etapa de pré-processamento dos dados, onde há a consolidação destes em um único local, o que exige a transferência de grandes volumes de dados entre as unidades remotas e a unidade central.

Por outro lado, muitas organizações mantêm bases de dados geograficamente distribuídas como uma maneira de aumentar a segurança das suas informações [Lam *et al*., 2004] [Zhang *et al*., 2003]. Assim, mesmo que ocorram eventuais falhas em alguma das políticas de segurança, o invasor tem acesso apenas à parte das informações disponíveis. Algumas empresas, mesmo tendo a necessidade de compartilhar suas informações com outras empresas parceiras e até concorrentes, não o fazem em nome da preservação da privacidade das suas informações. Isso acontece, por exemplo, em aplicações médicas e comerciais.

Recentemente, uma classe de algoritmos tem surgido com o objetivo de realizar mineração de dados em unidades distribuídas, reunindo posteriormente os resultados em uma unidade central [Forman e Zhang, 2000] [Silva e Klusch, 2006]. Esta abordagem é denominada mineração de dados distribuída e inclui diversas atividades como análise de agrupamentos e classificação de dados distribuídos. A pesquisa abordada neste trabalho concentra-se especificamente na análise de agrupamentos sobre bases de dados distribuídas.

Esta tese apresenta uma arquitetura, composta por uma estratégia e um conjunto de algoritmos, que tem por finalidade efetuar análise de agrupamentos sobre bases de dados distribuídas, mantendo um nível de confidencialidade que garanta a segurança dos dados analisados. A arquitetura atua sobre dados horizontal e verticalmente distribuídos, o que permite sua aplicação sobre bases de dados distribuídas entre várias unidades remotas. Uma aplicação para a arquitetura é a análise de agrupamentos sobre bases de dados comerciais de clientes com o objetivo de realizar segmentação de mercado,

permitindo inclusive, o compartilhamento de dados entre empresas parceiras e até concorrentes.

1.8 Objetivos e Contribuições do Trabalho

O objetivo principal desse trabalho é apresentar uma arquitetura, composta por uma estratégia e um conjunto de algoritmos, que permite realizar análise de agrupamentos sobre bases de dados distribuídas a partir da utilização de algoritmos tradicionais, tais como mapas auto-organizáveis e *k*-médias, aumentando a segurança e a privacidade aos dados analisados durante o processo.

A arquitetura proposta, denominada partSOM, apresenta três contribuições principais para a tarefa de análise de agrupamentos sobre bases de dados distribuídas: i) realiza agrupamento de dados de forma distribuída, sem necessidade de centralização dos dados em um único local; ii) mantém um nível de confidencialidade que garante a segurança dos dados compartilhados; e iii) contribui para a diversidade das informações obtidas durante o processo de análise de agrupamentos.

A primeira contribuição está associada à estratégia utilizada pela arquitetura, que possibilita realizar análise de agrupamentos sobre bases de dados distribuídas a partir do uso de algoritmos tradicionais, tais como *k*-médias e mapas auto-organizáveis e sem necessidade de centralização dos dados em um único local. A arquitetura possui aplicabilidade tanto sobre bases de dados divididas horizontalmente quanto verticalmente, apresentando eficácia semelhante à obtida com o uso desses algoritmos sobre bases de dados centralizadas. Assim, é possível reduzir sensivelmente a quantidade de dados transferidos entre as unidades remotas e a unidade central.

A segunda contribuição da arquitetura partSOM está associada à necessidade de segurança e preservação da privacidade em mineração de dados distribuídos. A estratégia utilizada pela arquitetura substitui os dados reais de cada unidade por um conjunto de representantes com mesma distribuição estatística dos originais, permitindo manter um nível de confidencialidade que garante a segurança dos dados analisados.

A terceira contribuição da arquitetura provém das informações adicionais obtidas em cada uma das análises parciais que são realizadas. Em um processo tradicional, onde os dados são analisados apenas de forma centralizada, somente um resultado final é obtido. Com o uso da arquitetura partSOM é possível obter-se, além do resultado final, um conjunto de resultados parciais, sendo um para cada subconjunto dos dados existente, o que contribui para o aumento das informações e do conhecimento obtidos durante o processo de análise de agrupamentos.

Ademais, a tese apresenta ainda uma ampla revisão bibliográfica sobre bases de dados geograficamente distribuídas, questões relacionadas à segurança e privacidade de dados distribuídos e discute alguns dos métodos de particionamento de dados mais utilizados. São analisadas vantagens e limitações relacionadas ao uso comitês de

classificação e agrupamento e, finalmente, são apresentadas algumas propostas de combinação de resultados parciais obtidos através do uso de comitês.

1.9 Organização do Documento

Este primeiro capítulo apresentou uma introdução ao tema proposto, incluindo uma contextualização histórica e uma rápida revisão de alguns conceitos relacionados ao processo de descoberta de conhecimento em bancos de dados e mineração de dados. Foram apresentados ainda os principais problemas relacionados ao processo de análise de agrupamentos em bases de dados distribuídas, além dos objetivos e contribuições do trabalho.

O segundo capítulo apresenta os conceitos relacionados à análise de agrupamentos, incluindo uma breve formalização matemática, algumas medidas de similaridade, dissimilaridade e métricas de validação de agrupamentos, além dos métodos e algoritmos mais usados, destacando-se a quantização vetorial e os algoritmos *k*-médias e mapas auto-organizáveis.

No terceiro capítulo é discutida a aplicação da análise de agrupamentos em segmentação de mercado, onde são apresentados alguns dos problemas associados a essa tarefa, como a questão da privacidade e segurança das informações. Além disso, é apresentada uma revisão de diversos trabalhos relacionados ao tema.

No quarto capítulo é feita uma revisão bibliográfica sobre comitês de agrupamento, com foco nas propostas baseadas nos algoritmos *k*-médias e mapas auto-organizáveis, que são os algoritmos mais comuns em análise de agrupamentos e utilizados neste trabalho. São apresentadas as vantagens do uso de comitês sobre as abordagens tradicionais, com destaque para as técnicas mais utilizadas durante a etapa de fusão dos resultados.

O quinto capítulo apresenta a arquitetura proposta nesse trabalho, buscando mostrar o seu potencial para tratar os problemas aqui expostos. Após a apresentação de cada etapa da estratégia e do detalhamento dos algoritmos, a arquitetura é aplicada sobre uma pequena base de dados com o objetivo de ilustrar seu funcionamento e as vantagens obtidas com as análises parciais que a arquitetura permite obter, melhorando a qualidade dos agrupamentos detectados. Em seguida, é feita uma avaliação da arquitetura, onde são apresentadas as suas vantagens e limitações. Ao final do capítulo, são feitas algumas considerações sobre o processo de seleção de atributos que permitem obter vantagens no uso da arquitetura em problemas reais.

O sexto capítulo apresenta resultados da aplicação da arquitetura e das estratégias propostas sobre algumas bases de dados, tanto artificiais, quanto reais. O sétimo e último capítulo traz uma avaliação final dos resultados obtidos nos experimentos do capítulo anterior, além da conclusão e algumas propostas de trabalhos futuros.

1.10 Resumo do Capítulo

Este capítulo apresentou um breve resumo do processo de informatização nas empresas e organizações, desde a revolução industrial até a revolução da informação, demonstrando como essa evolução impulsionou o surgimento de bases de dados cada vez maiores e geograficamente distribuídas.

Em seguida, foram apresentados conceitos relacionados ao processo de descoberta de conhecimento em bases de dados e ao processo de mineração de dados. Dentre as técnicas abordadas, há um destaque para a análise de agrupamentos e os problemas relacionados à utilização desta técnica sobre bases de dados distribuídas. No próximo capítulo serão abordados as principais técnicas e algoritmos relacionados ao processo de análise de agrupamentos.

Capítulo 2
Análise de Agrupamentos

Analisar e agrupar objetos semelhantes entre si, observando algum critério de similaridade entre eles, é uma das tarefas inerentes ao comportamento humano. Normalmente, a análise de agrupamentos está associada às atividades de seleção, classificação e organização. Tais atividades estão presentes em tarefas do nosso cotidiano e, na maioria das vezes, são feitas de forma intuitiva, sem uma clara percepção do indivíduo de sua realização. Por exemplo, na disposição de produtos em uma prateleira de supermercado, obedecendo a critérios de categoria e marca; na escolha dos nossos amigos, a partir de um conjunto de afinidades com a nossa personalidade ou na arrumação de CDs e DVDs em uma estante, separados por estilo ou artista.

Análise de agrupamentos pode ser definida como o processo de dividir um conjunto de dados em um determinado número de grupos ou agrupamentos (do inglês, *clusters*), cujos itens de cada agrupamento sejam semelhantes entre si e diferentes dos itens que compõem os demais agrupamentos.

A análise de agrupamentos não é uma técnica recente. Existem registros de trabalhos nessa área desenvolvidos por Aristóteles (384 – 322 a.C.), que realizou uma classificação de espécies animais, dividindo-os em um grupo de sangue vermelho e outro grupo sem sangue. Mais tarde, no século XVIII, Carolus Linnaeus publicou a mais importante e utilizada taxonomia científica que classifica as coisas vivas, sendo por isso considerado o pai da taxonomia moderna [Frei, 2006]. A descrição inicial da análise de agrupamentos foi formulada por Tryon, no final da década de 30 [Tryon, 1939]. Entretanto, a partir da publicação do livro *Taxonomia Numérica*, a técnica tornou-se definitivamente conhecida, impulsionada pela popularização dos computadores [Sneath e Sokal, 1973].

Uma das dificuldades em lidar com a análise de agrupamentos está na extensa lista de denominações existentes para a mesma técnica. De acordo com a nomenclatura adotada na área, podem-se encontrar os seguintes termos: *análise de agrupamentos* ou *análise de conglomerados* (estatística), *classificação automática de dados* (engenharia), *taxonomia numérica* (biologia), *nosologia* (medicina), *análise tipológica* ou *tipologia* (arquitetura), entre outros [Costa, 1999] [Frei, 2006].

Por se tratar de uma técnica estatística de uso geral, é possível encontrar aplicações para a análise de agrupamentos nos mais diversos domínios e áreas de atuação. A maioria

das aplicações está relacionada a dados sobre os quais se tem pouca ou nenhuma informação sobre a sua estrutura. Por isso, a análise de agrupamentos é uma técnica bastante utilizada para descrição de dados desconhecidos.

2.1 Definição

A análise de agrupamentos faz parte de um conjunto de técnicas estatísticas multivariadas que possibilitam analisar um conjunto de itens considerando, simultaneamente, múltiplas características de cada um deles [Hair Jr. *et al.*, 2005]. Essa técnica tem por objetivo identificar subconjuntos significativos de itens, mutuamente excludentes, com base nas similaridades existentes entre as características de cada item.

Mais formalmente, dado um conjunto de n padrões de entrada $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$, onde cada $\boldsymbol{x}_i = (x_{i1}, x_{i2}, ..., x_{ip})^{\mathrm{T}} \in \mathbb{R}^p$ representa um vetor p-dimensional e cada medida x_{ij} representa um atributo (ou variável) da base de dados de entrada, um processo de análise de agrupamentos consiste em encontrar uma partição C de $\mathbf{X}$, denotada por $C = \{C_1, C_2, ..., C_k\}$, com $k \leq n$, que satisfaça ao seguintes requisitos:

i) $C_i \neq \emptyset,\ i = 1 \ldots k;$

ii) $\bigcup_{i=1}^{k} C_i = \mathbf{X};$

iii) $C_i \cap C_j = \emptyset,\ i = 1 \ldots k, j = 1 \ldots k, i \neq j.$

O primeiro requisito refere-se à exigência de que cada agrupamento possua pelo menos um elemento do conjunto $\mathbf{X}$, não devendo haver, portanto, agrupamentos vazios. O segundo requisito define que a união de todos os elementos dos k agrupamentos corresponda ao conjunto de entrada, garantindo-se que cada padrão de entrada pertença a algum dos agrupamentos. Por fim, o terceiro requisito determina que um padrão de entrada não pertença, simultaneamente, a mais de um agrupamento. Em alguns casos, o último requisito pode ser relaxado, quando se deseja que um mesmo item possa fazer parte, simultaneamente, de mais de um agrupamento. Essa abordagem é denominada partição suave (*soft partition*), em oposição aos métodos tradicionais, denominados de partição rígida (*hard partition*). Nesse trabalho, são analisados apenas os algoritmos que pertencem a essa última abordagem.

2.2 O Processo de Análise de Agrupamentos

Em sua forma convencional, um banco de dados (relacional) é um conjunto de tabelas bidimensionais inter-relacionadas através de um ou mais atributos. Cada tabela é uma estrutura bidimensional de linhas e colunas, onde cada linha representa um registro do banco de dados e cada coluna representa um atributo associado àquele registro. A Figura 2.1 sugere uma amostra típica de uma tabela em um banco de dados.

A proliferação de bases de dados nas mais diversas áreas, incluindo empresas e instituições governamentais e não-governamentais, tem exigido o desenvolvimento de métodos e técnicas cada vez mais eficientes que permitam analisar e extrair informações

e conhecimento embutido nos dados. Nesse contexto, a análise de agrupamento desempenha um papel importante na descrição dos dados analisados, pois permite visualizar relacionamentos frequentemente ocultos em bases de dados muito volumosas e/ou de alta dimensionalidade.

Nº	Nome	Sexo	Idade	Renda	Estado Civil	Filhos	...	Estado
1	A. Araújo	M	39	R$ 2.300,00	Casado	3	...	RN
2	Q. Queiroz	F	82	R$ 1.350,80	Viúvo	2	...	PB
3	W. Wang	M	21	R$ 720,50	Solteiro	1	...	CE
4	E. Eudes	F	18	R$ 1.420,00	Solteiro	0	...	SP
5	S. Silva	M	16	R$ 450,00	Solteiro	0	...	RN
6	G. Gomes	M	42	R$ 32.827,52	Casado	2	...	DF
7	K. Key	F	38	R$ 410,50	Divorciado	1	...	SE
...	...	...	...	...	...	...	...	...
N	M. Mendes	M	21	R$ 3.500,00	Casado	4	...	BA

Figura 2.1: Exemplo de uma tabela em um banco de dados.

O processo de identificar agrupamentos de elementos com características semelhantes em um conjunto de dados é composto de três etapas principais: pré-processamento, processamento e pós-processamento. Tais etapas também estão presentes em outras tarefas de mineração de dados, tais como: regressão, classificação, sumarização e obtenção de regras de associação. O que difere uma da outra, em cada tarefa, é a etapa central, que é responsável pela aplicação dos algoritmos. A seguir, estão descritas cada uma dessas etapas.

2.2.1 Pré-processamento dos Dados

Antes da etapa de aplicação dos algoritmos de análise de agrupamento, assim como em outros processos de mineração de dados, é necessário realizar uma etapa de pré-processamento, que é a etapa responsável pela adequação dos dados aos algoritmos que serão utilizados. Essa etapa do processo compreende um conjunto de procedimentos relacionados à captação, organização, treinamento e preparação dos dados, incluindo, entre outras, a correção de dados errados, a eliminação de dados desnecessários e a formatação dos dados de acordo com as necessidades dos algoritmos [Goldschmidt e Passos, 2005] [Soares, 2007].

Existe um vasto conjunto de opções relacionadas à etapa de pré-processamento dos dados. Alguns dos procedimentos mais utilizados na etapa de pré-processamento são apresentados a seguir [Goldschmidt e Passos, 2005] [Tan *et. al.*, 2006]:

- *Seleção dos dados*: essa etapa compreende a seleção de quais dados entrarão no processo de mineração de dados. Atributos como nome, número de RG e CPF normalmente não são considerados em um processo de análise de agrupamento e podem ser eliminados da base de dados;

- *Derivação de novos atributos*: essa etapa consiste na criação de atributos derivados a partir de outros atributos já existentes que não possam ser utilizados diretamente pelo algoritmo de agrupamento de dados. Por exemplo, o atributo '`data_de_nascimento`' pode ser substituído pelo atributo '`idade`', mais facilmente manuseado pelo algoritmo;
- *Limpeza dos dados*: em aplicações reais, dados costumam estar incompletos, além de conter ruídos ou inconsistências, como valores anômalos ou distorcidos (do inglês, *outliers*) provenientes de falhas no processo de captação. Na etapa de limpeza corrigem-se eventuais erros, por isso, registros contendo dados ausentes, inconsistentes ou ruidosos podem ser eliminados, substituídos ou preenchidos com valores que não interfiram no processo de mineração;
- *Codificação*: a etapa de codificação é responsável por converter os dados para os formatos necessários. Por exemplo, dados categóricos podem ser convertidos em numéricos ou em uma representação binária, para se que ajustem aos requisitos de entrada dos algoritmos;
- *Normalização*: a maioria das métricas de distância leva em consideração a diferença entre os valores dos atributos de cada elemento do conjunto de dados. Assim, atributos que possuam uma escala de valores de maior grandeza podem influenciar mais que outros. Nessa etapa, os valores de cada atributo são ajustados dentro de uma mesma escala, tipicamente em intervalo como [-1, 1] ou [0, 1].
- *Partição do conjunto de dados de entrada*: essa etapa prevê a partição do conjunto de entrada em dois subconjuntos distintos, um para treinamento do algoritmo utilizado e outro para a realização de testes que possam validar o modelo obtido. O principal objetivo é verificar se os resultados obtidos são uma boa abstração do conjunto de dados de entrada, evitando que ele represente apenas os dados para os quais foi treinado.

Ao final da fase de pré-processamento, os dados normalmente estão dispostos em única tabela conhecida como matriz de dados, que deve satisfazer às necessidades do algoritmo que será utilizado. A matriz de dados **D** é formada por um conjunto de *n* vetores, onde cada vetor representa um elemento do conjunto de entrada. Cada vetor possui *p* componentes, que correspondem ao conjunto de atributos que o identificam. Um exemplo de matriz de dados para a tabela da Figura 2.1 é apresentado na Figura 2.2.

0,72457	-0,72457	0,20077	-0,27575	-0,72457	···	-0,35355	-0,35355
-1,2076	1,2076	2,1741	-0,36094	-0,72457	···	-0,35355	-0,35355
0,72457	-0,72457	-0,62526	-0,41751	1,2076	···	-0,35355	-0,35355
-1,2076	1,2076	-0,76294	-0,35473	1,2076	···	-0,35355	2,4749
...	...	...	...	...	...	...	...
0,72457	-0,72457	-0,62526	-0,16805	-0,72457	···	-0,35355	-0,35355

Figura 2.2: Exemplo de uma matriz de dados obtida após a etapa de pré-processamento.

2.2.2 Aplicação dos Algoritmos

Uma vez que os dados estejam adequados à utilização, essa fase corresponde à aplicação de um ou mais algoritmos de análise de agrupamento sobre a matriz de dados com a finalidade de particionar o conjunto de entrada em agrupamentos distintos de itens similares.

De acordo com Goldschmidt e Passos (2005), todo o processo de KDD deve ser norteado por objetivos e esses objetivos definem qual tarefa de mineração de dados deve ser executada. Conforme foi citado anteriormente, as etapas de pré-processamento e pós-processamento são comuns a outras tarefas de mineração de dados. A etapa de aplicação dos algoritmos é a responsável direta pelo resultado que se deseja obter, sendo portando o diferencial entre um ou outro processo.

A saída produzida por essa etapa pode variar em função do algoritmo escolhido. No entanto, em se tratando de tarefas de análise de agrupamentos, o algoritmo deve fornecer como saída uma indicação do número de grupos existentes (caso essa informação não seja conhecida) e um modelo que permita associar cada elemento do conjunto de entrada a um dos grupos identificados. Eventualmente, pode ser necessária a aplicação da etapa de pós-processamento para que se tenha um resultado definitivo.

2.2.3 Processamento dos Resultados

A fase de pós-processamento visa analisar e interpretar os resultados obtidos na etapa anterior para que se obtenha um resultado final. Nessa etapa, com a ajuda de um especialista no domínio da aplicação, é possível identificar falhas ou realizar ajustes necessários ao modelo para uma melhor compreensão dos resultados.

Em uma tarefa de análise de agrupamentos, por exemplo, uma possível simplificação do modelo obtido poderia ser a unificação de dois ou mais agrupamentos em um único, uma vez que agrupamentos com poucos elementos, que tenham características muito semelhantes a outros, podem ser unificados, simplificando a solução obtida. Por outro lado, agrupamentos que contenham elementos muito distintos podem ser desassociados para formar novos agrupamentos.

2.3 Medidas de Similaridade e Dissimilaridade

A tarefa de identificar itens semelhantes dentre os existentes no conjunto de entrada pressupõe a adoção de alguma métrica de distância entre os itens que possa determinar a proximidade entre eles. Existem dois tipos de métricas de distância: a similaridade mostra a semelhança entre os itens, ou seja, quanto maior a similaridade, mais parecidos (ou próximos) são os itens. A dissimilaridade mede a diferença entre os itens, quanto maior a dissimilaridade, mais diferentes (ou distantes) eles o são [Frei, 2006].

Considerando cada item do conjunto de entrada como sendo um vetor no espaço p-dimensional, uma função de distância entre dois itens $\boldsymbol{x}_i$ e $\boldsymbol{x}_j$ do conjunto $\mathbf{X}$ pode ser definida como sendo:

$$d\colon \mathbf{X} \times \mathbf{X} \rightarrow \mathbb{R}$$
$$d_{ij} = d(\boldsymbol{x}_i, \boldsymbol{x}_j)$$

onde d_{ij} é um valor real associado a cada par de itens do conjunto de entrada e é calculado a partir de uma medida de similaridade (ou dissimilaridade) que atenda às seguintes premissas:

i) $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = d(\boldsymbol{x}_j, \boldsymbol{x}_i)$, $\forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbf{X}$ (simetria);

ii) $d(\boldsymbol{x}_i, \boldsymbol{x}_j) \geq 0$, $\forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbf{X}$ (positividade);

iii) $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = 0$ sse $\boldsymbol{x}_i = \boldsymbol{x}_j$, $\forall \boldsymbol{x}_i, \boldsymbol{x}_j \in \mathbf{X}$ (reflexividade);

iv) $d(\boldsymbol{x}_i, \boldsymbol{x}_j) \leq d(\boldsymbol{x}_i, \boldsymbol{x}_k) + d(\boldsymbol{x}_k, \boldsymbol{x}_j)$, $\forall \boldsymbol{x}_i, \boldsymbol{x}_j, \boldsymbol{x}_k \in \mathbf{X}$ (desigualdade triangular)

Na literatura, são apresentadas diversas métricas de similaridade e dissimilaridade que atendem a essas condições. A escolha da métrica está associada a características do conjunto de entrada, tais como a natureza das variáveis (discretas, contínuas, binárias, etc.), a escala das medidas (nominal, ordinal, intervalar, etc.), o formato dos agrupamentos no espaço p-dimensional (esféricos, quadrados, elípticos, etc.) e, até mesmo, a preferência do pesquisador [Kasznar e Gonçalves, 2007] [Bussab *et al*., 1990].

A seguir são apresentadas algumas métricas de similaridade e dissimilaridade bastante usadas em tarefas de análise de agrupamentos. Foram consideradas apenas métricas associadas a variáveis numéricas, uma vez que os algoritmos aqui considerados trabalham somente com esse tipo de variável.

Distância Euclidiana

A distância euclidiana é a métrica mais utilizada em tarefas de classificação e análise de agrupamentos, sendo uma generalização da distância entre dois pontos em um plano cartesiano e é dada pela a raiz quadrada da soma dos quadrados das diferenças de valores de cada atributo. Matematicamente, é definida por:

$$d_{ij} = \left(\sum_{f=1}^{p} \left| \boldsymbol{x}_{if} - \boldsymbol{x}_{jf} \right|^2 \right)^{1/2} \qquad (2.1)$$

onde $\boldsymbol{x}_{if}$ e $\boldsymbol{x}_{jf}$ são dois vetores de entrada no espaço p-dimensional, $\boldsymbol{x}_{if}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_i$ e $\boldsymbol{x}_{jf}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_j$.

Distância de Manhattan

A distância de Manhattan (também conhecida como distância quarteirão ou distância *city block*) é uma variação da distância euclidiana. No caso de vetores do espaço p-dimensional, corresponde à distância entre dois pontos medida ao longo de eixos com ângulos retos. Matematicamente, é definida por:

$$d_{ij} = \sum_{f=1}^{p} \left| \boldsymbol{x}_{if} - \boldsymbol{x}_{jf} \right| \tag{2.2}$$

onde $\boldsymbol{x}_{if}$ e $\boldsymbol{x}_{jf}$ são dois vetores de entrada no espaço p-dimensional, $\boldsymbol{x}_{if}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_i$ e $\boldsymbol{x}_{jf}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_j$.

Distância de Minkowski

A distância de Minkowski é uma generalização das distâncias euclidiana e de Manhattan. Matematicamente, a distância de Minkowski de ordem n entre dois vetores é definida por:

$$d_{ij} = \left(\sum_{f=1}^{p} \left| \boldsymbol{x}_{if} - \boldsymbol{x}_{jf} \right|^{n} \right)^{1/n} \tag{2.3}$$

onde $\boldsymbol{x}_{if}$ e $\boldsymbol{x}_{jf}$ são dois vetores de entrada no espaço p-dimensional, $\boldsymbol{x}_{if}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_i$ e $\boldsymbol{x}_{jf}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_j$.

A distância de Manhattan corresponde ao caso particular em que $n = 1$ e a distância euclidiana corresponde ao caso particular em que $n = 2$.

Distância de Chebychev

A distância de Chebychev (ou distância do máximo valor) é um caso particular da distância de Minkowski, obtida quando $n \to \infty$. Corresponde ao maior valor absoluto entre as diferenças de valores de cada atributo. No caso de vetores do espaço p-dimensional, é a máxima distância entre dois pontos em qualquer das dimensões. Matematicamente, é definida por:

$$d_{ij} = \lim_{n \to \infty} \left(\sum_{f=1}^{p} \left| \boldsymbol{x}_{if} - \boldsymbol{x}_{jf} \right|^{n} \right)^{1/n} = \max_{1 \le f \le p} \left| \boldsymbol{x}_{if} - \boldsymbol{x}_{jf} \right| \tag{2.4}$$

onde $\boldsymbol{x}_{if}$ e $\boldsymbol{x}_{jf}$ são dois vetores de entrada no espaço p-dimensional, $\boldsymbol{x}_{if}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_i$ e $\boldsymbol{x}_{jf}$ corresponde ao f-ésimo atributo do vetor $\boldsymbol{x}_j$.

2.4 Métodos e Algoritmos Mais Utilizados

Tradicionalmente, algoritmos de agrupamento de dados são divididos em métodos hierárquicos e não-hierárquicos [Kantardzic, 2002] [Larose, 2005] [Pedrycz, 2005]. Os hierárquicos trabalham agregando os objetos do conjunto de entrada em uma estrutura hierárquica, semelhante a uma árvore de grupos e podem ser subdivididos em aglomerativos e divisivos [Han e Kamber, 2006].

Nos métodos aglomerativos, o processo se inicia com um conjunto de n grupos, cada um contendo um único item. Em seguida, é realizada uma série de agrupamentos sucessivos nos quais os itens são agregados para formar novos conjuntos (abordagem *bottom-up*). Uma vez que dois elementos sejam unidos, eles permanecerão unidos até o final do processo.

Nos métodos divisivos acontece justamente o contrário, o processo se inicia com um único conjunto contendo todos os itens, que são continuamente desagregados até que cada item pertença a um conjunto unitário (abordagem *top-down*). Uma vez que dois itens sejam separados, jamais voltarão a fazer parte do mesmo agrupamento [Albuquerque *et al*., 2006]. Em ambos os casos, os resultados parciais de cada iteração servem como entrada para o passo seguinte do algoritmo, em um processo recursivo, até que todos os elementos tenham sido processados.

As principais vantagens dos métodos hierárquicos são a sua facilidade de implementação e a forma de apresentação dos resultados, normalmente através de uma estrutura hierárquica denominada dendograma, que permite manter um histórico de todos os passos seguidos pelo algoritmo. Sua principal desvantagem é o alto custo computacional, por este motivo não são indicados para conjuntos de dados muito grandes.

Conforme demonstrado em Jain *et al.* (1999), os métodos aglomerativos possuem complexidade $O(n^2 \log n)$, o que é uma grande desvantagem dessa abordagem. Os métodos divisivos, em particular, são ainda mais ineficientes, uma vez que possuem complexidade exponencial $O(2^n)$. Para um conjunto de n itens, temos $2^{n-1} - 1$ partições possíveis [Xu e Wunsch II, 2005]. Tais métodos precisam considerar todas as possíveis partições de um conjunto em dois subconjuntos, sendo viáveis apenas para conjuntos de dados muito pequenos.

Os resultados de um algoritmo hierárquico são normalmente apresentados na forma de um dendrograma, que é um tipo de diagrama cuja ordenação hierárquica se assemelha aos ramos de uma árvore que se vão dividindo sucessivamente [Pedrycz, 2005]. As divisões do dendrograma correspondem a cada uma das etapas do algoritmo, o que permite traçar um histórico dos passos que foram realizados desde o início até o resultado final. Um exemplo de dendograma é apresentado na Figura 2.3. As linhas horizontais indicam pontos onde é possível interromper o processo e obter uma solução particular para o problema.

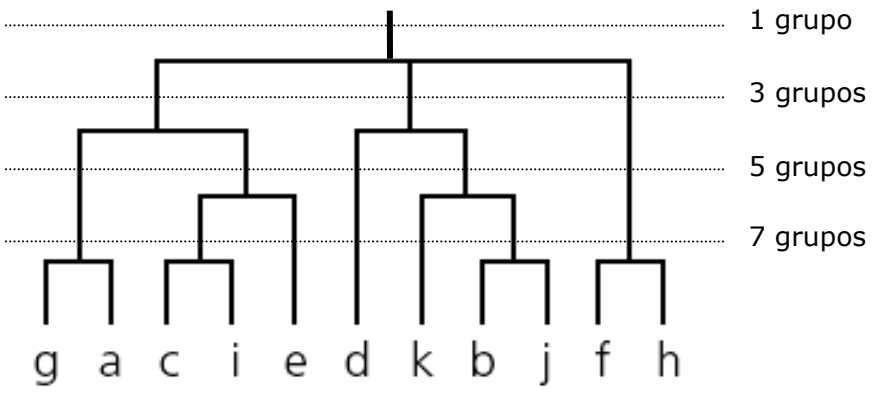


Figura 2.3: Exemplo de um dendograma.

A grande maioria dos métodos hierárquicos aglomerativos é bastante semelhante entre si, com pequenas diferenças em relação à forma de computar a distância entre os agrupamentos. A seguir, são apresentados cinco dentre os métodos mais comuns desta categoria:

- Ligação Simples, também conhecido como vizinho mais próximo, é um dos mais simples dentre os algoritmos hierárquicos aglomerativos. Sua principal característica é considerar sempre a menor distância ao medir a dissimilaridade entre dois agrupamentos de elementos;
- Ligação Completa, também conhecido como vizinho mais distante, é bem semelhante ao anterior. Porém, sua principal característica é considerar sempre a maior distância ao medir a dissimilaridade entre dois agrupamentos de elementos;
- Ligação Média, também conhecido como distância média e bastante semelhante aos anteriores. Nesse caso, é considerada a distância média (aritmética) ao medir a dissimilaridade entre dois agrupamentos de elementos;
- Ligação Mediana, também conhecido como método dos centroides, é uma variação do método anterior. Nesse caso, são considerados os valores dos centroides de cada grupo ao medir a dissimilaridade entre dois agrupamentos de elementos;
- Método de Ward busca obter sempre o mínimo desvio padrão entre os dados de cada grupo. Nesse caso, ao medir a dissimilaridade entre dois grupos de elementos é considerada a soma das diferenças entre os elementos de cada agrupamento e o valor médio do agrupamento.

Uma vez que são pouco utilizados na prática por causa da sua complexidade computacional, os métodos hierárquicos foram aqui descritos para fins didáticos de ilustração do processo de agrupamento; entretanto, dada a sua menor eficiência [Jain *et al.*, 1999] [Xu e Wunsch II, 2005], não foram utilizados nos experimentos realizados neste trabalho.

Na literatura, é comum encontrar autores que tratam todos os demais métodos como pertencentes à classe dos não-hierárquicos (ou particionais) [Kantardzic, 2002] [Larose, 2005]. Porém, alguns trabalhos recentes fornecem uma classificação mais detalhada, argumentando que as classificações disponíveis não são convenientes, principalmente em função da variedade de técnicas utilizadas e da sobreposição de alguns algoritmos em mais de uma classe [Xu e Wunsch II, 2005] [Berkhin, 2006].

Enquanto os métodos hierárquicos atuam através de sucessivas junções (ou disjunções) sobre um conjunto de dados para determinar novos agrupamentos, os métodos particionais tentam detectar os agrupamentos a partir da disposição dos dados no espaço p-dimensional, seja de forma aleatória ou identificando áreas com maior densidade populacional. Os métodos baseados no conceito de quantização vetorial dividem o

conjunto de dados em *k* partições e, em seguida, utilizam alguma técnica de realocação iterativa para melhorar o particionamento obtido, remanejando elementos de um agrupamento para outro [Han e Kamber, 2006], sendo que tais métodos obtêm melhores resultados quando operam sobre agrupamentos convexos [Berkhin, 2006].

Existem, ainda, diversos outros tipos de algoritmos de agrupamento, cuja classificação é baseada principalmente em função das técnicas que utilizam. Como exemplos, podemos citar: algoritmos baseados em grafos, em técnicas de busca combinatória, em algoritmos genéticos, em redes neurais artificiais, em lógica difusa, etc.

Esse trabalho aborda apenas os dois algoritmos: *k*-médias (*k-means*) e mapas auto-organizáveis (*self-organizing maps*), que serão detalhados a seguir. Antes disso, é feita uma breve introdução ao conceito de quantização vetorial. Primeiro, pela sua influência nos dois algoritmos que seguem. Segundo, pela importância com que será utilizada no decorrer do trabalho.

2.4.1 Quantização Vetorial

Quantização Vetorial (QV) é uma técnica de agrupamento de dados cujo objetivo é dividir um espaço de entrada em um número pré-definido de regiões distintas e eleger um vetor para ser o representante de cada região. Assim, todos os elementos do conjunto de entrada que pertencem a uma mesma região podem ser representados pelo vetor correspondente àquela região, conforme ilustrado na Figura 2.4. Essa estratégia é empregada para reduzir a quantidade de espaço necessária para armazenar ou transmitir um conjunto de dados e tem sido amplamente utilizada em tarefas de agrupamento de dados e compressão de sinais, especialmente voz e imagem.

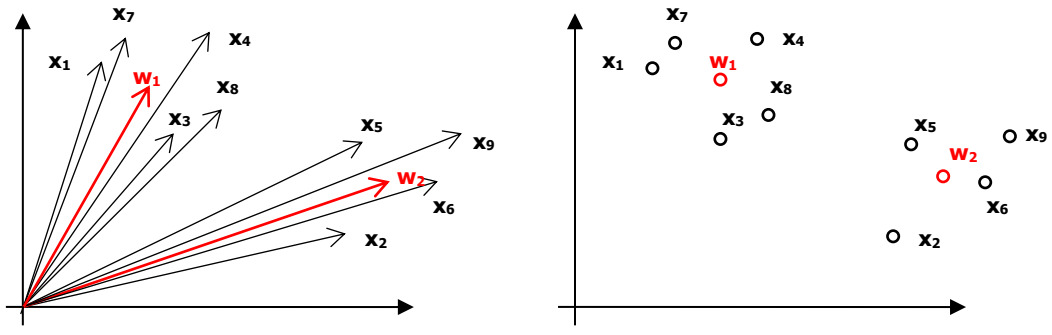


Figura 2.4: Representação do processo de quantização vetorial no plano 2D.

Proposto por Linde *et al*. (1980), o algoritmo LBG é um dos mais importantes da área, não apenas pela sua ampla utilização, mas principalmente por ter influenciado o surgimento de diversos outros algoritmos que utilizam funções de custo para cômputo dos agrupamentos [Kohonen, 2001] [Fleck, 2004].

A QV permite discretizar um espaço contínuo de entrada, representando um conjunto contínuo de dados através de um número finito de vetores, de forma que a cada vetor contínuo de entrada seja atribuído um vetor protótipo que melhor representa as suas características. A saída do sistema é um conjunto finito de vetores-representantes do espaço de entrada, conhecidos como *codebook* (além de possuir outras denominações, como vetor de referência, vetor código e dicionário). O *codebook* é uma representação discreta do espaço de entrada.

Mais formalmente, dado um conjunto de n padrões de entrada $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$, onde cada $\boldsymbol{x}_i = (x_{i1}, x_{i2}, ..., x_{ip})^{\mathrm{T}} \in \mathbb{R}^p$ representa um vetor p-dimensional e dado um conjunto de vetores referência $\mathbf{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k\}$, a quantização vetorial pode ser definida como um mapeamento M do espaço de amostras $\mathbf{X}$ para o conjunto $\mathbf{W}$, de tal forma que:

$$M : \mathbb{R}^p \rightarrow \mathbf{W}$$

$$\boldsymbol{w}_\xi = \min(d(\boldsymbol{x}_i, \boldsymbol{w}_j)),\ i = 1..n, j = 1..k$$

onde $\boldsymbol{x}_i$ é o i-ésimo elemento do conjunto de entrada, $\boldsymbol{w}_j$ é o j-ésimo elemento do conjunto de vetores referência $\mathbf{W}$ e $\boldsymbol{w}_\xi$ é o elemento de $\mathbf{W}$ que melhor representa $\boldsymbol{x}_i$, medido através da utilização de uma função custo, que computa a distância entre o elemento $\boldsymbol{x}_i$ e cada elemento de $\mathbf{W}$.

Obviamente, sempre haverá um erro entre os vetores originais (do espaço de amostras) e os seus representantes (do conjunto de vetores referência), conhecido na literatura como distorção. De acordo com Jain (1999) e Kohonen (2001), a função padrão mais intuitiva e frequentemente utilizada pelos métodos particionais de agrupamento para mensurar a distorção é a função erro médio quadrático (do inglês, *mean square error*, ou MSE), dada por:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}\left(\boldsymbol{x}_i - \boldsymbol{w}_\xi\right)^2 \tag{2.5}$$

onde $\boldsymbol{x}_i$ é o i-ésimo elemento do conjunto de entrada e $\boldsymbol{w}_\xi$ é o elemento de $\mathbf{W}$ que melhor representa $\boldsymbol{x}_i$. Este erro pode ainda ser definido como a esperança do quadrado da distância Euclidiana. Fleck (2004) lembra que alguns itens devem ser observados na escolha da função custo, como o fato de ser derivável em todos os pontos e ser convergente, sugerindo a seguinte função, que atende a esses requisitos:

$$f_x(\boldsymbol{w}) = \sum_{i=1}^{n}\frac{\left(\boldsymbol{x}_i - \boldsymbol{w}\right)^2}{2} \tag{2.6}$$

cuja primeira derivada é dada por:

$$f'_x(\boldsymbol{w}) = \sum_{i=1}^{n}\left(\boldsymbol{w} - \boldsymbol{x}_i\right) \tag{2.7}$$

e segunda derivada é dada por:

$$f''_x(\boldsymbol{w}) = N \tag{2.8}$$

o que garante que f_x é convexa e tem um ponto de mínimo local quando f_x' é zero, que pode ser calculado da seguinte forma:

$$f_x'(\boldsymbol{w}) = \sum_{i=1}^{n} (\boldsymbol{w} - \boldsymbol{x}_i) = 0$$

$$n\boldsymbol{w} - \sum_{i=1}^{n} (\boldsymbol{x}_i) = 0$$

$$\boldsymbol{w} = \frac{1}{n} \sum_{i=1}^{n} (\boldsymbol{x}_i) \tag{2.9}$$

Da equação 2.9 pode-se concluir que o ponto onde o erro mínimo é obtido corresponde à média dos elementos do grupo.

O método do gradiente descendente, também conhecido como método da descida mais íngreme, é uma das técnicas mais utilizadas na formação de grupos através processos iterativos, como os métodos particionais de agrupamento [Fleck, 2004]. Esse método considera que, partindo de um valor aleatório inicial de $\boldsymbol{w}$, é possível atingir o $\boldsymbol{w}_\xi$ através de constantes atualizações de $\boldsymbol{w}$ na direção do valor mínimo da função f_x. A atualização é definida por:

$$\boldsymbol{w}(t+1) = \boldsymbol{w}(t) - \eta \Delta E \tag{2.10}$$

onde t representa a iteração e η é o termo de aprendizagem, que normalmente decresce a cada iteração e ΔE é o gradiente de erro.

Considerando que o conjunto de vetores referência $\mathbf{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k\}$ é formado por k elementos, a cada iteração (ou época, segundo a literatura) o algoritmo recalcula cada elemento $\boldsymbol{w}_\xi \in \mathbf{W}$ de forma a que este melhor represente a região e ele associada. Assim, conforme descrito em Haykin (2001), desde que os parâmetros do algoritmo sejam determinados e ajustados adequadamente, o método do gradiente descendente converge gradativamente para uma solução ótima, utilizando a função custo como referência.

O gradiente de erro (ΔE) é dado pela variação da primeira derivada da função f_x, apresentado na equação 2.7 [Fleck, 2004]. Substituindo a derivada da função custo na equação 2.10, temos:

$$\boldsymbol{w}_\xi(t+1) = \boldsymbol{w}_\xi(t) - \eta \left(\sum_{i=1}^{n} (\boldsymbol{w} - \boldsymbol{x}_i) \right)$$

$$\boldsymbol{w}_\xi(t+1) = \boldsymbol{w}_\xi(t) + \eta (\boldsymbol{X}(t) - \boldsymbol{w}_\xi(t)) \tag{2.11}$$

onde $\boldsymbol{w}_\xi(t)$ representa o protótipo (vetor representante) de cada agrupamento na iteração t e $\mathbf{X}(t)$ representa o conjunto de dados de entrada na iteração t. A equação 2.11 serve de base para vários algoritmos que utilizam funções de custo para realizar análise de agrupamento (orientados à busca do representante médio do agrupamento), dentre eles os algoritmos *k*-médias e mapas auto-organizáveis.

2.4.2 Algoritmo *k*-Médias

Proposto inicialmente por MacQueen (1967), o *k*-médias (originalmente *k-means*) é um algoritmo de análise de agrupamentos particional, sendo um dos mais conhecidos e utilizados em tarefas de análise de agrupamentos, principalmente pela sua simplicidade e facilidade de implementação.

Assim como em outros algoritmos de agrupamento, o objetivo do algoritmo *k*-médias é agrupar um conjunto de *n* itens em *k* grupos, com base em alguma medida de similaridade, que normalmente é a distância Euclidiana. O *k*-médias é anterior ao algoritmo LBG e ambos são muito parecidos e baseados nos mesmos princípios. De fato, Linde *et al*. (1980) concordam que a ideia básica de encontrar partições com distorção mínima (LBG) e centroides (*k*-médias) é exatamente a mesma, o que diferencia os dois é a sequência de treinamento dos dados, o que torna os resultados obtidos, em geral, ligeiramente distintos.

O LBG é treinado de forma sequencial, incorporando um a um os vetores de treinamento e finalizando após a codificação do último vetor. Ao invés disso, o *k*-médias considera todos os vetores em cada iteração e repete o processo até a convergência [Linde *et al*., 1980].

A convergência acontece quando não há mudanças nos valores dos centroides ou quando o processamento atinge o valor limite de iterações, normalmente muito alto. Ao final do processamento, cada elemento é dito pertencer ao agrupamento representado pelo seu centroide.

O algoritmo *k*-médias é facilmente descrito e os seus passos são apresentados a seguir:

Algoritmo *k*-médias

1. **Defina o valor de *k*, correspondente ao número de agrupamentos da amostra;**
2. **Escolha aleatoriamente um conjunto de *k* centroides para representar os agrupamentos;**
3. **Calcule uma matriz de distâncias entre cada um dos elementos do conjunto de dados e os cada centroide;**
4. **Associe cada elemento ao seu centroide mais próximo;**
5. **Recalcule o valor de cada centroide a partir da média dos valores dos elementos que pertence a este centroide, gerando uma nova matriz de distâncias;**
6. **Retorne ao passo 4 e repita até a convergência.**

O algoritmo *k*-médias possui complexidade linear $O(npk)$, onde n e p são, respectivamente, o número de elementos e a dimensionalidade do conjunto de dados e k é o número de agrupamentos desejado. O *k*-médias possui boa escalabilidade, uma vez

que os valores de p e k são, na maioria das vezes, muito menores do que n [Xu e Wunsch II, 2005]. Além disso, por ser baseado no princípio da quantização vetorial, o algoritmo funciona bem sobre agrupamentos compactos, hiperesféricos e bem definidos.

Entre as desvantagens do *k*-médias está a necessidade de se fornecer um valor pré-definido para k, o número de agrupamentos, o que muitas vezes é feito de forma aleatória. A estratégia mais utilizada para contornar essa dificuldade é executar o algoritmo diversas vezes, para diferentes valores de k e medir-se a coesão dos agrupamentos detectados através de índices de validação de agrupamentos. Chiang e Mirkin (2007) apresentam diversas outras técnicas para abordar esse problema.

Leisch (1998) aponta como principal desvantagem do *k*-médias o fato do mesmo ser um algoritmo não-determinístico, fortemente influenciado tanto pelos valores de inicialização como por pequenas mudanças no conjunto de treinamento, que podem ocasionar grandes alterações na solução para a qual o algoritmo converge, o que o torna um algoritmo bastante instável. Como a escolha dos valores iniciais dos centroides, normalmente é feita de forma aleatória ou a partir de elementos que compõem o conjunto de dados de entrada, essa estratégia é bastante criticada e algumas variações têm sido propostas para melhorar o desempenho do algoritmo [Mirkin, 2005].

Han e Kamber (2006) ressaltam ainda que o *k*-médias não é um método adequado para lidar com agrupamentos com formato não-convexo ou com agrupamentos de diferentes tamanhos, além de ser bastante sensível a ruídos e valores anômalos (do inglês, *outliers*), de forma que um pequeno número de dados com tais características é capaz de influenciar significativamente os valores dos centroides.

A despeito de todas as críticas, o *k*-médias é um dos algoritmos de análise de agrupamento mais pesquisados, possuindo um grande número de variantes que diferem em pequenos detalhes, como na forma de selecionar os centroides iniciais, no cálculo da similaridade entre os centroides e os elementos do conjunto de entrada e nas estratégias utilizadas para computar os centroides de cada agrupamento [Han e Kamber, 2006].

Alguns exemplos de variações do *k*-médias são o *k*-modas (*k-modes*), que usa o conceito de moda, ao invés de média, para agrupar dados categóricos e o *k*-representantes (*k-medoids*), que usa exemplos reais do conjunto de entrada para representar os centros dos agrupamentos, reduzindo a influência de ruídos e distorções. Além disso, outros algoritmos que foram desenvolvidos em seguida, como o LBG, o Expectation-Maximization e os mapas auto-organizáveis, compartilham de ideias comuns com o *k*-médias.

2.4.3 Mapas Auto-Organizáveis

Redes neurais artificiais (RNA) são uma importante ferramenta computacional, com forte inspiração neurobiológica e largamente utilizada na solução de problemas complexos que, em geral, não podem ser tratados eficientemente através de soluções algorítmicas tradicionais [Haykin, 2001]. Aplicações para RNA incluem reconhecimento

de padrões; análise e processamento de sinais; tarefas de análise, diagnóstico e prognóstico; classificação e agrupamento de dados.

Dentre os inúmeros modelos de redes neurais existentes, um deles, em particular, tem sido amplamente utilizado em tarefas de classificação automática de dados, visualização de dados de dimensão elevada e na redução de dimensionalidade: os mapas auto-organizáveis (*self-organizing maps* – SOM) [Kohonen, 2001]. Os mapas auto-organizáveis foram desenvolvidos no final da década do 90, inspirados em estudos sobre redes associativas e sistemas auto-organizáves [Kohonen, 1984] [Kohonen, 1990].

As redes neurais SOM, também chamados de mapas de Kohonen, constituem uma classe de rede neural, de aprendizado não supervisionado, conhecidas como redes competitivas. Neste tipo de rede, todos os neurônios (unidades básicas de processamento da rede) recebem o mesmo estímulo de entrada e competem entre si para identificar quem é o neurônio vencedor (aquele mais similar ao estimulo de entrada).

Uma importante característica das RNA do tipo SOM é a compressão de informação, enquanto mantêm preservadas relações topológicas e métricas existentes nos dados de entrada. Isso significa que elementos do conjunto de entrada que possuam características semelhantes tendem a permanecer juntos quando projetados na camada de saída da rede.

A arquitetura de uma rede neural do tipo SOM é extremamente simples, consistindo apenas de duas camadas de neurônios (Figura 2.5). A primeira camada (de entrada), composta por um vetor com p neurônios, representa a dimensionalidade do conjunto de entrada (ou seja, a quantidade de atributos da tabela de dados). Cada neurônio de entrada está conectado a todos os neurônios da camada seguinte. A segunda camada, também conhecida como camada de saída, representa o mapa onde o conjunto de entrada será projetado, sendo composta por um conjunto de neurônios, normalmente dispostos na forma de um vetor (unidimensional) ou de uma matriz (bidimensional), onde cada neurônio está conectado apenas aos seus vizinhos.

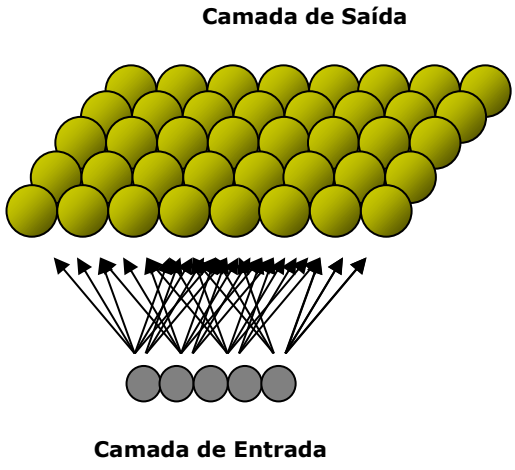


Figura 2.5: Arquitetura de uma rede neural do tipo SOM.

Durante a etapa de treinamento de uma rede neural do tipo SOM, cada representante do conjunto de entrada é selecionado aleatoriamente e apresentado à camada de entrada da rede. Uma função de ativação calcula a semelhança entre o vetor de entrada e todos os neurônios do mapa. O neurônio da camada de saída que se apresentar como mais similar ao neurônio de entrada é declarado vencedor e os seus pesos sinápticos, assim como os dos seus vizinhos, são realçados. O processo se repete com os demais vetores do conjunto de entrada, por várias épocas, até que a rede esteja treinada.

A função de similaridade comumente utilizada para calcular a distância entre o vetor de entrada e os neurônios da rede é a distância euclidiana, apresentada anteriormente na equação 2.1, que para a rede SOM é dada por:

$$d_{ij} = \left(\sum_{f=1}^{p} \left| \boldsymbol{x}_{if} - \boldsymbol{w}_{jf} \right|^2 \right)^{1/2} \tag{2.12}$$

onde $\boldsymbol{x}_{if}$ representa o *f*-ésimo atributo do *i*-ésimo vetor de entrada no espaço *p*-dimensional e $\boldsymbol{w}_{jf}$ representa o *f*-ésimo atributo do *j*-ésimo neurônio da camada de saída.

Para identificação do neurônio vencedor (*best match unit*, ou simplesmente, *bmu*), verifica-se dentre todos os neurônios da camada de saída, qual deles possui a menor distância para o vetor de entrada, usando-se a equação:

$$\boldsymbol{w}_{\xi} = \min\left(d_{ij}\right) \tag{2.13}$$

onde $\boldsymbol{w}_{\xi}$ representa o neurônio vencedor e d_{ij} representa a distância euclidiana entre um elemento do conjunto de entrada e um neurônio da camada de saída

Os pesos sinápticos do neurônio vencedor e da sua vizinhança são atualizados através da equação:

$$\boldsymbol{m}_i(t+1) = \boldsymbol{m}_i(t) + h_{ci}(t)\left[\boldsymbol{x}(t) - \boldsymbol{m}_i(t)\right] \tag{2.14}$$

onde *t* representa o tempo, $\boldsymbol{x}(t)$ representa um elemento qualquer do conjunto de entrada no instante *t* e h_{ci} determina o raio de vizinhança que será modificado, normalmente sendo reduzido na medida em que o algoritmo de treinamento avança. A equação 2.14 é equivalente à equação 2.11, adaptada para o algoritmo SOM.

Em uma tarefa de identificação de padrões, uma das principais aplicações para este tipo de rede, ser o neurônio vencedor significa ser o mais semelhante, dentre os existentes no mapa de saída, ao valor apresentado à entrada da rede. O neurônio vencedor, juntamente com a sua vizinhança, tem seus valores realçados, de forma que se a mesma entrada for apresentada à rede posteriormente, aquela região do mapa irá ficar ainda mais realçada.

Uma vez concluído o processo de treinamento, a rede SOM pode ser utilizada para classificar padrões e agrupá-los segundo suas características presentes no espaço de entrada. Uma vez que a rede associa cada padrão apresentado na camada de entrada a um neurônio da camada de saída, pode-se usar esse mapeamento para agrupar dados semelhantes.

As redes do tipo SOM possuem a habilidade de formar um mapa topográfico dos padrões de entrada, de forma que a disposição dos neurônios na grade (mapa SOM) reflete características estatísticas contidas nos dados de entrada. Tal característica permite que elementos próximos entre si no conjunto de entrada permaneçam próximos na projeção realizada na camada de saída. Com isso, as redes SOM preservam a topologia entre os espaços de entrada e saída, permitindo que se enxerguem padrões de comportamentos que seriam invisíveis em um ambiente *p*-dimensional [Costa, 1999].

Outras propriedades da rede SOM podem ser resumidas como descrito a seguir [Haykin, 2001] [Gonçalves *et al.*, 2008]:

- *Aproximação do espaço de entrada*: o SOM tem como objetivo básico armazenar um conjunto grande de vetores de entrada encontrando um conjunto menor de protótipos **W** (também chamados de vetores de pesos sinápticos ou neurônios), de modo a fornecer uma boa aproximação para o espaço de entrada original. A base teórica dessa estratégia está fundamentada na teoria da quantização vetorial, cuja motivação é a redução de dimensionalidade ou a compressão de dados;
- *Ordenação Topológica*: ao realizar o mapeamento não-linear dos vetores de entrada para o arranjo de neurônios da rede, o algoritmo do SOM tenta preservar ao máximo a topologia do espaço original, ou seja, procura fazer com que neurônios vizinhos no espaço de saída apresentem vetores de pesos que representem padrões vizinhos no espaço de entrada;
- *Casamento de Densidade*: o mapeamento efetuado pelo SOM reflete a distribuição de probabilidade dos dados no espaço de entrada original. Regiões do espaço de entrada de onde os vetores de amostra $\boldsymbol{x}_i$ são retirados, com uma alta probabilidade de ocorrência, são mapeadas para domínios maiores no espaço de saída e, portanto, com melhor resolução que regiões no espaço de entrada de onde vetores de amostra $\boldsymbol{x}_i$ são retirados com uma baixa probabilidade de ocorrência.

Visualização de Resultados no SOM

Uma forma comumente utilizada para visualizar os resultados obtidos em um processo de análise de agrupamentos que utiliza o algoritmo SOM é através da matriz de distâncias unificadas [Ultsch, 1993], ou matriz-U, como é mais conhecida. A matriz-U representa distâncias entre os neurônios vizinhos do vetor de referência através de uma escala de níveis de cinza, de forma que neurônios próximos (semelhantes entre si) são representados por cores claras, enquanto neurônios distantes são representados por cores escuras. Assim, é possível identificar os grupos detectados pelo algoritmo analisando as regiões adjacentes que possuem tonalidades próximas.

A matriz-U é uma grade com $(2.l - 1)$ linhas e $(2.c - 1)$ colunas, onde l é o número de linhas e c é o número de colunas da camada de saída do SOM. O valor de cada elemento uh_i da matriz-U representa a distância média entre o neurônio $\boldsymbol{w}_i$ e seus vizinhos

imediatos, representados por $N(i)$. Cada elemento da matriz-U é calculado da seguinte forma:

$$uh_i = \frac{1}{n}\sum_j d(\boldsymbol{w}_i, \boldsymbol{w}_j), j \in N(i), n = |N(i)| \tag{2.15}$$

onde $\boldsymbol{w}_i$ e $\boldsymbol{w}_j$ representam neurônios da camada de saída, $N(i)$ representa a vizinhança de $\boldsymbol{w}_i$ e $|N(i)|$ representa o número de elementos vizinhos a $\boldsymbol{w}_i$.

Posteriormente, novas formas de visualização de resultados a partir de derivações da matriz-U foram propostas, entre elas a matriz-P e a matriz-U* [Ultsch, 2003] [Ultsch, 2005]. A matriz-P é uma estrutura de visualização semelhante à matriz-U. No entanto, enquanto a matriz-U é calculada com base na distância entre os elementos da grade, a matriz-P é calculada a partir da densidade existente no conjunto de entrada associada a cada elemento $\boldsymbol{w}_i$ da camada de saída. Assim, a visualização dos seus resultados da matriz-P indica a densidade de elementos do conjunto de entrada associados a cada região do mapa.

Cada elemento ph_i da matriz-U é calculado da seguinte forma:

$$ph_i = \left\{\boldsymbol{x} \in \boldsymbol{X} \mid d(\boldsymbol{x}, \boldsymbol{w}_j) < r > 0, r \in \Re\right\} \tag{2.16}$$

onde $\boldsymbol{x}$ representa os elementos do conjunto de entrada que estão associados a um determinado neurônio $\boldsymbol{w}_j$ e cuja distância $d(\boldsymbol{x},\boldsymbol{w}_j)$ é menor que um limiar r.

Ultsch e Moerchen (2005) argumentam que a maioria das pesquisas envolvendo o algoritmo SOM utiliza mapas de tamanho pequeno e que a preservação da topologia em um mapa pequeno é prejudicada. Fenômenos emergentes, nos quais se inspiram os sistemas auto-organizáveis em geral e, consequentemente, os mapas auto-organizáveis, normalmente envolvem um grande número de indivíduos em suas populações, tipicamente alguns milhões [Johnson, 2003] [Ultsch e Moerchen, 2005]. Assim, propõem o Emergent SOM (ou simplesmente, ESOM), composto por mapas sem bordas, dispostos em formato toroidal, com pelo menos 4.000 neurônios e com número de linhas diferente do número de colunas. Segundo os autores, essas restrições previnem erros de topologia e provenientes da existência de bordas nos mapas quadrados e hexagonais, além de evitar distorções na visualização dos resultados.

A fim de melhorar a visualização do ESOM, Ultsch (2003) propõe a matriz-U*, uma extensão da matriz-U e matriz-P, que combina as vantagens presentes em ambas as abordagens e facilita a identificação de agrupamentos não-triviais. A matriz-U* foi desenvolvida para facilitar a análise de mapas com grande número de neurônios, como os considerados no ESOM e pode ser utilizada em tarefas de agrupamento, visualização e classificação [Ultsch e Moerchen, 2005].

Métodos de segmentação automática de mapas neurais, que possibilitam melhor interpretação e análise do SOM, foram propostos por vários autores, envolvendo diversas técnicas diferentes, como segmentação morfológica da matriz-U [Costa, 1999] [Costa e Andrade Netto, 1999] [Gonçalves *et al.*, 2005], segmentação utilizando particionamento de grafos [Costa e Andrade Netto, 2003], segmentação do SOM por métodos de

agrupamentos hierárquicos com conectividade restrita [Costa, 2005] [Gonçalves *et al.*, 2007] e preenchimento através de algoritmos de inundação (do inglês, *flood-fill*) [Opolon e Moutarde, 2004] [Moutarde e Ultsch, 2005].

Métodos hierárquicos também têm sido abordados para particionamento recursivo de bases de dados e aplicações como compressão de imagens [Costa e Andrade Netto, 2001a] [Costa e Andrade Netto, 2001b] [Costa *et al*., 2003]. Recentemente, Gonçalves *et al.* (2006) apresentaram uma otimização do método proposto por Costa e Andrade Netto (1999) incorporando índices de validação de agrupamentos *CDbw* [Halkidi e Vazirgiannis, 2002], que trata de maneira eficiente agrupamentos com formatos arbitrários. Extensões do método para uso com agrupamento hierárquico do mapa de Kohonen aplicado em classificação de imagens de satélite foram apresentadas em Gonçalves *et al.* (2007) e Gonçalves *et al.* (2008).

Em casos onde ocorrem perdas significativas de topologia no mapeamento de espaços de entrada de elevada dimensão para redes SOM, Costa e Andrade Netto (2007) desenvolveram métodos de segmentação de mapas com espaço de saída 3*D*, que possibilitam menor distorção e melhor análise.

Dentro do mesmo objetivo, dois algoritmos denominados ViSOM (Visualisation Induced SOM) e gViSOM (growing ViSOM) [Yin, 2002] [Yin, 2008] foram propostos buscando minimizar grandes distâncias existentes entre neurônios da camada de saída que possam prejudicar ou ocultar a visualização dos agrupamentos detectados. A principal característica do ViSOM é regular a distância entre neurônios existentes no mapa, de forma a preservar, sob as mesmas proporções, a topologia existente nos dados de entrada. O resultado é uma projeção mais suavizada, que permite capturar detalhes que não são facilmente visíveis apenas com a matriz-U e suas extensões.

2.5 Métricas de Avaliação dos Resultados

Diversas medidas têm sido propostas para avaliação de tarefas de análise de agrupamentos [Kuncheva, 2004] [Pölzlbauer, 2004] [Frei, 2006]. A seguir, são apresentadas algumas métricas mais comumente utilizadas em processo de análise de agrupamentos com os algoritmos SOM e *k*-médias. A primeira delas é bastante utilizada em quantização vetorial. As duas seguintes são específicas para o algoritmo SOM e as demais são genéricas para tarefas de análise de agrupamentos.

Erro de Quantização Médio do Mapa

O erro de quantização médio de um mapa auto-organizável treinado mede a diferença média entre cada elemento do conjunto de entrada o neurônio mais semelhante a ele na camada de saída. O erro de quantização médio do mapa avalia a qualidade da quantização vetorial obtida. Matematicamente, é definido como:

$$qe = \frac{1}{n}\sum_{i=1}^{n}\left|\boldsymbol{x}_i - \boldsymbol{w}_{\xi}\right| \tag{2.17}$$

onde N representa o tamanho do conjunto de entrada, $\boldsymbol{x}_i$ representa um vetor do conjunto de entrada, $\boldsymbol{w}_\xi$ representa o seu *bmu* correspondente na camada de saída e $|\,.\,|$ representa a distância euclidiana.

Erro Topológico do Mapa

O erro topológico (*te*) de um mapa auto-organizável treinado é definido como sendo o percentual de vetores do conjunto de entrada para os quais o primeiro e segundo *bmu* associados no mapa não são neurônios adjacentes. O erro topológico mede a preservação da topologia do mapa, sendo matematicamente descrito da seguinte forma:

$$te = \frac{1}{n}\sum_{i=1}^{n} u(\boldsymbol{x}_i) \tag{2.18}$$

onde N representa o tamanho do conjunto de entrada, $\boldsymbol{x}_i$ representa um vetor do conjunto de entrada e $u(\boldsymbol{x})$ representa um elemento $\boldsymbol{x}_i$ do conjunto de entrada que possui o primeiro e segundo *bmu* vizinhos.

Medida de Distorção do Mapa

A medida de distorção (*adm*), calculada sobre um mapa auto-organizável treinado, indica a distorção sofrida pelo mapa durante o processo de treinamento. Conforme citado por Pölzlbauer (2004), a medida de distorção pode ser interpretada como a função energia do SOM, para um valor fixo de vizinhança, sendo dada por:

$$adm = \sum_i \sum_j h(bmu(i), j)(\boldsymbol{w}_\xi - \boldsymbol{x}_i)^2 \tag{2.19}$$

onde $\boldsymbol{x}_i$ representa um vetor do conjunto de entrada, $\boldsymbol{w}_\xi$ representa o seu *bmu* correspondente na camada de saída e $h(.)$ é a função vizinhança.

Índice de Davies-Bouldin

O índice de Davies-Bouldin é bastante utilizado para validar tarefas de agrupamento [Davies e Bouldin, 1979], sendo definido como descrito a seguir. Seja $C = \{C_1, C_2, ..., C_K\}$ uma partição do conjunto de entrada **X**. O índice de Davies-Bouldin para a partição C_i é calculado da seguinte forma:

$$db(i) = \frac{1}{K}\sum_{i=1}^{K} R_i \tag{2.20}$$

onde K é o número de partições existentes e R_i é similaridade relativa entre o agrupamento C_i e os demais agrupamentos. A similaridade R_{ij} entre os agrupamentos C_i e C_j é calculada da seguinte forma:

$$R_{ij} = \max_{i \neq j} \frac{\left(e_i / \sqrt{n_i}\right) + \left(e_j / \sqrt{n_j}\right)}{d_{ij}} \tag{2.21}$$

onde d_{ij} é a distância entre o elemento médio (centroide) dos agrupamentos i e j, n_k é o número de elementos do agrupamento k e e_k é a distância média quadrática entre os elementos no agrupamento k e o seu centroide, dada por:

$$e_k = \frac{1}{n_k} \sum_{i=1}^{n_k} (\boldsymbol{x}_i - \boldsymbol{w}_\xi)^2 \tag{2.22}$$

onde n_k é o número de elementos do agrupamento k, $\boldsymbol{x}_i$ é um elemento do agrupamento k e $\boldsymbol{w}_\xi$ representa o centroide do agrupamento k.

Diversos outros índices de validação de agrupamentos são propostos na literatura [Salazar *et al*., 2001] [Salazar *et al.*, 2002] [Shim *et al.*, 2005] [Kim e Ramakrishna, 2005] [Gonçalves *et al.*, 2006] [Saitta *et al.*, 2007]. Por razões de simplicidade, foram descritos apenas os que serão utilizados ao longo deste trabalho.

2.6 Resumo do Capítulo

Este capítulo apresentou a análise de agrupamentos, uma técnica estatística normalmente aplicada na descrição de dados sobre os quais se possui pouca eu nenhuma informação sobre a distribuição dos mesmos no espaço dimensional. Inicialmente, foi apresentada uma breve descrição matemática do processo, seguida de uma descrição das etapas que a compõem.

Em seguida, foram apresentadas as principais medidas de similaridade e os algoritmos mais utilizados em tarefas de análise de agrupamento. Ao final, foram descritas algumas métricas utilizadas para avaliar os resultados obtidos. No próximo capítulo será abordada uma aplicação prática para a arquitetura proposta neste trabalho no ramo de negócios: a segmentação de mercado, que utiliza a técnica de análise de agrupamentos para a seleção e identificação de consumidores com perfis semelhantes.

Capítulo 3
Segmentação de Mercado

Marketing é a atividade humana destinada a satisfazer necessidades e desejos do consumidor através dos processos de troca [Kotler e Keller, 2006]. O marketing possui dois objetivos distintos: o primeiro, mais filosófico, é o de buscar conhecer bem o mercado para atender melhor os seus consumidores com informações, produtos e serviços cada vez mais adequados. O segundo, mais prático, sugere que com mais conhecimento acerca do mercado, também é possível traçar estratégias mais eficientes que auxiliem no planejamento e tomada de decisões [Paço, 2007].

Atualmente, é praticamente impossível para uma empresa atender a todos os consumidores de um determinado mercado. Por isso, uma alternativa bastante viável é dividir o mercado em segmentos e atuar sobre um grupo particular de clientes, oferecendo produtos e/ou serviços especificamente desenvolvidos a partir do seu perfil [Las Casas, 2006]. Essa abordagem é denominada *segmentação de mercado* e permite reduzir custos, uma vez que os esforços podem ser direcionados apenas ao grupo selecionado.

Portanto, a segmentação de mercado é uma aplicação direta para a técnica de análise de agrupamentos, objetivando identificar consumidores semelhantes e agrupá-los de acordo com atributos, padrões de comportamento e perfis de compra. Este capítulo apresenta os tipos de segmentação e as principais variáveis consideradas em um processo de segmentação de mercado, analisa os desafios da segmentação de mercado em ambientes com bases de dados distribuídas e discute soluções para problemas relacionados à segurança e privacidade de dados distribuídos, além de arrolar diversos trabalhos relacionados aos assuntos discutidos.

3.1 Segmentação de Mercado

De acordo com Kotler e Keller (2006), um segmento de mercado consiste de um grande grupo de indivíduos que pode ser identificado a partir de características individuais, tais como: sexo, idade, localização geográfica, preferências, atitudes, hábitos e poder de compra, etc.

Segmentação de mercado pode ser definida como a ação de identificar e selecionar grupos distintos de consumidores que podem exigir produtos e/ou compostos de marketing semelhantes. Uma vez realizada a seleção dos indivíduos em grupos, uma

segunda etapa do processo prevê a identificação de características existentes em cada um dos grupos, visando o entendimento dos seus hábitos de consumo e possibilitando conhecê-los melhor [Punj e Stewart, 1983] [Paço, 2003].

Kotler e Keller (2006) sugerem efetuar segmentação de mercado observando os seguintes aspectos: *segmentação geográfica*; *segmentação demográfica*; *segmentação psicográfica* e *segmentação comportamental*. Cada um desses aspectos inclui um determinado subconjunto das variáveis existentes na base de dados e são chamados de *bases para a segmentação de mercado*. A Tabela 3.1 apresenta as variáveis mais comumente utilizadas em cada um dos aspectos.

Tabela 3.1: Bases para a segmentação de mercado

Aspecto	Variáveis Comumente Utilizadas
Segmentação Geográfica	país; região (N, NE, CO, SE, S); estado; cidade; bairro; porte do município (pequeno, médio, grande); densidade populacional; zona (rural, urbana, periférica); etc.
Segmentação Demográfica	idade; sexo; raça; nacionalidade; renda; classe social (A, B, C, D, E); grau de instrução; estado civil; tamanho da família; profissão; ocupação; religião; etc.
Segmentação Psicográfica	personalidade (autoritário, ambicioso, compulsivo, decidido, indeciso, etc.); estilo de vida (reservado, sociável, intelectual, etc.); atitude (formador de opinião, influenciável, indiferente, etc.); etc.
Segmentação Comportamental	influência de compra; hábitos de compra; fidelidade à marca (nenhuma, baixa, média, alta); tempo de relacionamento; benefícios que procura no produto (qualidade, preço, durabilidade); etc.

Alguns autores propõem outras bases para a segmentação, porém utilizando normalmente o mesmo conjunto de variáveis. Las Casas (2006) acrescenta a base aspectos relacionados com o produto, que inclui variáveis descritas como comportamentais por Kotler e Keller (2006). Schiffman e Kanuk (2000) propõem o uso de sete bases: geográfica; demográfica; psicográfica; sociocultural; relacionada com o uso; por uso-situação e por benefício.

Añaña *et al*. (2008) analisam a tarefa de segmentação de mercado a partir de dois métodos distintos: segmentação *a priori* e segmentação *a posteriori*. No primeiro método, o pesquisador determina um ou mais conjuntos de variáveis de interesse, levando em consideração as bases de segmentação. Depois de concluída a segmentação, associa cada segmento de indivíduos com algum grupo previamente rotulado.

Na segmentação *a posteriori*, o pesquisador seleciona um conjunto de variáveis que, eventualmente, pertence a diferentes bases de segmentação. Depois de concluída a segmentação, o pesquisador irá rotular os segmentos por meio de análises estatísticas, que passam a ter mais importância que a escolha das bases de segmentação utilizadas.

Essa segunda abordagem parece ser sempre mais adequada, uma vez que é possível que existam indivíduos que não pertencem a nenhum dos grupos previamente definidos no modelo *a priori*. No entanto, a escolha das variáveis é um fator crítico e a simples inclusão ou exclusão de uma única variável pode representar uma diferença considerável na segmentação final.

Serrentino (2006) apresenta o seguinte exemplo que ilustra bem essa situação: suponha duas mulheres que possuam perfis bem parecidos, casadas, residentes na mesma rua, mesma faixa etária e classe social e uma estrutura familiar parecida. Além disso, suponha que ambas trabalham na mesma empresa, em funções similares. Considere, entretanto, elas diferem em relação a um único atributo, a religião: uma delas é evangélica e a outra católica. Esse simples atributo que as diferencia pode influenciar significativamente nos hábitos de consumo de ambas, de forma que respondam de forma diferente aos mesmos estímulos veiculados em uma campanha de marketing.

Outros fatores, principalmente os relacionados a aspectos éticos, de responsabilidade social e proteção ao meio ambiente têm influenciado cada vez mais nas decisões de compra do consumidor [Burgaleta, 1995] [Daza, 2005] [León, 2008]. Por isso, esses fatores vêm sendo cada vez mais explorados competitivamente pelas empresas, tanto pela cobrança por parte do consumidor, que tem se mostrado mais exigente, como pela necessidade de cumprimento das exigências legais. Muito embora, ainda exista um número significativo de empresas que utiliza a questão ambiental apenas para projetar seus produtos, com poucas ações significativas que, de fato, venham a reduzir o impacto ambiental por ela causado [Izaguirre *et al*., 2007].

Assim, a análise posterior dos agrupamentos detectados durante o processo de segmentação ou propostos *a priori* é extremamente importante em determinados mercados. Principalmente nos casos em que os produtos e/ou serviços oferecidos pela empresa dependem de matéria-prima extraída diretamente da natureza (por exemplo, madeira, minerais, etc.) ou o processo de fabricação produz algum tipo de impacto direto no meio ambiente (produtos químicos, efluentes, etc.) [Burgaleta, 1992].

3.2 Marketing de Segmento Versus Marketing Individual

Se por um lado as empresas tentam segmentar o mercado em busca de encontrar grupos de clientes com hábitos de consumo semelhantes, por outro lado tais clientes estão cada vez mais exigentes. Cada indivíduo possui suas particularidades e especificidades, por isso os clientes atuais buscam soluções cada vez mais individualizadas que atentam às suas necessidades. Em alguns segmentos, como no mercado de moda e acessórios, a exclusividade é vista como um importante diferencial em relação à concorrência, onde os clientes buscam cada vez mais peças únicas, desenvolvidas sob medida para atender ao seu estilo pessoal.

O principal desafio para as empresas atuais é, portanto, sobreviver a essa dicotomia: reduzir custos operacionais a partir da utilização de técnicas como a

segmentação de mercado – que permite identificar nichos de consumidores propensos a absorver os seus produtos e/ou serviços – ao mesmo tempo em que necessitam possuir bases de dados com informações detalhadas de cada indivíduo – suficientes para criar ofertas personalizadas e atender às necessidades individuais de cada um. Essa última abordagem é denominada marketing individual, mas também é conhecida como marketing um-para-um, marketing de relacionamento ou marketing de banco de dados.

À primeira vista, o marketing de segmento e o marketing individual parecem ser contraditórios. De fato, as duas abordagens possuem objetivos distintos, enquanto a primeira permite identificar grupos de consumidores com perfis similares, a fim de criar compostos de marketing que possam atender a todo o grupo, a segunda busca identificar particularidades no perfil de cada cliente, a fim de criar ofertas únicas, totalmente personalizadas.

Entretanto, as duas estratégias podem e devem ser utilizadas conjuntamente, uma vez que nenhuma dela é capaz de atender a todas as necessidades das empresas. O marketing individual ainda é extremamente caro, por exigir um grande esforço na etapa de coleta de dados, e nem sempre possível de ser implementado. Além disso, é uma abordagem incompleta, cuja ênfase está na retenção de cliente e não na conquista de novos clientes e no alcance de mercado [Myers, 2000]. Por outro lado, em uma economia globalizada, os segmentos de mercado tornaram-se gigantescos, quase comparado às dimensões do marketing de massa.

Por exemplo, para um fabricante de automóveis, não é viável construir um modelo de veículo exclusivo para cada indivíduo. Da mesma forma, para uma indústria de perfumaria, seria praticamente impossível criar uma fragrância personalizada para cada cliente. No entanto, detalhes no acabamento interno do veículo ou na embalagem do perfume podem ser personalizados de acordo com o gosto do cliente, criando assim um produto adequado às suas necessidades gerais ou ao seu gosto pessoal. Assim, o marketing de segmento e o marketing individual podem atuar de forma complementar permitindo oferecer o produto mais adequado a cada cliente.

Outra opção para combinar ambas as abordagens é utilizar o marketing de segmento como ferramenta de venda, tanto na identificação de necessidades comuns a um grupo de clientes como na identificação de mercados-alvo e utilizar o marketing de relacionamento como ferramenta de pós-venda, já que o mesmo é direcionado à fidelização do cliente. À medida que se conhece melhor o cliente e os seus hábitos de consumo, é possível criar um relacionamento cada vez mais personalizado e duradouro.

A necessidade atual de criar ofertas individuais e personalizadas para atender às exigências de seus clientes obriga as empresas a buscar a utilização de métodos e técnicas que permitam entender esse mercado. Através da utilização massiva de recursos computacionais e ferramentas de análise de hábitos de compra, é possível fazer uso de técnicas de mineração de dados a fim de analisar grandes massas de dados e, através destas, conhecer melhor cada cliente.

3.3 Trabalhos Relacionados ao Tema

Inúmeras pesquisas têm sido desenvolvidas nos últimos anos em busca de métodos e técnicas que possibilitem realizar segmentação de mercado com melhor desempenho. Entre os métodos e técnicas mais utilizados estão: análise fatorial, análise de agrupamentos, análise de componentes principais, regressão logística e redes neurais. Steenkamp e Hofstede (2002) apresentam um levantamento dos métodos e técnicas mais utilizados na área em segmentação de mercados internacionais.

A utilização de redes neurais artificiais em tarefas de segmentação de mercado tem sido motivada pelo bom desempenho dos mapas auto-organizáveis em classificação automática de dados, redução de dimensionalidade e análise de agrupamentos. A seguir, são descritos diversos trabalhos que investigam a utilização de redes SOM em segmentação de mercado.

Vellido *et al*. (1999) investigam a utilização de redes neurais do tipo SOM na segmentação de mercado de consumidores que realizam compras através da Internet. A partir de dados obtidos através da aplicação de questionários *on-line*, uma rede SOM é utilizada para identificar padrões de comportamento e segmentar os denominados *e*-consumidores de acordo com critérios como propensão à realização de compras via Internet, rejeição à utilização esse canal, dificuldade em utilizar a tecnologia, entre outros [Vellido *et al*., 1999] [Vellido *et al*., 2002].

Medir a percepção do cliente em relação a quais atributos são mais determinantes no processo de decisão de compra é um problema bastante conhecido pelos profissionais de marketing de turismo e serviços de viagens, conforme avaliam Kim *et al*. (2003). Nesse artigo, os autores usaram um modelo de rede neural não supervisionada para avaliar a ponderação de diferentes atributos e descrever uma relação consumidor-produto que permite determinar quais *trade-offs* os viajantes mais velhos consideram quando decidem seus planos de viagem. O estudo considerou uma base com dados reais de turistas idosos da Austrália Ocidental.

A segmentação de mercados exclusivamente *on-line* também tem sido alvo de diversos trabalhos investigativos. Xiaoguang *et al*. (2007) propõem a aplicação de uma técnica conhecida como estimativa *fuzzy* para analisar, segmentar e reduzir riscos no mercado de serviços *on-line* chinês. Lee *et al*. (2005) utilizam uma abordagem baseada em duas camadas de redes SOM para analisar o mercado de jogos *on-line* chinês com o objetivo de identificar quem são os consumidores potenciais dentro desse mercado e quais suas principais motivações. O trabalho apresenta uma metodologia para identificar as variáveis que melhor segmentam o mercado para este tipo de consumidor.

Mais recentemente, Kim e Ahn (2008) propõem uma nova abordagem para realização de análise de agrupamentos baseado em algoritmos genéticos, ainda com o objetivo de segmentação de mercado de compras *on-line*. A pesquisa realizou o agrupamento com o algoritmo *k*-médias, porém os valores iniciais dos centroides foram otimizados a partir de um algoritmo genético. A técnica, denominada GA *k-means*, foi

aplicada a uma base de dados real de segmentação de mercado de compras on-line para construção de uma ferramenta de pré-processamento para sistemas de recomendação. Os resultados obtidos foram comparados com abordagens simples dos algoritmos *k*-médias e SOM, demostrando que o agrupamento GA *k-means* pode melhorar o desempenho da segmentação em comparação com outros algoritmos típicos de agrupamento.

Em instituições financeiras, tais como bancos e administradoras de cartão de crédito, a segmentação de mercado é uma ferramenta de extrema importância para conhecer melhor o perfil dos clientes, uma vez que tais empresas costumam possuir bases de dados muito volumosas e de alta dimensionalidade.

Hsieh (2004) descreve um modelo de segmentação de mercado aplicado a clientes de cartão de crédito de instituições financeiras. O modelo segmenta a base de clientes em três grandes grupos, utilizando uma rede SOM que analisa atributos relacionados ao perfil de uso do cartão de crédito. Em seguida, o algoritmo Apriori é utilizado para criar um sistema de regras de associação capaz de identificar o perfil dos clientes mais rentáveis para a instituição, com base nas características demográficas e geográficas dos grupos.

Zakrzewska e Murlewski (2005) comparam três algoritmos de análise de agrupamento para segmentação de uma base de dados de clientes de um banco, levando em consideração alguns critérios como eficácia, escalabilidade e capacidade de detectar exceções. Os resultados mostram que os algoritmos analisados possuem vantagens e limitações, mas o *k*-médias mostrou-se mais eficiente em bases de dados muito volumosas. Lin e Yang (2006) analisam o comportamento do consumidor feminino diante das ofertas de serviços bancários e procuram identificar fatores-chave utilizados por essas clientes na tomada de decisões. Para isso, consideram características demográficas e psicográficas relacionadas ao estilo de vida de cada cliente.

A combinação de múltiplas técnicas de segmentação de mercado também tem sido investigada na tentativa de otimizar o desempenho dos algoritmos de análise de agrupamentos aplicados sobre bases de dados reais. Kuo *et al.* (2006) apresentam um modelo para segmentação de mercado baseado na integração de mapas auto-organizáveis com o algoritmo *k*-médias genético. O método usa os mapas auto-organizáveis para determinar o número de agrupamentos e, em seguida, aplicam o algoritmo *k*-médias para encontrar a solução final. Segundo os autores, a combinação SOM e *k*-médias apresenta melhores resultados que sua utilização em separado.

Chi *et al*. (2000) propõem uma abordagem denominada FSOM (*fuzzy self-organizing maps*) que combina diversas outras abordagens, como lógica difusa (*fuzzy*), *k*-médias e mapas auto-organizáveis para tornar o processo de análise de agrupamentos menos dependente do usuário. Os autores propõem ainda usar o relacionamento entre os dados de entrada e saída obtidos com o FSOM para treinar uma rede neural usando o algoritmo *backpropagation*, com o objetivo de criar um sistema de suporte à decisão.

Lee e Park (2005) combinam diferentes ferramentas de inteligência comercial para criar um sistema de suporte à decisão que analisa e identifica perfis de consumidores

lucrativos para uma empresa. A abordagem utiliza uma metodologia de avaliação de eficiência denominada Análise por Envoltória de Dados (*Data Envelopment Analysis - DEA*) para medir o quanto nível de lucratividade do consumidor para a empresa e combina os algoritmos SOM e C4.5 para segmentar a base de clientes a partir de dados sócio-demográficos.

Embora os mapas auto-organizáveis tradicionais sejam frequentemente empregados em segmentação de mercado, algumas modificações têm sido propostas para facilitar sua utilização. Kiang (2001) propõe uma extensão das redes SOM para atividades de segmentação de mercado baseada na segmentação do mapa já treinado através de um algoritmo de continuidade restrita (do inglês, *contiguity-constrained*), conforme proposto em [Murtagh, 1995].

Kiang e Kumar (2001) analisam a utilização de mapas auto-organizáveis estendidos como uma alternativa mais eficiente que as técnicas tradicionais de análise fatorial em aplicações de mineração de dados [Kiang, 2001] [Kiang *et al*., 2004]. Posteriormente, os mesmos autores realizam uma comparação entre os algoritmos *k*-médias e o SOM estendido, mostrando as vantagens dessa abordagem principalmente sobre agrupamento de dados com distribuição assimétrica [Kiang e Kumar, 2004].

Ultsch (2002) utilizou uma combinação de mapas auto organizáveis emergentes e métodos de visualização de redes SOM baseados na U-Matrix para identificar estruturas multidimensionais em bases de dados de clientes de telefonia, tendo como finalidade a previsão e prevenção de rotatividade nos mercados de telefonia móvel da Europa Central. O conhecimento obtido pela abordagem proposta possibilitou uma compreensão mais clara acerca do perfil dos clientes e dos motivos relacionados à troca de operadora e, assim, construir um classificador de predição de rotatividade bastante eficaz para o gerenciamento de relacionamento com o cliente.

Em outro trabalho na área de telefonia, Kiang *et al*. (2006) apresentam uma aplicação dos mapas auto-organizáveis estendidos na segmentação de mercado de uma base de dados da empresa AT&T (*American Telephone and Telegraph Company*), comparando os resultados obtidos com uma abordagem em duas etapas que combina análise fatorial e *k*-médias. O artigo mostra que o modelo proposto fornece melhores resultados que as abordagens estatísticas tradicionais que reduzem a dimensionalidade com análise fatorial e utilizam alguma técnica de análise de agrupamentos (no caso, *k*-médias) para detectar os segmentos de clientes.

D'Urso e De Giovanni (2008) também utilizam mapas auto-organizáveis para segmentar mercado no setor de telecomunicações, mas consideram bases de dados temporais para esta finalidade, que contém dados dinâmicos relativos a tráfico, infraestrutura e capacidade de transmissão. Os resultados obtidos foram comparados com outra técnica de agrupamento denominada DTA (*Dynamic Tandem Analysis*), que combina análise fatorial dinâmica e análise de agrupamentos. Os resultados mostram que

cada uma das abordagens possui vantagens sobre a outra e devem ser utilizadas de forma complementar.

Kiang *et al.* (2005) e Kiang *et al.* (2007) analisam e discutem questões relacionadas à influência do tamanho da amostra utilizada no treinamento de algoritmos de análise de agrupamento. São analisadas duas abordagens, o SOM estendido e uma combinação de análise fatorial e *k*-médias utilizada em trabalhos anteriores. Nos dois artigos é possível verificar que a abordagem que utiliza o SOM consegue obter melhores resultados independentemente do tamanho da amostra utilizado.

Em um trabalho mais recente, Kiang e Fisher (2008) utilizam mapas auto-organizáveis para segmentar e visualizar as 79 mais bem classificadas escolas americanas que oferecem programas de MBA (*Master of Business Administration*), com o objetivo de ajudar alunos a escolher a instituição que melhor atende às suas necessidades. As instituições foram analisadas a partir de oito variáveis obtidas junto ao ranking anual da *US News and World Report*, que incluem uma média ponderada de vários fatores, como custos, qualidade da instituição, salário inicial, entre outros.

Nesse artigo, as instituições foram segmentadas em quatro agrupamentos (elite, prestígio, alto valor e acessível) através do SOM estendido [Kiang, 2001]. Os resultados foram comparados com o *k*-médias tradicional e com uma combinação de análise fatorial e *k*-médias utilizada em trabalhos anteriores. Além disso, foram utilizados índices estatísticos, como validação cruzada e o índice Rand. O artigo demonstra que o SOM estendido oferece vantagens sobre os demais métodos, sendo a principal delas a exibição dos resultados em um mapa bidimensional.

Añaña *et al.* (2008) analisam membros de comunidades virtuais em relação ao consumo de cerveja. São comparadas as técnicas de regressão logística e redes neurais, buscando identificar vantagens e limitações de cada uma delas na identificação de segmentos de consumidores.

O mercado de multimídia vem crescendo bastante nos últimos anos, fortemente impulsionado pelo surgimento de novas tecnologias de transmissão de dados via Internet. Uma das grandes novidades é o fornecimento de multimídia sob demanda (MOD – *multimedia on demand*) via linhas digitais de alta velocidade, que substitui com vantagens as transmissões a cabo e inclui diversos serviços adicionais, como vídeo sob demanda, interatividade, jogos *on-line*, etc.

Interessados em analisar o comportamento do consumidor frente ao fornecimento de multimídia sob demanda, Hung e Tsai (2008) propõem uma ferramenta visual para segmentação de mercado baseada em redes SOM, que permite explorar o relacionamento entre os fornecedores do serviço e seus clientes e oferecer suporte à tomada de decisões.

A ferramenta, denominada modelo de segmentação auto-organizável hierárquico (HSOS – *hierarchical self-organizing segmentation*), possibilita analisar uma base de dados de MOD e segmentar os clientes de acordo com o perfil de consumo através de uma estrutura hierárquica de redes SOM. A base de dados utilizada inclui dados

demográficos, psicográficos e atitudes de consumo de consumidores de MOD em Taiwan. Os resultados são comparados com um método aglomerativo e com outra abordagem hierárquica, o GHSOM (*Growing Hierarchical Self-Organizing Map*), mostrando as vantagens da abordagem no acompanhamento visual dos resultados.

Este trabalho apresenta uma arquitetura para segmentação de mercado baseada na segmentação de subconjuntos dos atributos através de múltiplos mapas auto-organizáveis, seguida da combinação dos resultados parciais. Sua principal contribuição é demonstrar que é possível realizar a segmentação global dos dados a partir da combinação de segmentações parciais e que os resultados obtidos nas análises parciais enriquecem ainda mais o processo de análise, justificando sua utilização.

3.4 Resumo do Capítulo

Este capítulo apresentou a segmentação de mercado como uma das principais aplicações para a tarefa de análise de agrupamentos. Inicialmente foram descritos os objetivos da segmentação de mercado como atividade/ferramenta de análise e suporte à decisão em administração e marketing. Em seguida, foram apresentadas as principais bases sobre as quais a segmentação de mercado é realizada e os atributos as técnicas mais utilizadas neste processo.

Posteriormente, o marketing de segmento foi confrontado com o marketing individual, tão difundido e proclamado nos dias atuais. Foi discutida a importância das duas abordagens, mostrando-se que ao invés de contraditórias, elas podem ser complementares. Finalmente, foi apresentada uma ampla revisão bibliográfica de trabalhos relacionados à utilização de mapas auto-organizáveis em tarefas de segmentação de mercado.

No próximo capítulo serão abordados os comitês de agrupamento, sistemas que combinam múltiplas soluções em análise de agrupamentos para produzir resultados mais robustos e confiáveis.

Capítulo 4
Comitês de Agrupamento

Comitês de agrupamento (do inglês, *cluster ensembles*) podem ser brevemente definidos como uma combinação de duas ou mais soluções provenientes da aplicação de diferentes algoritmos de agrupamento ou variações de um mesmo algoritmo sobre um conjunto de dados, ou ainda, sobre subconjuntos deste.

A combinação de vários algoritmos de agrupamento tem como objetivo produzir resultados mais robustos e confiáveis que a utilização de algoritmos individuais, por isso os comitês de agrupamento têm sido propostos em diversas aplicações que envolvem agrupamento e classificação de dados.

Este capítulo apresenta uma revisão bibliográfica sobre trabalhos que abordam o uso de comitês de classificadores e de agrupamento. Inicialmente, é destacada a tarefa de análise de agrupamento sobre dados geograficamente distribuídos. Depois, são discutidas diferentes alternativas para particionamento de dados e questões relacionadas à segurança e privacidade em agrupamento de dados distribuídos. No final, são apresentadas algumas técnicas de fusão de informações utilizadas na literatura para combinar resultados provenientes de múltiplas soluções de agrupamento.

4.1 Bases de Dados Geograficamente Distribuídas

A busca por algoritmos que realizem mineração de dados de forma distribuída não é recente, tendo sido impulsionada principalmente pelo surgimento dos primeiros sistemas gerenciadores de banco de dados distribuídos e da necessidade de analisar esses dados na forma em que eles se encontravam dispersos [DeWitt e Gray, 1992] [Souza, 1998].

Existe uma série de limitações que dificultam a utilização de técnicas tradicionais de mineração de dados sobre bases de dados distribuídas. A abordagem comumente utilizada é a reunião de todas as bases distribuídas em uma unidade central, seguida da aplicação dos algoritmos. Nesses casos é importante levar em consideração algumas questões, como a possibilidade da existência de dados similares com nomes e formatos diferentes, diferenças nas estruturas que armazenam os dados e conflitos entre os dados de uma base e outra [Zhang *et al*., 2003]. Além disso, a unificação de todos os registros em um único banco de dados pode levar a perda de informações significativas, uma vez

que valores estatisticamente interessantes em um contexto local podem passar despercebidos quando reunidos a outros de maior volume.

Além disso, a integração de diversas bases de dados em um único local não é aconselhada quando se trata de bases de dados muito volumosas. Se uma organização possui grandes bases de dados dispersas e precisa reunir todos os dados a fim de aplicar sobre elas os algoritmos de mineração de dados, esse processo pode exigir grandes transferências de dados, o que poderá ser lento e dispendioso [Forman e Zhang, 2000]. Ademais, qualquer mudança que ocorra nos dados distribuídos, como por exemplo, a inclusão de novas informações ou alterações daquelas já existentes terá que ser atualizada junto à base de dados central. Isso exige uma política complexa de atualização de dados, com sobrecarga de transferência de informações dentro do sistema.

Segurança e privacidade dos dados estão entre os principais fatores que motivam a criação e manutenção de bases de dados distribuídas [Lam *et al*., 2004]. Dessa forma, um argumento adicional para que algumas organizações mantenham suas bases de dados geograficamente distribuídas baseia-se na justificativa de aumentar a segurança das suas informações, pois se por acaso uma das políticas de segurança falha, o invasor tem acesso apenas a uma parte das informações existentes. Questões relacionadas à segurança e privacidade dos dados serão discutidas adiante, na seção 4.4.

Em função de todos esses problemas relacionados à integração de bases de dados, no final dos anos 90 começaram a surgir várias pesquisas sobre algoritmos para efetuar mineração de dados de forma distribuída, conforme pode ser visto em Bhaduri *et al*. (2006), que mantêm uma bibliografia atualizada sobre o tema[1].

4.2 Comitês de Classificação e Agrupamento

A combinação do resultado de diversos métodos de agrupamento, formando um comitê de agrupamento, surgiu como uma extensão direta dos sistemas que utilizam múltiplos classificadores [Kuncheva, 2004]. A utilização de sistemas com múltiplos classificadores, baseados na combinação dos resultados de diferentes algoritmos de classificação, tem sido proposta como um método para desenvolvimento de sistemas de classificação de alta performance com aplicações na área de reconhecimento de padrões [Roli *et al*., 2001].

Estudos teóricos e práticos confirmam que diferentes tipos de dados requerem diferentes tipos de classificadores [Ho, 2000], o que, pelo menos em tese, justifica a utilização de comitês. Entretanto, longe de ser um consenso, a utilização de sistemas com múltiplos classificadores e de comitês de agrupamento é questionada por diversos autores, tanto por exigir um maior esforço computacional, como por requerer a utilização de complicados mecanismos de combinação de resultados [Kuncheva, 2003].

[1] Disponível em: <http://www.cs.umbc.edu/~hillol/DDMBIB/>

Roli *et al*. (2001) argumentam que o crescente interesse em sistemas com múltiplos classificadores é proveniente das dificuldades em se decidir qual o melhor classificador individual para um determinado problema. Os autores analisam e comparam seis métodos para projetar sistemas com múltiplos classificadores e concluem que, embora tais métodos apresentem características interessantes, nenhum deles é capaz de garantir um projeto ideal de um sistema com múltiplos classificadores.

Ho (2002) critica os sistemas com múltiplos classificadores argumentando que ao invés de concentrar esforços em procurar o melhor conjunto de atributos e o melhor classificador, o problema passa a ser o de buscar o melhor conjunto de classificadores e o melhor método para combiná-los. Argumenta ainda que, em seguida, o desafio passa a ser o de encontrar o melhor conjunto de métodos de combinação de resultados e a melhor forma de usá-los. Assim, o foco do problema é esquecido e, cada vez mais, o desafio passa a ser a utilização de teorias e esquemas de combinação mais complicados.

Strehl (2002) afirma ser amplamente reconhecida a ideia de que a combinação de múltiplos classificadores ou de múltiplos modelos de regressão possa fornecer resultados superiores se comparados a um único modelo. Entretanto, alerta que não existem abordagens reconhecidamente eficientes para combinar múltiplos algoritmos não-hierárquicos de agrupamento. Nesse trabalho, o autor propõe uma solução para esse problema através de um *framework* para segmentação de consumidores a partir de dados comportamentais.

Apesar de todas as dificuldades relatadas, tanto os sistemas com múltiplos classificadores quanto os comitês de agrupamento vêm sendo cada vez mais utilizados. Zhao *et al*. (2005) apresentam uma boa revisão da área, onde relatam diversas aplicações para comitês de classificadores baseados em redes neurais, que incluem reconhecimento de padrões, diagnósticos de doenças e tarefas de classificação. Oza e Tumer (2008) fazem o mesmo em um trabalho mais recente, onde apresentam diversas aplicações reais onde a utilização de comitês de classificadores tem obtido maior sucesso em relação ao uso de classificadores individuais, incluindo sensoriamento remoto, reconhecimento de padrões e medicina. Fern (2008) analisa como combinar várias soluções disponíveis para formar um comitê de agrupamento mais eficiente, baseado em dois fatores muito importantes no desempenho de um comitê: a qualidade e a diversidade de soluções.

Leisch (1998), um dos pioneiros na área de comitês de agrupamento, introduziu um algoritmo denominado *bagged clustering*, que executa múltiplas instâncias do algoritmo *k*-médias na tentativa de obter uma estabilidade nos resultados e combina os resultados parciais utilizando um método de particionamento hierárquico.

Em outro trabalho introdutório sobre análise de agrupamentos distribuída, Forman e Zhang (2000) apresentam uma técnica que paraleliza múltiplos algoritmos baseados em centroides, como *k*-médias e EM (*expectation maximization*) a fim de obter maior eficiência no processo de mineração de múltiplas bases de dados distribuídas. Os autores reforçam a necessidade de preocupação em relação a reduzir sobrecarga de comunicação

entre as bases, diminuir o tempo de processamento e minimizar a necessidade de máquinas poderosas e com capacidades de armazenamento estendidas.

Kargupta *et al.* (2001) destacam a ausência de algoritmos que efetuem análise de agrupamentos em conjuntos de dados heterogêneos utilizando PCA de forma distribuída e apresentam um algoritmo denominado CPCA (*Collective Principal Component Analysis*) para análise de agrupamentos de dados heterogêneos de alta dimensionalidade. Os autores também discutem a preocupação em reduzir as taxas de transferências de dados em um ambiente de dados distribuídos.

A definição de comitês de agrupamento apresentada na introdução deste capítulo é propositadamente genérica, de forma a incluir diversas possibilidades de utilização de algoritmos de agrupamento e combinação de resultados existentes na literatura. De fato, Kuncheva (2004) sugere quatro abordagens para a construção de sistemas de classificadores, que podem ser estendidas para a construção de comitês de agrupamento:

i) Aplicação de várias instâncias de um mesmo algoritmo sobre uma mesma base de dados, alterando-se os parâmetros de inicialização do algoritmo e combinando os resultados destes;
ii) Aplicação de diferentes algoritmos de agrupamento sobre uma mesma base de dados, com objetivo de analisar qual algoritmo obtém um melhor agrupamento dos dados;
iii) Aplicação de várias instâncias de um mesmo algoritmo de agrupamento sobre subconjuntos de amostras ligeiramente diferentes, obtidas com ou sem reposição;
iv) Aplicação de várias instâncias de um mesmo algoritmo de agrupamento sobre diferentes subconjuntos de atributos;

Haykin (2001) descreve as redes neurais como processadores maciçamente paralelamente distribuídos, o que sugere que o treinamento de um comitê de agrupamento baseado em redes neurais pode ser feito de forma distribuída [Vrusias *et al*., 2007]. Além disso, existem na literatura diversas pesquisas com o objetivo de abordar o treinamento de redes neurais em paralelo, particularmente de mapas auto-organizáveis [Yang e Ahuja, 1999] [Calvert e Guan, 2005] [Vin *et al.*, 2005].

Este tipo de treinamento gera inúmeros desafios, uma vez que, via de regra, algoritmos de redes neurais são não-determinísticos e baseados em um conjunto de parâmetros de inicialização e treinamento. Assim, como as redes neurais costumam ser bastante sensíveis aos parâmetros de inicialização, as escolhas realizadas durante o processo de treinamento terminam por influenciar diretamente nos resultados obtidos.

Algumas pesquisas na área exploram essa particularidade das redes neurais para construir comitês baseados na execução de um mesmo algoritmo com diferentes conjuntos de inicialização e treinamento. Dentro dessa abordagem, *bootstrap aggregating*, *baggind* e *boosting* são algumas das técnicas de treinamento que têm sido empregadas com relativo sucesso no treinamento de comitês, conforme descrito em

[Breiman, 1996] [Freund e Schapire, 1999] [Dietterich, 2000] [Bakker e Heskes, 2003] [Frossyniotis *et al*., 2004] [Vin *et al.*, 2005]. Embora tais técnicas tenham demonstrado a variação de possibilidades existente e os benefícios dessas abordagens, ficaram evidentes alguns problemas que precisam ser considerados quando se treinam comitês concorrentemente com subconjuntos distintos de entradas, como o custo computacional e os mecanismos de fusão de resultados.

A utilização de agrupamentos (*clusters*[2]) de computadores e de grades (*grids*) computacionais tem sido frequentemente considerada na realização de treinamento distribuído de diversos tipos de redes neurais, como redes perceptrons de múltiplas camadas (MLP), mapas auto-organizáveis (SOM) e redes de função de base radial (RBF) [Calvert e Guan, 2005]. Hämäläinen (2002) apresenta uma revisão sobre diversas implementações paralelas utilizando mapas auto-organizáveis.

Neagoe e Ropot (2001) apresentam um modelo de classificador neural, denominado mapas auto-organizáveis concorrentes (CSOM), que é formado por uma coleção de pequenas redes SOM. O modelo CSOM apresenta algumas diferenças conceituais em relação ao SOM tradicional, a mais significativa delas está no algoritmo de treinamento, que é supervisionado. O número de redes SOM utilizadas no modelo deve ser igual ao número de classes de espaço de saída. Para cada rede SOM individual, um subconjunto de treinamento específico é utilizado, de forma que a rede seja treinada para ser especializada em uma determinada classe do espaço de saída. Assim, ao final da fase de treinamento, cada SOM tornou-se especialista na classe que representa.

Durante a utilização do classificador, o mapa que apresentar o menor erro de quantização é declarado vencedor e seu índice é o índice da classe a qual o padrão pertence. Nos testes realizados com o modelo CSOM, os autores consideram três aplicações nas quais o modelo apresenta bons resultados: reconhecimento de face, reconhecimento de fala e classificação de imagens de satélite multiespectrais [Neagoe e Ropot, 2002] [Neagoe e Ropot, 2004].

Arroyave *et al*. (2002) apresentam uma implementação paralela de múltiplas redes SOM utilizando um *cluster Beowulf*, com aplicação na organização de documentos de texto. Nessa abordagem, um grande mapa auto-organizável é dividido em várias partes de mesmo tamanho e distribuído entre as máquinas do *cluster*. O treinamento também é realizado de forma distribuída, onde cada unidade escrava recebe cada um dos dados de entrada da unidade mestre e retorna o seu *bmu*, que é compartilhado com as demais em um processo cooperativo.

Vrusias *et al*. (2007) propõem o treinamento de mapas auto-organizáveis, de forma distribuída, usando uma grade computacional. Os autores sugerem o uso de uma arquitetura e uma metodologia para treinamento de um comitê de SOM distribuídos ao longo de uma grade computacional, onde são considerados: o número ideal de mapas no

[2] Usado neste contexto, o termo *cluster* significa um agrupamento de computadores que trabalham conjuntamente, como se fossem um único computador, para realização de uma tarefa.

comitê, o impacto dos diferentes tipos de dados usados no treinamento e o período mais apropriado para atualização dos pesos.

O treinamento prevê atualizações periódicas nos pesos dos mapas, onde resultados parciais de cada unidade são enviados para a unidade mestre no início de cada etapa de treinamento e esta se encarrega de efetuar uma média dos valores recebidos e reenviá-los para as unidades escravas. Como há muita interação entre as partes no decorrer do treinamento, o tempo gasto nessa operação pode ser alto, influenciando diretamente no tempo de treinamento do mapa. Portanto, segundo os autores, essa abordagem só obtém bons resultados em *clusters* dedicados.

Os autores realizaram uma série de experimentos e obtiveram algumas conclusões importantes, que podem ser estendidas a outros algoritmos de treinamento paralelo de redes SOM:

a) Se o tempo de latência dos membros do comitê, os ajustes de pesos periódicos e o tempo de sincronismo dos mapas forem muito pequenos, em relação ao tempo computacional de cada etapa de treinamento, a utilização de um comitê de SOM produz bons resultados em relação ao tempo de treinamento e precisão;
b) Nos testes realizados, o número ideal de mapas em um comitê situou-se entre 5 e 10 redes;
c) A escolha dos valores dos parâmetros utilizados no treinamento (taxa de aprendizagem e decremento) e a frequência com que são feitos os cálculos das médias dos mapas também são fatores de grande importância na redução do erro quadrático médio;
d) O comitê de SOM apresenta resultados bem superiores à medida que se aumenta a dimensão do conjunto de dados.

Georgakis *et al*. (2005) propõem a utilização de um comitê de mapas auto-organizáveis na tentativa de aumentar a performance na organização e recuperação de documentos. Vários mapas são treinados simultaneamente, com subconjuntos de dados ligeiramente diferentes. Na etapa posterior, os mapas são comparados e os neurônios dos membros do comitê são alinhados para formar o mapa final. Os neurônios mais similares de cada mapa são combinados, através de uma média aritmética de seus pesos sinápticos, para compor um novo neurônio do mapa final.

Durante o treinamento, são retiradas amostras aleatoriamente selecionadas a partir do conjunto de dados para alimentar cada um dos membros do comitê. O algoritmo é utilizado para segmentar um repositório de documentos em grupos, de acordo com o seu conteúdo semântico. Os experimentos realizados mostram que a performance do algoritmo proposto é superior ao SOM tradicional, em relação à precisão de recuperação de documentos com base em seu conteúdo semântico.

A aplicação de comitês de agrupamento sobre diferentes subconjuntos de atributos tem sido analisada principalmente em tarefas de segmentação de imagens. Picture SOM

(ou simplesmente, PicSOM) é uma arquitetura hierárquica na qual vários algoritmos e métodos podem ser aplicados em conjunto para recuperação de imagens baseado em conteúdo [Laaksonen *et al*., 1999] [Laaksonen *et al*., 2000] [Laaksonen *et al*., 2002].

Originalmente, a arquitetura PicSOM utiliza múltiplas instâncias do algoritmo TS-SOM, composto por árvores estruturadas de mapas auto-organizáveis, organizadas de forma hierárquica [Koikkalainen, 1994]. Cada TS-SOM é treinado com um conjunto diferente de características, tais como cor, textura ou forma. A arquitetura PicSOM é um exemplo de combinação de redes SOM cujo resultado é um sistema robusto para recuperação de imagens baseado em similaridade de conteúdo.

Georgakis e Li (2006) propõem uma modificação do PicSOM usando uma técnica denominada *bootstrapping* durante a etapa de treinamento. Essa técnica divide aleatoriamente o espaço de entrada em uma série de subconjuntos que são usados na etapa de treinamento do SOM. Em seguida, os mapas treinados são combinados em um único mapa, para formar o resultado final. Segundo os autores, a abordagem obtém resultados mais precisos que o PicSOM original.

Yu *et al*. (2007) apresentam uma proposta de uma arquitetura para segmentar imagens baseado em um comitê de algoritmos EM (*expectation maximization*). A arquitetura inicia extraindo informações de cor e textura da imagem, que são processadas em separado. Uma etapa posterior combina as regiões vizinhas segmentadas individualmente, considerando ainda informações relacionadas à posição dos *pixels* na imagem.

Jiang e Zhou (2004) apresentam outra proposta de uso de comitês de redes SOM para segmentação de imagens, baseado apenas em informações de cor e posição dos *pixels*. A abordagem proposta combina os resultados parciais através de um esquema de votação ponderada avaliado através do índice de informação mútua, que mede a semelhança entre as partições.

A maioria dos algoritmos de análise de agrupamentos lida apenas com dados numéricos, apesar de existirem algumas variações desses algoritmos especificamente desenvolvidos para tratar de dados categóricos. Nos casos de bases de dados com ambos os tipos de valores, alguns ajustes são necessários durante a etapa de pré-processamento, como por exemplo, a conversão de dados categóricos em dados binários mutuamente exclusivos. Essa conversão eleva ainda mais a dimensionalidade da base de dados, pois cria uma coluna adicional para cada possível valor do atributo.

Algumas abordagens alternativas para codificação de variáveis categóricas em numéricas são apresentadas na literatura. Rosario *et al*. (2004) propõem um método que analisa como determinar a ordem e o espaçamento entre variáveis nominais e como reduzir o número de valores distintos a considerar, com base na abordagem DQC (*Distance-Quantification-Classing*).

He *et al*. (2005) analisam a influência dos tipos de dados no processo de agrupamento e propõem uma abordagem diferente, efetuando a divisão do conjunto de

atributos em dois subconjuntos, um apenas com os atributos numéricos e outro apenas com os atributos categóricos. Em seguida, propõem o agrupamento de cada um dos subconjuntos isoladamente, utilizando algoritmos apropriados para cada um dos tipos. No final, os resultados de cada um dos agrupamentos são combinados em uma nova base de dados, que é submetida mais uma vez a um algoritmo de agrupamento para dados categóricos.

Luo *et al*. (2007) propõem um método alternativo ao particionamento de dados para geração de subconjuntos de treinamento de comitês, baseado na adição de ruído aos dados originais. O método propõe a utilização de ruídos artificiais para produzir variabilidade nos dados na execução dos algoritmos de agrupamento. Os dados artificiais gerados são uma aproximação dos dados reais, onde são computados a média e o desvio-padrão da amostra para gerar dados a partir da distribuição gaussiana encontrada.

Recentemente, têm surgido diversos trabalhos na área de comitês de agrupamento aplicados a bioinformática, particularmente na análise de expressão gênica. Silva (2006) investiga o uso de comitês de agrupamento na análise de expressão gênica. Os dados são analisados através de três diferentes algoritmos de agrupamento (*k*-médias, EM e hierárquico com ligação média) e os resultados são combinados através de diferentes técnicas, como votação, re-rotulagem e grafos. Os resultados mostram que essa abordagem obtém um resultado superior ao uso de técnicas individuais, particularmente quando são utilizados comitês compostos por diferentes algoritmos.

Faceli (2006) propõe uma arquitetura para análise exploratória de dados através de técnicas de agrupamento. A arquitetura é formada por um comitê de agrupamentos multi-objetivo, que executa vários algoritmos conceitualmente diferentes com várias configurações de parâmetros, combina as partições resultantes desses algoritmos e seleciona as partições com os melhores compromissos de diferentes medidas de validação. Entre as bases de dados utilizadas para validação da proposta, também estão incluídas algumas de expressão gênica.

4.3 Métodos de Particionamento de Dados

Existem duas situações distintas que demandam a necessidade de efetuar a análise de agrupamentos de forma distribuída. A primeira ocorre quando o volume de dados a analisar é relativamente grande, o que demanda um esforço computacional considerável, às vezes até inviável, para realizar essa tarefa. Por isso, a melhor alternativa é particionar os dados, agrupá-los de forma distribuída e unificar os resultados. A segunda ocorre quando os dados estão naturalmente distribuídos entre várias unidades separadas geograficamente e o custo associado à sua centralização é muito alto.

Determinadas aplicações atuais possuem bases de dados tão volumosas que não é possível mantê-las integralmente na memória principal, mesmo utilizando máquinas robustas. Kantardzic (2002) apresenta três abordagens para resolver esse problema:

i) Armazenam-se os dados em memória secundária e agrupam-se subconjuntos dos dados isoladamente. Os resultados parciais são guardados e, em uma etapa posterior, são reunidos para agrupar o conjunto inteiro;
ii) Utiliza-se um algoritmo de agrupamento incremental, onde cada elemento é trazido individualmente para a memória principal e associado a um dos grupos existentes ou alocado em um novo grupo. Os resultados são guardados e o elemento é descartado, para dar lugar ao próximo;
iii) Utiliza-se uma implementação paralela, onde vários algoritmos trabalham simultaneamente sobre os dados armazenados, aumentando a eficiência.

Nos casos em que o conjunto de dados se encontra unificado e necessita ser dividido em subconjuntos em função do seu tamanho, duas abordagens são normalmente utilizadas: particionamento horizontal e vertical (Figura 4.1).

A primeira abordagem é mais utilizada e consiste em dividir horizontalmente a base de dados, criando subconjuntos homogêneos dos dados, de forma que cada algoritmo opera sobre registros diferentes, porém considerando o mesmo conjunto de atributos. Outra abordagem é dividir verticalmente a base de dados, criando subconjuntos heterogêneos dos dados, nesse caso cada algoritmo opera sobre os mesmos registros, mas tratando de atributos diferentes.

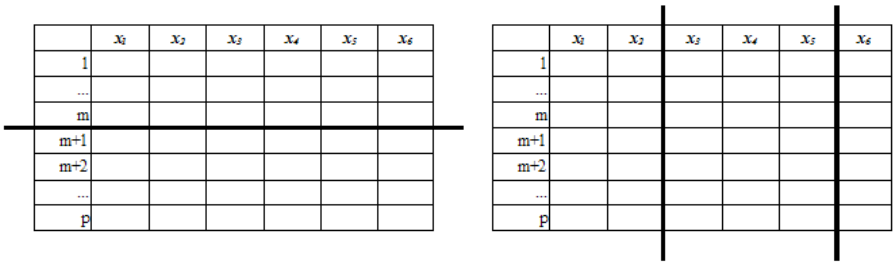


Figura 4.1: Ilustração das abordagens de particionamento horizontal e vertical.

Nos casos em que o conjunto de dados já se encontra particionado, como em aplicações que possuem bases de dados distribuídas, além das duas abordagens já citadas, é possível ainda encontrar situações em que os dados estejam dispersos simultaneamente de ambas as formas, denominado de particionamento arbitrário dos dados, que é uma generalização das abordagens anteriores [Jagannathan e Wright, 2005]

Tanto o particionamento horizontal quanto o particionamento vertical de bases de dados são comuns em diversas áreas de pesquisa, principalmente em ambientes com sistemas e/ou bancos de dados distribuídos, dos quais fazem parte as aplicações comerciais. A forma como os dados estão dispersos em um ambiente com bases de dados geograficamente distribuídas depende de uma série de fatores que nem sempre consideram a tarefa de análise de agrupamentos como uma prioridade dentro do processo.

Necessidades operacionais desses sistemas podem influenciar diretamente na forma de distribuição dos dados e os algoritmos de mineração de dados devem ser robustos o suficiente para lidar com essas limitações. Por exemplo, no projeto de um banco de dados distribuído, é importante gerar fragmentos que contenham atributos fortemente relacionados, de forma a garantir um bom desempenho nas operações de armazenamento e recuperação de informações [Son e Kin, 2004].

Estudos recentes sobre tecnologias de particionamento de dados buscam atender a essa demanda, particularmente em situações onde a falta de compatibilidade entre a distribuição de dados e as consultas efetuadas pode afetar o desempenho do sistema. Quando aplicado a bancos de dados distribuídos, o particionamento vertical oferece duas grandes vantagens que podem influenciar na performance do sistema. Primeiro, a frequência de consultas necessárias para acessar diferentes fragmentos de dados pode ser reduzida, pois é possível obter as informações necessárias com um número menor de consultas SQL. Segundo, a quantidade de informações desnecessárias recuperadas em uma consulta tradicional e transferidas para a memória também pode ser reduzida [Son e Kin, 2004].

Se por um lado os métodos de particionamento de dados mantêm o foco na performance das consultas, buscando o número de partições mais adequado para agilizar o processo de recuperação dos dados armazenados, a presença de variáveis redundantes ou fortemente correlacionadas em um processo de análise de agrupamentos com mapas auto-organizáveis não é aconselhável [Kohonen, 2001].

Portanto, para se obter um melhor resultado na análise dos dados, o mais aconselhado é distribuir geograficamente os dados de forma que variáveis correlacionadas ficassem em diferentes unidades. Entretanto, em situações onde as bases de dados já estejam geograficamente distribuídas, não sendo possível alterar a sua estrutura, e a existência de variáveis fortemente correlacionadas pode comprometer os resultados, pode-se utilizar técnicas estatísticas tais como Análise de Componentes Principais ou Análise Fatorial para selecionar um subconjunto mais adequado de variáveis e reduzir esses problemas.

A arquitetura apresentada neste trabalho foi desenvolvida com foco em bases de dados geograficamente distribuídas, independente dos critérios utilizados no particionamento. Por isso, buscou-se uma solução que fosse estável a qualquer forma de particionamento, independente de ser horizontal, vertical ou arbitrário, muito embora todas as técnicas propostas possam ser empregadas para melhorar o seu desempenho.

4.4 Segurança e Privacidade de Dados

A necessidade de garantir a confidencialidade das informações durante um processo de extração de conhecimento em bases de dados é uma área de pesquisa bastante atual dentro da comunidade científica [Kapoor *et al*., 2007]. Pesquisas envolvendo tecnologias relacionadas à segurança e preservação de privacidade em bases de dados

tiveram um crescimento inesperado nos últimos anos, impulsionadas pela crescente preocupação dos indivíduos em compartilhar suas informações pessoais via Internet e a preocupação das empresas em garantir a segurança destas informações [Verykios *et al*., 2004].

É sabido que a combinação de várias fontes de dados durante um processo de KDD melhora o processo de análise, embora ponha em risco a segurança e privacidade dos dados envolvidos no processo [Oliveira e Zaïane, 2007]. Por isso, algoritmos de mineração de dados que operam de forma distribuída precisam considerar não apenas a forma como os dados estão distribuídos entre as unidades, de forma a evitar transferências desnecessárias, mas devem garantir que os dados transferidos estejam protegidos contra eventuais tentativas de apropriação indevida.

Na medida em que repositórios digitais têm se tornado cada vez mais susceptíveis a ataques, empresas e organizações em todo o mundo têm sido frequentemente responsabilizadas por abusos, uma vez que os governos têm adotado legislações cada vez mais rigorosas com relação à privacidade dos dados coletados. Por isso, tais preocupações têm exigido novos avanços na área de mineração de dados distribuída, a fim de garantir a manutenibilidade das políticas de segurança e privacidade [Kapoor *et al*., 2006].

Um mercado potencialmente interessante para a mineração de dados distribuída são as organizações corporativas, compostas por um número significativo de empresas que atuam em torno de uma atividade principal, tais como os arranjos produtivos locais, os aglomerados de empresas, as redes corporativas, as cooperativas e as franquias. A aplicação simultânea de algoritmos de mineração de dados sobre as bases de dados das diversas empresas que atuam no mesmo setor permite obter informações mais completas e conhecimento mais preciso sobre este segmento, aumentando o conhecimento do grupo sobre a área de negócio [Thomazi, 2006].

No entanto, apesar das vantagens óbvias dessa abordagem, a maioria das empresas participantes de organizações corporativas opta por analisar apenas suas bases de dados individuais. Restrições legais de segurança impedem o compartilhamento de informações pessoais dos clientes entre empresas parceiras em vários países e criam uma série de problemas relativos à preservação de privacidade, inibindo as empresas de adotarem essa estratégia de compartilhamento de dados.

Análise de agrupamentos com preservação de privacidade surge como uma solução para este problema, permitindo que as partes cooperem entre si na extração do conhecimento sem que nenhuma delas tenha que revelar seus dados individuais para as outras. Essa abordagem concentra seus esforços em algoritmos que garantam privacidade e segurança aos dados envolvidos no processo, principalmente em aplicações onde a segurança é de fundamental importância, como em aplicações médicas e comerciais [Berkhin, 2006] [Silva, 2006].

Verykios *et al*. (2004) discutem o estado da arte em segurança e privacidade de dados, apresentando as três técnicas mais comuns: as baseadas em heurísticas, que

buscam alterar propositadamente alguns valores da base de dados, porém evitando perdas no processo; as baseadas em criptografia, que codificam os dados para evitar o acesso às informações pelas demais partes e as baseadas em reconstrução dos dados, que utilizam alguma técnica para introduzir uma perturbação nos dados, mantendo as relações existentes entre eles. Os autores apresentam ainda uma classificação dos principais algoritmos de mineração de dados de acordo com as técnicas apresentadas.

As primeiras referências aos problemas relacionados à segurança no processo de KDD surgiram ainda na década de 1990 [O'Leary, 1991] [Piatetsky-Shapiro, 1995] [Clifton e Marks, 1996]. No entanto, as primeiras pesquisas com resultados concretos na área de mineração de dados com preservação da privacidade foram publicadas por Agrawal e Srikant (2000) e Lindell e Pinkas (2000). Os primeiros, baseados em uma técnica de reconstrução dos dados conhecida como randomização, que introduz ruídos junto aos dados reais, evitando que os mesmos sejam reconstituídos, porém mantendo as relações existentes entre eles. Os últimos, utilizando uma técnica de criptografia denominada *Secure Multi-party Computation* (SMC), para classificação de dados sobre bases horizontalmente distribuídas. A técnica de SMC foi proposta por Goldreich *et al*. (1987), a partir da ideia original de Yao (1986).

Apesar de ambas as abordagens não considerarem a necessidade de redução de transferência de dados entre as unidades, diversos outros trabalhos que vieram em seguida são extensões diretas do uso dessas técnicas. Agrawal e Aggarwal (2001) deram sequência ao primeiro trabalho, adicionando mais privacidade aos dados e incluindo o uso do algoritmo EM na etapa de reconstrução dos dados. Em seguida, Evfimievski *et al*. (2002) adaptaram o algoritmo para extração de regras de associação sobre atributos categóricos, adicionando ruído aos dados e medindo a influência desses ruídos no resultado final.

A técnica utilizada no segundo trabalho, baseada em SMC, foi investigada em diversas outras pesquisas da área. Apesar de sua eficácia em garantir a segurança dos dados minerados, sua aplicação em tarefas de mineração de dados tem se mostrado ineficiente em virtude de sua complexidade [Clifton *et al*., 2002] [Du e Atallah, 2001]. Mais recentemente, algumas variações dessa técnica têm sido investigadas, no sentido de reduzir a complexidade.

Kantarcioglu e Vaidya (2002) criticam a segurança dos processos de randomização e a complexidade dos algoritmos de SMC, além da necessidade de todas as unidades estarem conectadas durante todo o processo. Como alternativa, apresentam uma arquitetura que contorna essas limitações na extração de regras de associação em bases de dados distribuídas com informações sobre clientes. No entanto, a arquitetura requer que a base de dados seja inteiramente transferida para a unidade central, o que a torna inviável em muitas aplicações de mineração de dados.

Vaidya e Clifton (2003) propõem uma implementação distribuída do algoritmo *k*-médias, baseada na técnica SMC, para análise de agrupamentos sobre bases de dados

verticalmente distribuídas. Lin *et al*. (2005) adaptam a mesma ideia para utilização com o algoritmo EM. Mais recentemente, Vaidya *et al*. (2006) sumarizam as técnicas mais utilizadas para mineração de dados com preservação de privacidade em tarefas de predição e descrição, tanto sobre bases de dados horizontalmente quanto verticalmente particionadas.

Técnicas estatísticas têm sido utilizadas para garantir a segurança e privacidade de dados em tarefas de agrupamento. Merugu e Ghosh (2003) apresentam uma arquitetura para agrupamento de dados distribuídos baseada uma técnica denominada *generative models*, que realiza uma perturbação dos dados com base em um modelo estatístico, para garantir a privacidade. Klusch *et al*. (2003) propõem um algoritmo para agrupamento distribuído baseado em estimação local de densidade. O algoritmo trabalha de forma distribuída, utilizando uma função objetivo para extrair a densidade local das partições e combina os agrupamentos parciais enviando informações sobre os núcleos dos agrupamentos para a unidade central. A privacidade e a segurança dos dados são mantidas, uma vez que apenas informações sobre os núcleos dos agrupamentos são compartilhadas.

Estivill-Castro (2004) propõe um método que combina um protocolo de comunicação entre duas ou mais partes, baseado no SMC, e o uso do algoritmo *k*-representantes (*k-medoids*), uma variante mais robusta do *k*-médias, para o agrupamento de dados verticalmente particionados. Outra abordagem baseada no algoritmo *k*-representantes propõem o uso de uma técnica de criptografia denominada cifragem homomórfica para permitir o compartilhamento dos dados entre as partes sem por em risco a segurança [Zhan, 2007]. Posteriormente, Zhan (2008) expandiu essa técnica para outras tarefas de mineração de dados. Jha *et al*. (2005) propõem a utilização do algoritmo *k*-médias através uso de dois protocolos de segurança, avaliação polinomial e cifragem homomórfica.

O problema que surge quando informações confidenciais podem ser deduzidas a partir dos dados disponibilizados a usuários não autorizados é conhecido como problema da inferência em bancos de dados [Verykios *et al*., 2004] [Farkas e Jajodia, 2002]. Oliveira e Zaïane (2003) introduzem um conjunto de métodos para perturbação de dados, baseados em transformações geométricas (translação, rotação, alteração na escala) no espaço *p*-dimensional. Inicialmente, são eliminados quaisquer atributos que possam ser utilizados para identificação individual dos objetos. Em seguida, o método efetua diversas transformações geométricas nos dados, mantendo as relações estatísticas entre eles, mas evitando que os mesmos sejam reconstruídos.

Posteriormente, Oliveira e Zaïane (2004) propõem um aprimoramento no método de transformações baseadas em rotação geométrica para proteger os valores dos atributos enquanto estes são compartilhados em um processo de agrupamento. A principal vantagem do método proposto é que ele é independente do algoritmo de agrupamento. Mais recentemente, os autores combinam os resultados de estudos anteriores em um novo

método para análise de agrupamentos com preservação de privacidade, denominado *Dimensionality Reduction-Based Transformation* (DRBT), com aplicações no setor comercial [Oliveira e Zaïane, 2007].

Jagannathan e Wright (2005) introduzem o conceito de particionamento arbitrário dos dados, que é uma generalização dos particionamentos horizontal e vertical e apresentam um método para tarefas de agrupamento de dados com o algoritmo *k*-médias sobre bases de dados arbitrariamente particionadas. O método utiliza um protocolo baseado em criptografia para garantir a privacidade dos dados. Jagannathan *et al*. (2006) sugerem uma variante mais segura do algoritmo *k*-médias anteriormente proposto, porém para agrupamento sobre bases de dados horizontalmente distribuídas.

İnan *et al*. (2006) e İnan *et al*. (2007) abordam a análise de agrupamentos com proteção da privacidade através de um algoritmo que permite construir uma matriz de dissimilaridade entre objetos sobre bases de dados horizontalmente distribuídas utilizando SMC para garantir a segurança. O algoritmo trabalha adequadamente com atributos numéricos e categóricos e a matriz de dissimilaridade construída pode ser aplicada a outras tarefas de mineração de dados.

Kapoor *et al*. (2007) apresentam um algoritmo denominado PRIPSEP (*PRIvacy Preserving SEquential Patterns*), baseado na técnica SMC, que permite minerar padrões sequenciais sobre bases de dados distribuídas, ao mesmo tempo em que mantém a preservação da privacidade dos indivíduos.

Em alguns trabalhos mais recentes, Vaidya (2008) apresenta e discute diversos métodos para mineração de dados que operam de forma distribuída sobre bases de dados verticalmente particionadas, enquanto que Kantarcioglu (2008) faz o mesmo para métodos que operam de forma distribuída sobre bases de dados horizontalmente particionadas.

Fung *et al*. (2008) propõem uma arquitetura para análise de agrupamento de dados que converte um processo de análise de agrupamento em uma atividade de classificação. O algoritmo proposto realiza o agrupamento dos dados e associa os dados a um conjunto de classes. Depois, codifica os dados reais através de rótulos e transmite os dados codificados e as respectivas classes para as outras unidades, dessa forma, preservando a privacidade dos dados envolvidos no processo.

4.5 Combinação de Resultados em Comitês

Um problema inerente à utilização de comitês de agrupamento é a combinação dos resultados parciais. Strehl (2002) descreve bem essa questão e apresenta as três abordagens mais comuns para solucionar esse problema sob diferentes pontos de vista. A primeira abordagem consiste em analisar a semelhança entre as diferentes partições produzidas através do uso de métricas de similaridade entre partições. A segunda utiliza hipergrafos para representar os relacionamentos entre os objetos e aplica algoritmos de particionamento de hipergrafos sobre eles para encontrar os agrupamentos. Na terceira

abordagem, os elementos do conjunto de entrada são rotulados e, em seguida, os rótulos são combinados para apresentar um resultado final, normalmente através de algum sistema de votação.

Strehl e Ghosh (2002) introduzem o problema de combinar múltiplas partições de um conjunto de objetos em uma única partição consolidada a partir dos rótulos parciais obtidos. Em suma, o objetivo dessa abordagem é obter um conjunto de rótulos que correspondem ao resultado de cada partição e, considerando apenas os resultados parciais, combiná-los para obter um resultado consenso, sem levar em consideração características anteriores sobre os objetos que determinaram as partições.

De fato, essa é a forma de fusão de resultados mais popular entre as três apresentadas por Strehl (2002) para tarefas de agrupamento de dados e as dificuldades associadas à sua utilização têm sido investigadas em diversos outros trabalhos que abordam o tema [Dimitriadou *et al*., 2001] [Frossyniotis *et al*., 2004] [Zhou e Tang, 2006] [Tumer e Agogino, 2008].

Alguns trabalhos baseados em comitês de SOM introduzem técnicas específicas de combinação de resultados através de técnicas de fusão dos mapas. Na abordagem proposta por Vrusias *et al*. (2007), um grande mapa auto-organizável é dividido em pequenos sub-mapas e enviado a diversas unidades de uma grade computacional, para ser treinado em paralelo. Cada unidade treina seu sub-mapa com um subconjunto diferente dos dados. Neste caso, a fusão dos resultados é feita a partir das médias dos mapas treinados individualmente. Os valores médios dos neurônios são calculados a partir da média aritmética, em cada dimensão, para cada nó do SOM no comitê. Esse cálculo é feito após um número pré-fixado de iterações na fase de treinamento.

Como cada rede neural é treinada a partir de seu respectivo subconjunto de dados, esse processo tende a perder em precisão em relação ao treinamento de uma única rede com todos os dados disponíveis, porém se obtém mais eficiência em relação ao tempo gasto no treinamento. Por outro lado, o comitê tem o potencial de gerar melhores resultados que uma única rede neural, uma vez que uma quantidade maior de treinamento pode ser realizada no mesmo intervalo de tempo.

Em outra proposta, Georgakis *et al*. (2005) sugerem um comitê de mapas auto-organizáveis treinados simultaneamente com subconjuntos de dados ligeiramente diferentes e utilizados para organizar e recuperar documentos. Nesse caso, a fusão dos resultados também é realizada através de uma média aritmética de seus pesos sinápticos, mas combinando-se os neurônios mais similares de cada mapa para compor um novo neurônio do mapa final. A dificuldade dessa proposta é manter a topologia dos mapas parciais no mapa final. A mesma estratégia é utilizada em um trabalho posterior para recuperação de imagens baseada em conteúdo [Georgakis e Li, 2006].

Hore *et al*. (2006) descrevem algumas formas de fusão de resultados baseado em combinação de rótulos e mostram que esses métodos não são adequados para aplicação em bases de dados muito volumosas. Por isso, apresentam uma proposta de comitês de

agrupamento que extrai um conjunto de centroides, rotula os centroides e combina esses resultados para formar os agrupamentos do conjunto original. Além disso, o trabalho inclui um processo de filtragem para agrupamentos mal-formados por falhas na inicialização, na distribuição dos dados ou pela existência de ruídos.

4.6 Resumo do Capítulo

Este capítulo discutiu o uso de comitês em tarefas de classificação e de agrupamento de dados. Foram analisadas questões relacionadas à existência de bases de dados geograficamente distribuídas e os mecanismos utilizados para o particionamento dos dados. Também foi apresentada uma ampla revisão sobre algoritmos e estratégias que abordam o uso de comitês, principalmente em tarefas de agrupamento.

Em seguida, foram abordadas questões relacionadas à segurança e privacidade em agrupamento de dados distribuídos. No final, foram apresentadas algumas técnicas de fusão de informações utilizadas para combinar resultados provenientes de múltiplas soluções de agrupamento.

No próximo capítulo será apresentada a arquitetura partSOM como uma proposta para realização de análise de agrupamento sobre bases de dados geograficamente distribuídos.

Capítulo 5
A Arquitetura partSOM

Computação bioinspirada é a área da Computação que busca inspiração na Natureza – particularmente na Biologia – para o desenvolvimento de novas técnicas de solução de problemas. Essa área desenvolve algoritmos e ferramentas baseados em processos naturais e inclui técnicas como redes neurais, computação evolutiva, algoritmos genéticos, programação genética e vida artificial. Resultados dessas pesquisas têm sido aplicados com sucesso em áreas diversas como reconhecimento de padrões, otimização, planejamento de cidades e roteamento de pacotes em redes de telecomunicações [Azevedo *et al*., 2000] [Johnson, 2003] [Whitby, 2004].

Redes neurais artificiais são uma abstração para o funcionamento do cérebro humano. São constituídas por várias unidades de processamento simples paralelamente distribuídas, denominadas neurônios, que têm a capacidade de armazenar conhecimento e torná-lo disponível para o uso [Haykin, 2001].

Muitos pesquisadores da área discordam quanto ao funcionamento de cérebros e computadores, defendendo que as redes neurais artificiais são muito diferentes das redes neurais biológicas no tocante ao processamento de informações [Teixeira, 1998] [Whitby, 2004] [Braga *et al*., 2007]. Entretanto, Haykin (2001) cita dois aspectos comuns aos dois sistemas: i) ambas as redes adquirem seu conhecimento interagindo com o ambiente, através de um processo de aprendizagem; ii) ambas armazenam o conhecimento adquirido através de um mecanismo denominado de pesos sinápticos, baseado na força de conexão entre os seus neurônios.

Este capítulo apresenta a arquitetura partSOM, baseada em redes neurais auto-organizáveis, cuja aplicação é direcionada à solução de problemas que envolvem a realização de análise de agrupamentos sobre bases de dados distribuídas. Inicialmente são discutidos alguns princípios básicos do funcionamento do cérebro humano que motivaram o desenvolvimento da arquitetura. A seguir, é apresentada a arquitetura partSOM e os seus respectivos algoritmos. Depois disso, é analisado o funcionamento da arquitetura através da simulação de um exemplo simples, utilizando uma base de dados de pequeno porte. Os resultados são comparados com aqueles obtidos pelos métodos tradicionais de análise de agrupamentos. Ao final, é apresentado um breve resumo do capítulo.

5.1 Motivação Biológica

O córtex cerebral humano é uma fina camada de células, com superfície total de aproximadamente 2.400 cm^2, espessura variando entre 1,5 e 4,5mm e massa aproximada de cerca de 1.300g [Kandel *et al.*, 1997] [Gazzaniga *et al.*, 2006]. O córtex, que em latim significa casca, corresponde à camada mais externa do cérebro e possui algo em torno de 10^9 neurônios, sua célula fundamental, sendo o local de processamento neuronal mais sofisticado [Haykin, 2001].

Toda a comunicação dentro do cérebro ocorre através de descargas eletroquímicas, denominadas sinapses, onde um neurônio devidamente estimulado por impulsos nervosos ou por outros neurônios é capaz de produzir descargas elétricas que podem estimular outros neurônios. Cada neurônio pode ter conexões com até 100 mil outros neurônios, formando uma imensa teia de conexões que se estendem por todo o córtex cerebral [Greenfield, 2000] [Rutkowski, 2005].

Apesar de bem mais lentos que os circuitos de silício, os neurônios conseguem compensar sua baixa velocidade de processamento em função da sua enorme quantidade, da intrínseca forma como estes estão interconectados e da capacidade de processamento em paralelo [Haykin, 2001]. Além disso, consomem menos energia e podem lidar com ruído [Pinker, 1998], o que os torna extremamente eficientes em determinadas tarefas que exigem processamento simultâneo de grandes volumes de informações.

A Figura 5.1 ilustra uma visão lateral do córtex cerebral humano [Gray, 1918]. As áreas coloridas correspondem a diferentes regiões do cérebro, cada uma delas associada a diversas funções. Existem áreas específicas do córtex cerebral, como o córtex visual, córtex auditivo e córtex somatossensorial que são responsáveis por realizar, receber e processar estímulos específicos. Outras regiões, como o córtex parietal ou córtex de associação, são responsáveis por estabelecer relações entre um sistema sensorial a outro.

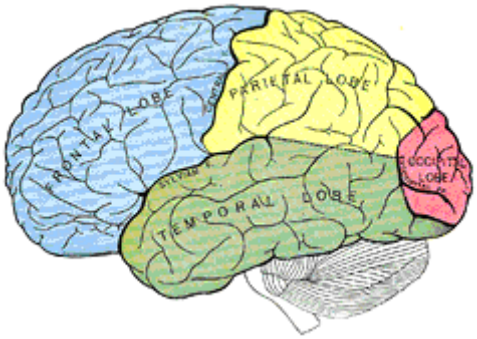


Figura 5.1: Ilustração do córtex cerebral humano.

Entretanto, não há uma área cerebral destinada a uma função específica ou que esteja diretamente relacionada a uma única atividade. Durante a execução de uma tarefa específica pelo cérebro humano, várias regiões cerebrais diferentes funcionam de modo simultâneo, contribuindo para execução dessa tarefa [Greenfield, 2000]. Apesar de

possuir áreas pré-determinadas a desempenhar diferentes funções, as informações acerca de um objeto que está sendo percebido dispersam-se por diversas partes do córtex cerebral [Pinker, 1998].

Kohonen (2001) relata que as áreas específicas do córtex cerebral (visual, auditivo, sensorial, etc.) ocupam menos de 10% do seu total e que situadas entre estas regiões existem as denominadas áreas associativas, nas quais os sinais de diferentes modalidades convergem. Portanto, para o processamento adequado das informações sobre um objeto em análise, é necessário ativar um mecanismo capaz de unir dados geograficamente separados dentro do cérebro.

Pinker (1998) afirma que o cérebro precisa ser constituído de partes especializadas para ser capaz de resolver problemas especializados, muito embora essas partes nem sempre sejam geograficamente bem delimitadas, podendo estar fragmentadas em regiões adjacentes, interligadas por meio de fibras que as fazem atuarem como uma unidade. Além disso, durante o processamento de informações, o cérebro humano pode fazer uso de mecanismos distintos. Ao assistir TV, por exemplo, é necessário ativar simultaneamente áreas responsáveis por atividades relacionadas à visão, audição, raciocínio lógico, emocional, etc.

A maioria das redes neurais baseia-se no modelo de McCulloch-Pitts, uma abstração extremamente simples de um neurônio biológico proposta por McCulloch e Pitts (1943). Um neurônio biológico possui um conjunto de entradas, denominadas dendritos, responsáveis por receber estímulos (na forma de impulsos) e enviá-los ao corpo celular, que realiza o processamento. Caso os estímulos recebidos atinjam um determinado limiar, um processo conhecido como sinapse produz um novo impulso, que é conduzido pelo axônio até os terminais sinápticos, para estimular outros neurônios [Ferneda, 2006].

O neurônio artificial simula esse processo através de um somador com entradas ponderadas (Σ) e de uma função de ativação (φ), que determina a propagação ou não da entrada recebida. A Figura 5.2 apresenta o mapeamento de um neurônio biológico em um modelo neurônio artificial.

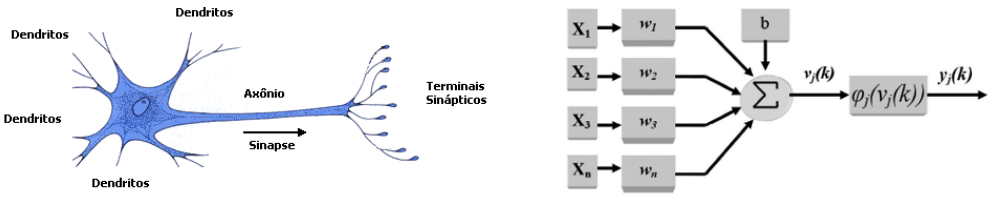


Figura 5.2: Mapeamento de um neurônio biológico em um neurônio artificial.

Embora as redes neurais sejam frequentemente criticadas em sua analogia às funções desempenhadas pelo cérebro humano, os avanços obtidos em áreas como reconhecimento de padrões têm justificado a sua utilização [Whitby, 2004]. Ademais,

recentes avanços obtidos nas áreas da neurociência têm influenciado os mecanismos de construção e treinamento de redes neurais, tornando essa área de pesquisa cada vez mais interdisciplinar [Sabbatini e Cardoso, 2002].

Os mapas auto-organizáveis, propostos por Kohonen, são constituídos por um conjunto de neurônios dispostos em uma grade e foram inspirados na estrutura do córtex visual [Kohonen, 1982a] [Kohonen, 1982b] [Fausett, 1994]. De acordo com Kohonen (2001), o algoritmo de treinamento dos mapas auto-organizáveis, baseado no princípio da aprendizagem competitiva (*winner-take-all*), é fortemente inspirado na forma de aprendizado do córtex visual e possui várias semelhanças com os sistemas emergentes e auto-organizáveis.

Em sua essência, o cérebro humano é um sistema formado por partes que agem colaborativamente para atingir seus objetivos. O córtex cerebral é dividido em regiões e, apesar de cada região responder a estímulos específicos e executar um conjunto de funções pré-determinadas, é a integração das partes que possibilita o seu desempenho de forma unificada. Utilizando essa abstração, a arquitetura partSOM proposta neste trabalho sugere a utilização de diversos mapas auto-organizáveis em paralelo, onde cada mapa atua com base em diferentes sub-conjuntos de dados, proporcionando diferentes visões das partes que compõem um mesmo conjunto de dados.

A arquitetura partSOM possui forte inspiração neurobiológica, uma vez que pressupõe a utilização simultânea de várias redes SOM que analisam diferentes subconjuntos de dados, fornecendo diferentes visões do conjunto de entrada. A ideia inserida na arquitetura proposta neste trabalho sugere que cada rede SOM possa analisar e capturar um contexto diferente, sendo responsável por tratar adequadamente um subconjunto dos dados, apresentando um funcionamento semelhante a cada uma das partes do córtex cerebral.

5.2 A Arquitetura partSOM

Algoritmos que realizam análise de agrupamentos de forma distribuída, normalmente a fazem em duas etapas. Em um primeiro momento, os dados são analisados localmente em cada uma das unidades que fazem parte da base de dados distribuída. Em uma segunda etapa, uma unidade central reúne os resultados parciais e os combina para obter um resultado global.

A arquitetura partSOM fornece uma estratégia para agrupar itens semelhantes em bases de dados horizontalmente, verticalmente ou arbitrariamente distribuídas, utilizando algoritmos tradicionais de análise de agrupamento, tais como mapas auto-organizáveis e *k*-médias. O processo também é realizado em duas etapas: na primeira etapa, um algoritmo de agrupamento é aplicado localmente em cada uma das bases distribuídas. Na segunda etapa, que distingue a arquitetura partSOM das demais estratégias, um algoritmo de agrupamento é novamente aplicado, desta vez sobre os representantes de cada uma das bases distribuídas, para criar um resultado definitivo.

Uma visão geral da arquitetura partSOM pode ser observada na Figura 5.3, onde são apresentadas todas as etapas do processo, que serão detalhadas a seguir.

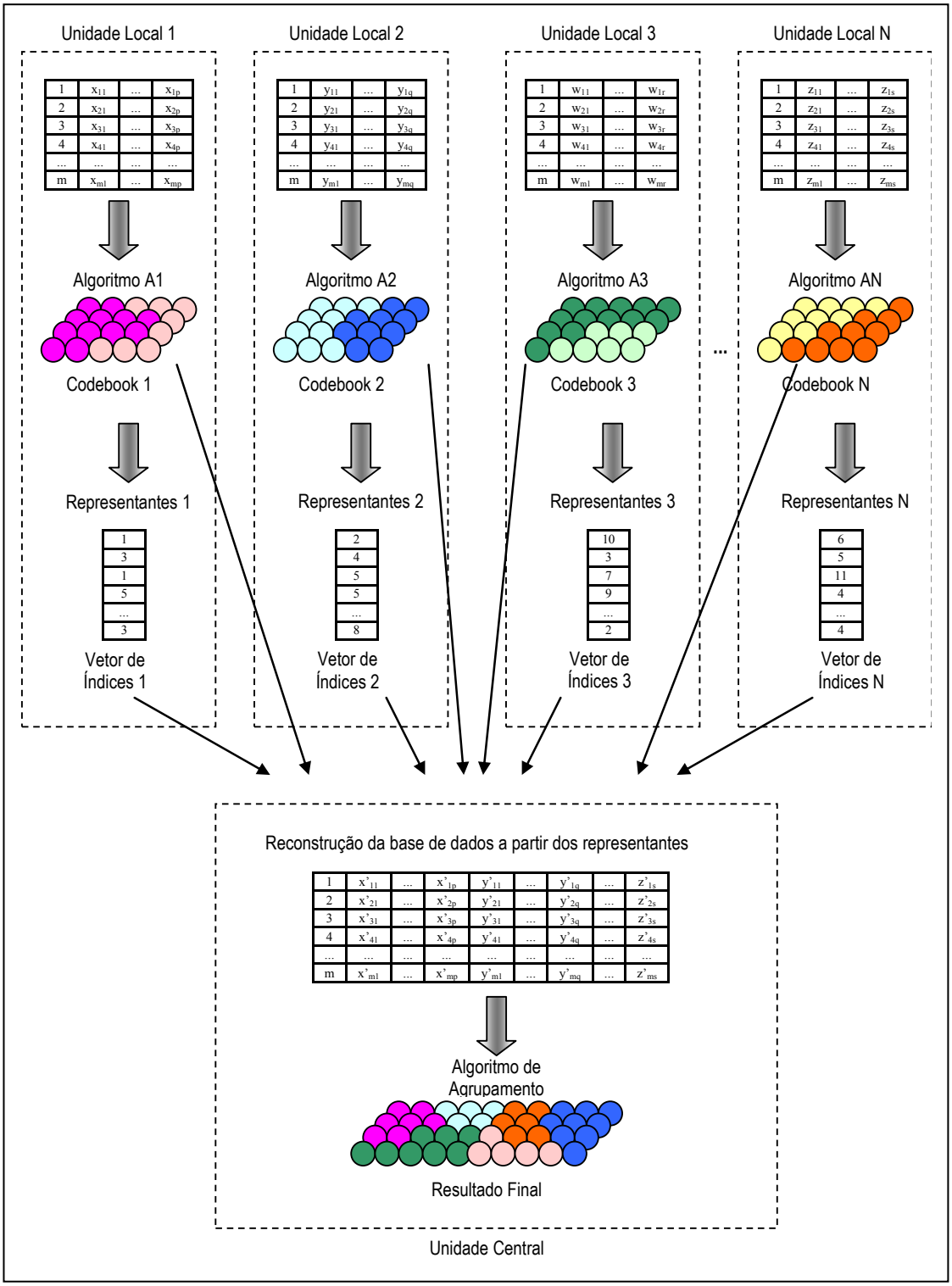


Figura 5.3: Visão geral da arquitetura partSOM.

A etapa local da estratégia proposta pode ser vista como um processo de quantização vetorial, onde o algoritmo de agrupamento é aplicado com o objetivo de

realizar uma compressão dos dados locais a serem enviados à unidade central. A segunda etapa do processo ocorre na unidade central, que recebe os dados processados em cada unidade remota e realiza, de fato, o processo de análise de agrupamentos, fornecendo o resultado final desejado.

Apesar da arquitetura proposta neste trabalho ser baseada na utilização de mapas auto-organizáveis e, por isso mesmo, ser denominada partSOM, a estratégia utilizada pela arquitetura possibilita a utilização de outros algoritmos de análise de agrupamentos, como o *k*-médias, por exemplo. A única restrição é que os algoritmos utilizados na etapa local do processo sejam do tipo baseado em protótipos (*prototype-based*), ou seja, possuam um *codebook* associado ao processo de agrupamento.

A seguir, são apresentados os algoritmos correspondentes às etapas local e central da arquitetura partSOM. Em seguida, são discutidos os passos que compõem cada uma das suas etapas:

Etapa Local

1. **A unidade local aplica um algoritmo de agrupamento sobre os dados locais e obtém um vetor referência de cada unidade**
2. **Os dados da unidade local são projetados sobre o seu vetor referência, criando-se um vetor de índices que representam cada um dos itens da unidade**
3. **O vetor de índices e o vetor referência de cada unidade local são enviados à unidade central**

No passo 1, cada unidade local aplica um algoritmo de agrupamento sobre seus dados e o resultado dessa fase é um conjunto de vetores referência (*codebook*) treinado para cada unidade. Como cada *codebook* é treinado individualmente, a partir de um subconjunto dos dados, cada um deles irá capturar a topologia do seu espaço de entrada, armazenando-a em seus pesos sinápticos. O *codebook*, apesar de possuir apenas um subconjunto dos dados originais, mantém as características existentes dos dados de entrada e constitui um conjunto de representantes dos dados locais.

No passo 2, cada vetor de dados de cada um dos subconjuntos de entrada é comparado ao seu respectivo *codebook*, com o objetivo de identificar qual dentre os elementos do *codebook* é o mais semelhante a ele. O índice do *codebook* mais semelhante (*codeword*) é armazenado em um vetor de índices, que possui o mesmo número de linhas do conjunto de entrada, mas apenas uma coluna. O procedimento de escolha dos representantes é efetuado pelo *algoritmo de escolha de representantes*, que será detalhado a seguir. Este algoritmo recebe um conjunto de n padrões de entrada $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$ e o *codebook* treinado $\mathbf{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k\}$ e retorna um vetor **R** que contém o índice do *bmu* de cada elemento de **X**.

```
              Algoritmo de Escolha dos Representantes

Entrada:    conjunto de entrada (X); codebook (W)
Saída:      conjunto de representantes (R)

algoritmo escolha_representantes(X,W)
para cada x_i ∈ X
      j = argmin d(x_i,w_j)
      R[i] = j
fim
retorne(R)
```

Dependendo do algoritmo de agrupamento utilizado no passo 1, o *codebook* poderá conter *codewords* com pouca ou nenhuma representatividade junto ao conjunto de entrada. No caso do algoritmo SOM, esses elementos são denominados neurônios inativos (*dead neurons*), sendo comumente abordados na literatura [Kamimura, 2003]. Embora os neurônios inativos ajudem a manter a topologia dos dados de entrada quando estes estão projetados no mapa, essas unidades podem ser descartadas sem prejuízo em um processo de quantização vetorial que utilize o SOM, pois tais elementos não serão referenciados na remontagem dos dados.

No caso do algoritmo *k*-médias, elementos do *codebook* com pouca representatividade podem corresponder a exceções ou ruídos existentes nos dados de entrada e, eventualmente, esses elementos podem ser descartados do conjunto de representantes sem grande prejuízo para a manutenção da distribuição estatística dos dados. Por isso, em ambos os casos é possível incluir uma etapa de poda antes da transferência dos dados para a unidade central, de forma a reduzir o tamanho do *codebook*, evitando-se transferir elementos que não serão utilizados (ou que não possuam representatividade) na reconstrução dos dados. O procedimento de redução do *codebook* é efetuado por um *algoritmo de poda*, que será detalhado a seguir.

O algoritmo de poda recebe o conjunto de entrada $\mathbf{X} = \{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_n\}$, o *codebook* treinado $\mathbf{W} = \{\boldsymbol{w}_1, \boldsymbol{w}_2, ..., \boldsymbol{w}_k\}$, o conjunto de representantes **R** e um valor inteiro θ que corresponde ao limiar (*threshold*) de representatividade exigido para cada elemento. Em seguida, o algoritmo busca por elementos cuja representatividade seja inferior ou igual ao limiar recebido e elimina-os do *codebook*. Ao final, o algoritmo de escolha de representantes é chamado mais uma vez para selecionar novamente os representantes de cada elemento do conjunto de entrada e atualizar os índices.

É importante ressaltar que o algoritmo de poda é uma etapa totalmente opcional, cujo objetivo é reduzir ainda mais a quantidade de dados transferidos entre as unidades

remotas e a unidade central. No caso particular, em que o valor de θ é igual a zero, apenas os neurônios inativos são eliminados, sem alteração nenhuma no resultado final.

Algoritmo de Poda

Entrada: conjunto de entrada (X); *codebook* original (W);
conjunto de representantes (R); valor do limiar (θ)
Saída: *codebook* modificado (W');
conjunto de representantes modificado (R')

```
algoritmo poda(X,W,R,θ)
para cada w_j ∈ W
      cont = 0
      para cada r_i ∈ R
            se (R[i] = j)
                  cont = cont + 1
            fim
      fim
      if (cont <= θ)
            W' = remova(W,j)
      fim
fim
R' = escolha_representantes(X,W')
retorne(W,R')
```

No passo 3, cada uma das unidades locais envia o seu vetor de índices e o seu respectivo *codebook* à unidade central. Esta é uma das principais vantagens da arquitetura partSOM em relação aos métodos tradicionais, ao invés de enviar todo o subconjunto de dados à unidade central, apenas os seus representantes são enviados, diminuindo-se sensivelmente o fluxo de dados entre as unidades remotas e a unidade central.

Etapa Central

4. **A unidade central remonta as bases remotas a partir dos vetores de índices e *codebooks* recebidos**
5. **A unidade central aplica um algoritmo de agrupamento sobre os dados reconstruídos, obtendo o resultado final do processo**

No passo 4 é realizado o procedimento inverso, com o objetivo de remontar a base de dados a partir dos valores recebidos. Cada elemento de cada um dos vetores de índice

é substituído pelo seu *codeword* equivalente, que se encontra presente no seu *codebook* correspondente, a fim de reconstruir uma tabela com valores similares àqueles existentes no subconjunto de entrada. Os dados são então justapostos em uma nova tabela, respeitando-se a mesma ordem em que estavam em suas bases de dados originais.

A expectativa é de que o resultado obtido nessa etapa possa ser generalizado como sendo equivalente ao conjunto de dados original. Os dados obtidos após a etapa 4 (e que servirão de entrada na etapa 5) correspondem a valores similares aos originais, uma vez que os elementos enviados no *codebook* são representantes próximos dos valores encontrados em cada um dos conjuntos de entrada.

Finalmente, no passo 5 é realizada uma nova segmentação, dessa vez sobre os dados reconstruídos a partir dos representantes recebidos. Um novo algoritmo de agrupamento é aplicado sobre os dados e o resultado obtido nessa segmentação é muito próximo ao que seria obtido com os dados originais.

5.3 Um Exemplo Prático

A seguir, é apresentado um exemplo prático de uso da arquitetura partSOM sobre uma base de dados bem simples, a fim de ilustrar o funcionamento de cada uma das partes que compõem o modelo proposto. A base de dados escolhida foi Animais [Kohonen, 2001], que contém 13 atributos sobre 16 espécies de animais e cujos dados são apresentados na Tabela 5.1. Os atributos que compõem a base de dados podem ser divididos em três grupos distintos: *Tamanho* – que corresponde ao porte do animal; *Características Físicas* – que descrevem particularidades da anatomia dos animais; e *Hábitos* – que identificam o comportamento do animal. Todos os atributos são binários e os pertencentes ao grupo *Tamanho* são mutuamente excludentes.

Tabela 5.1: Base de dados Animais

	Tamanho			*Características Físicas*						*Hábitos*			
Animal	peq	med	gde	2 patas	4 patas	pelos	casco	crina /juba	penas	caçar	correr	voar	nadar
Pombo	1	0	0	1	0	0	0	0	1	0	0	1	0
Galinha	1	0	0	1	0	0	0	0	1	0	0	0	0
Pato	1	0	0	1	0	0	0	0	1	0	0	0	1
Ganso	1	0	0	1	0	0	0	0	1	0	0	1	1
Coruja	1	0	0	1	0	0	0	0	1	1	0	1	0
Falcão	1	0	0	1	0	0	0	0	1	1	0	1	0
Águia	0	1	0	1	0	0	0	0	1	1	0	1	0
Raposa	0	1	0	0	1	1	0	0	0	1	0	0	0
Cão	0	1	0	0	1	1	0	0	0	0	1	0	0
Lobo	0	1	0	0	1	1	0	1	0	1	1	0	0
Gato	1	0	0	0	1	1	0	0	0	1	0	0	0
Tigre	0	0	1	0	1	1	0	0	0	1	1	0	0
Leão	0	0	1	0	1	1	0	1	0	1	1	0	0
Cavalo	0	0	1	0	1	1	1	1	0	0	1	0	0
Zebra	0	0	1	0	1	1	1	1	0	0	1	0	0
Vaca	0	0	1	0	1	1	1	0	0	0	0	0	0

Um processo de análise de agrupamentos tradicional que utilize o algoritmo *k*-médias, estipulando-se o valor de $k = 3$, obtém como resultado a segmentação da base em

três subconjuntos, conforme apresentado na Figura 5.4, onde se podem observar claramente a distinção entre 'aves', 'carnívoros' e 'herbívoros'.

Animal	*Agrupamento*
Pombo	1
Galinha	1
Pato	1
Ganso	1
Coruja	1
Falcão	1
Águia	1
Raposa	2
Cão	2
Lobo	2
Gato	2
Tigre	2
Leão	2
Cavalo	3
Zebra	3
Vaca	3

Figura 5.4: Segmentação da base de dados Animais obtida com o algoritmo *k*-médias.

Se, ao invés do *k*-médias, for utilizado o algoritmo SOM, algumas informações sobre a topologia existente no conjunto de entrada são adicionadas ao resultado através da forma como os animais estão dispostos no mapa, conforme apresentado na Figura 5.5, que inclui a matriz-U e os indivíduos associados aos seus respectivos *bmu*. Com o uso do SOM, é possível distinguir suaves diferenças entre algumas espécies que pertencem ao mesmo agrupamento na segmentação obtida com o *k*-médias e que são posicionadas mais próximas ou menos próximas entre si no mapa 2D em função das suas similaridades. No entanto, nenhuma informação sobre quais atributos influenciaram na distribuição obtida pode ser recuperada do resultado final.

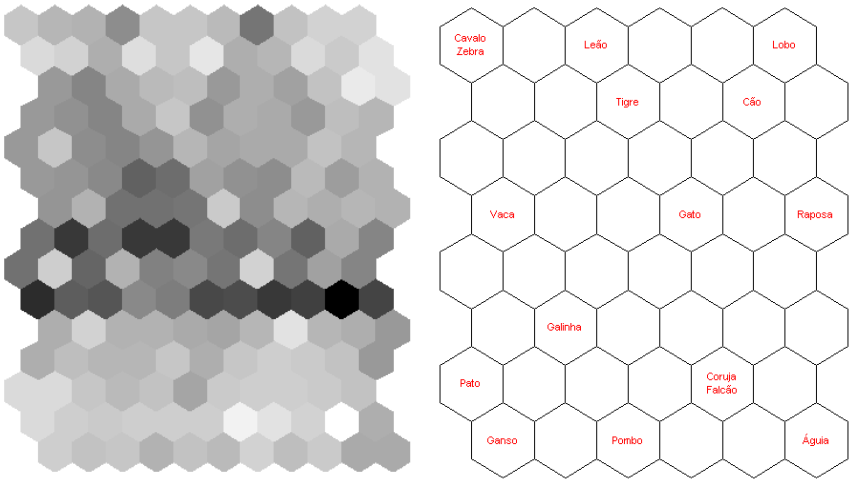


Figura 5.5: Segmentação da base de dados Animais obtida com o algoritmo SOM.

Para o processo de análise dos dados com a arquitetura partSOM, a base de dados foi dividida verticalmente, considerando-se os três aspectos existentes no conjunto de

atributos: tamanho, características físicas e hábitos. Cada um desses aspectos foi analisado individualmente, de forma a capturar a distribuição dos dados sob os diferentes aspectos e verificar como estes podem influenciar no resultado final. O resultado da segmentação dos mapas parciais é apresentado na Figura 5.6, assim como o resultado da segmentação final sobre os dados remontados.

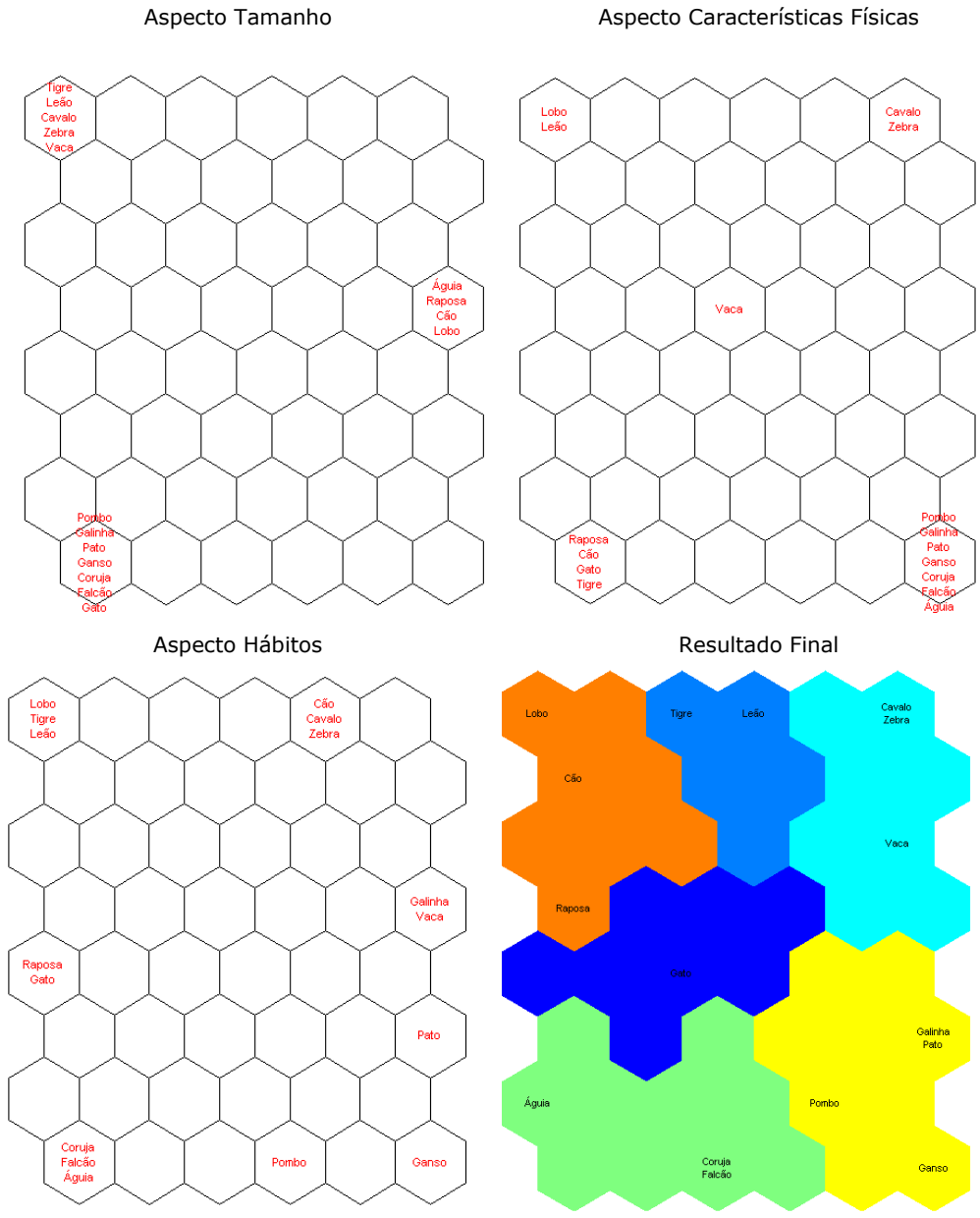


Figura 5.6: Segmentações parciais e final obtidas com a arquitetura partSOM.

A utilização da arquitetura partSOM fornece algumas informações adicionais sobre a influência que cada conjunto de atributos exerce sobre processo de segmentação da base, que não podem ser obtidas em um processo de segmentação tradicional, conforme pode ser analisado nas visões parciais. Por exemplo, se analisarmos a

segmentação do conjunto de dados sob o aspecto ‘tamanho’, é possível verificar que existem 3 grupos distintos de animais. No entanto, sob o aspecto ‘características físicas’, existem 5 grupos bem definidos, sendo possível verificar que, observando desse ponto de vista, os indivíduos ‘leão’ e ‘lobo’ se destacando dos demais carnívoros.

Entretanto, a segmentação obtida com os atributos correspondentes ao aspecto ‘hábitos’ apresenta uma riqueza maior de detalhes que não é possível identificar nem nos demais mapas, nem tampouco na segmentação total, onde todos os atributos são considerados simultaneamente. Nesse caso, tais informações são extremamente valiosas para se entender melhor o comportamento dos indivíduos da base de dados, separando aves predadoras, como ‘coruja’, ‘águia’ e ‘falcão’, de outras espécies não-predadoras, como ‘pombo’, ‘ganso’, ‘pato’ e ‘galinha’. Além disso, os ‘carnívoros’ que possuem hábitos de caça foram separados do ‘cão’, que tem comportamento mais doméstico.

Supondo um ambiente em que os dados estivessem verticalmente distribuídos dessa maneira. Uma rápida avaliação dos mapas parciais obtidos em cada segmentação mostra que seria possível reduzir sensivelmente a quantidade de dados transmitidos entre as unidades remotas e a unidade central. Na unidade 1 seria necessário um *codebook* com apenas 3 centróides para representar todos os 16 indivíduos. Na unidade 2 seria necessário um *codebook* com 5 centróides e na unidade 3 seriam necessários 8 centróides. Assim, apenas com os três *codebooks* e com um vetor de índices para cada conjunto, seria possível remontar o conjunto original de dados.

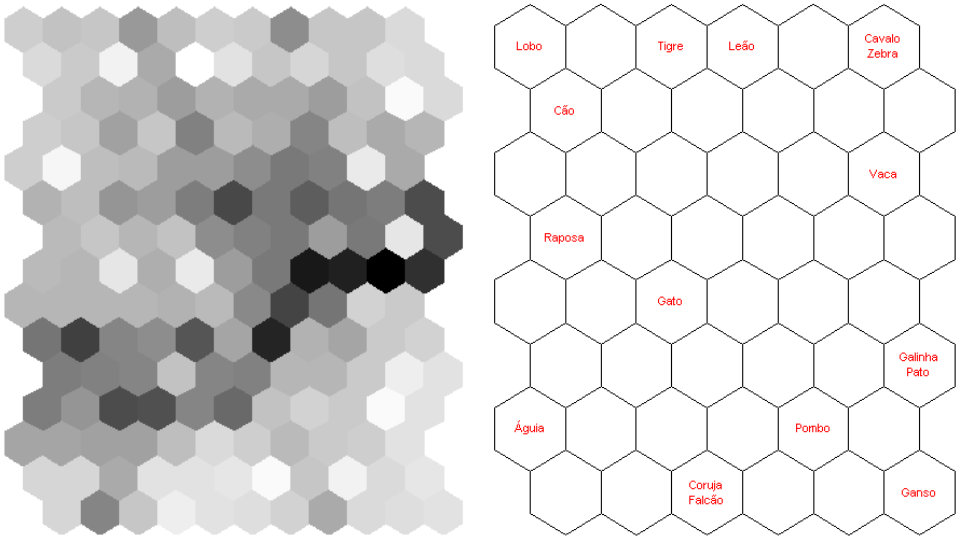


Figura 5.7: Segmentação da base de dados Animais obtida com a arquitetura partSOM.

A Figura 5.7 mostra a matriz-U e os indivíduos rotulados sobre as unidades do mapa, obtidos após a etapa central da arquitetura partSOM. Por se tratar de uma base de dados extremamente pequena, utilizada apenas com a finalidade de demonstrar o

funcionamento da arquitetura e com indivíduos bem semelhantes entre si, o conjunto de dados remontados é praticamente igual ao original e os resultados finais obtidos com a abordagem tradicional e com a abordagem proposta são bastante próximos.

5.4 Avaliação da Arquitetura partSOM

De acordo com Kargupta *et al*. (2000), um algoritmo típico de mineração de dados distribuídos deve possuir as duas seguintes etapas:

i) Realizar a análise dos dados locais para a geração de modelos parciais;
ii) Combinar os modelos de dados parciais de diferentes unidades a fim de construir um modelo global.

Abordagens convencionais podem ser ambíguas e susceptíveis a gerar erros na criação dos modelos parciais dos dados [Kargupta *et al*., 2000]. Por isso, é necessário tomar certas precauções na escolha da abordagem utilizada, a fim de garantir que os modelos gerados nesta etapa de análise realmente reflitam as características existentes nos dados.

A primeira etapa da arquitetura partSOM utiliza um processo de quantização vetorial para efetuar uma compressão nos dados de entrada e assim, reduzir a quantidade de dados transferidos à unidade central. Como em qualquer processo de compressão de dados, existem perdas associadas à quantização vetorial e, eventualmente, uma parte das informações existentes nos dados de entrada será descartada durante a primeira etapa do algoritmo.

Entretanto, conforme descrito em Costa (1999), um processo de quantização vetorial aproxima uma função densidade de probabilidade do conjunto de entrada através de um conjunto finito de vetores de referência. Assim, se o conjunto de vetores referência escolhido para representar os dados de entrada for suficientemente representativo para capturar a distribuição estatística dos dados no espaço de entrada, as relações de proximidade entre os elementos de entrada serão mantidas. Dessa forma, mesmo que o processo de quantização vetorial apresente perdas, essas perdas tendem a ser minimizadas com a escolha adequada de um bom conjunto de representantes.

Uma alternativa para minimizar as perdas ocorridas com o processo de quantização vetorial é a utilização de informações estatísticas adicionais contidas na amostra original, de forma que os dados reconstruídos sejam o mais próximo possível dos dados de entrada. A matriz de covariância (ou matriz de dispersão) de um conjunto de dados permite extrair a variância e a correlação entre as amostras, sendo uma solução eficiente para criar amostras aleatórias que contenham as mesmas características estatísticas da amostra original.

Assim, se as matrizes de covariância de cada agrupamento forem extraídas nas unidades remotas e enviadas junto com os *codebooks*, de forma que cada centróide carregue consigo informações adicionais sobre a covariância dos dados que representa, essas informações podem ser utilizadas para gerar amostras com distribuição estatística

ainda mais próxima do conjunto de dados original, contribuindo para redução das perdas associadas ao processo de quantização vetorial.

A Figura 5.8 apresenta esses conceitos graficamente. Os pontos pretos (lado esquerdo da figura) representam os dados originais do conjunto de entrada. Os pontos vermelhos representam os vetores referência, que constituem os representantes do conjunto original. Os pontos azuis (lado direito) representam os valores recriados a partir da descrição estatística contida nas matrizes de covariância. Mesmo que os dados originais sejam removidos, os representantes destes ainda mantêm a mesma distribuição estatística no espaço, sendo possível recriar um conjunto de dados com distribuição estatística semelhante.

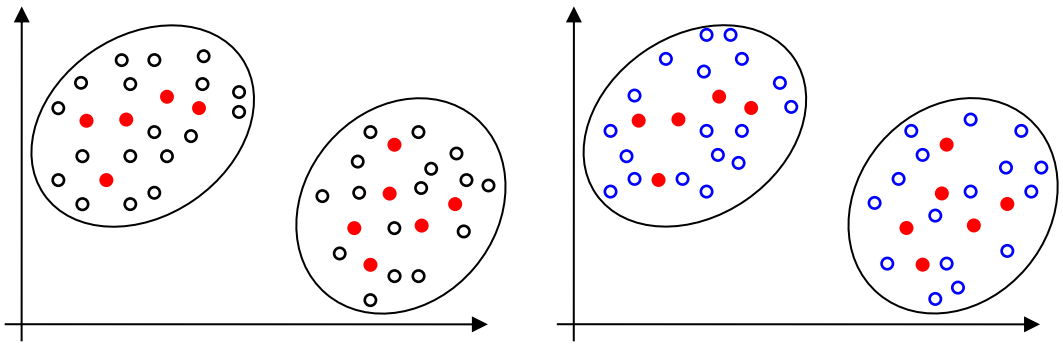


Figura 5.8: Distribuição estatística do conjunto de representantes no espaço 2D.

A taxa de compressão do algoritmo pode ser controlada a partir do tamanho do *codebook*. Se o tamanho do *codebook* for muito pequeno, têm-se uma compressão maior, ou seja, menos dados serão transferidos entre as unidades remotas e a unidade central e a perda de informação é maior. Por outro lado, se o tamanho do *codebook* for grande, uma maior quantidade de dados será transferida entre as unidades remotas e a unidade central, no entanto a taxa de compressão será menor e menos informação será descartada.

A ideia central do processo de quantização vetorial é o uso de um ou mais *codebooks* como representantes de um agrupamento. Essa técnica é a base do algoritmo CURE [Guha *et al*., 1998], além de ter sido explorada em outros algoritmos de agrupamento de dados. O CURE (*Clustering Using REpresentatives*) é um algoritmo hierárquico que utiliza um número pré-definido de centróides para representar cada agrupamento. Em cada etapa do processo, um conjunto de representantes é escolhido aleatoriamente ao longo do agrupamento e, em seguida, esses representantes são movidos em direção ao centro do agrupamento, tornando-o menos sensível a ruídos. Na arquitetura partSOM, o uso de mapas auto-organizáveis na primeira etapa favorece a escolha dos representantes, uma vez que o SOM captura adequadamente a topologia dos dados de entrada.

Hore *et al*. (2006) também apresentam uma proposta de utilização de agrupamento de centróides como uma generalização do agrupamento do conjunto total. No entanto, na abordagem proposta por esses autores, é feita a rotulação dos centróides e, em seguida, os rótulos são agrupados, não sendo possível a análise estatística dos dados no resultado final. Na proposta da arquitetura partSOM, os centróides é que são novamente agrupados para formar o resultado final.

Um modelo de análise de agrupamentos distribuída aplicada a segmentação de mercado deve considerar questões relacionadas à segurança e privacidade dos dados analisados. Os modelos parciais obtidos na etapa local devem garantir que nenhuma informação submetida à unidade central possa ser utilizada para obtenção dos dados originais. No caso da arquitetura partSOM, a exemplo do que é proposto em Klusch *et al*. (2003), somente os representantes de cada partição são enviados à unidade central, nenhuma informação real é transmitida.

É importante destacar ainda que nenhum dos modelos de comitês de agrupamento analisados na literatura considera a reaplicação de um algoritmo de análise de agrupamento sobre o conjunto de dados obtido a partir da primeira etapa. Essa reavaliação dos dados possui duas vantagens: i) fornece uma visão global dos dados, que normalmente não se obtém com os comitês tradicionais e ii) permite capturar detalhes parciais que poderiam estar escondidos ou passar despercebidos numa análise global, considerando todos os atributos.

A arquitetura partSOM pode utilizar qualquer algoritmo de análise de agrupamentos que seja baseado em protótipos (*prototype-based*). Nesse trabalho foram utilizados os algoritmos *k*-médias e SOM, mas outros algoritmos podem ser testados sem perda de generalidade da proposta.

5.4.1 Principais Vantagens da Arquitetura partSOM

Um dos pontos fortes da arquitetura partSOM é a possibilidade de se obter avaliações parciais durante a etapa local de agrupamento. Considerando-se, por exemplo, uma base de dados de clientes de uma empresa particionada horizontalmente, de maneira que cada unidade local possua os mesmos atributos, mas relacionados a clientes de regiões diferentes.

Em um processo de análise de agrupamentos tradicional, as bases de dados são reunidas em uma matriz de dados única na unidade central e, em seguida, segmentadas, para se obter o resultado desejado. A estratégia proposta pela arquitetura partSOM permite que se obtenha avaliações parciais de cada uma das bases isoladamente, antes de se obter o resultado global. Dessa forma, é possível identificar determinadas estruturas em uma base que não estão presentes na outra. Tais estruturas podem significar, por exemplo, um determinado perfil de clientes que só é encontrado em uma das regiões.

Por outro lado, a estratégia proposta pela arquitetura partSOM também oferece vantagens na avaliação parcial de bases de dados com particionamento vertical.

Considerando-se, agora, uma base de dados de clientes de uma empresa particionada verticalmente, de maneira que cada unidade local possua diferentes tipos de informações relacionados ao mesmo grupo de clientes. A segmentação dos clientes baseada em um conjunto específico de atributos e considerando diferentes pontos de vista (por exemplo: demográfico, geográfico, psicográfico, comportamental) pode contribuir para entender melhor o mercado.

Por exemplo, é possível concluir que do ponto de vista geográfico a empresa possui 3 perfis de clientes (possivelmente relacionados ao bairro ou região onde residem). No entanto, do ponto de vista comportamental, existem 5 perfis distintos. Uma análise mais detalhada pode revelar que perfil de cliente está associado a qual região e facilitar a criação de uma estratégica de marketing direcionada a um determinado segmento.

Além disso, uma avaliação parcial dos resultados pode revelar detalhes que passariam despercebidos em um processo de análise global, com os dados reunidos a outros de maior volume. Um exemplo disso é o caso das duas clientes com perfis aparentemente similares, citado na seção 3.1 desse trabalho. Uma análise parcial dos atributos psicográficos referentes às duas clientes certamente apresentaria as diferenças existentes entre os perfis de ambas, o que poderia não ficar tão evidente em uma análise que considerasse todos os atributos simultaneamente.

Outra vantagem das avaliações parciais é a minimização das perdas existentes em qualquer procedimento de análise de agrupamentos e que estão associadas ao processo de redução da dimensionalidade. Uma vez que cada unidade local é treinada com um subconjunto dos dados e que a redução na dimensionalidade do conjunto de entrada é menos drástica, as perdas relacionadas ao processo serão menores, pelo menos na etapa local de cada unidade.

A privacidade e a segurança dos dados e informações analisados durante o processo de segmentação das bases de dados são garantidas através de mecanismo baseado no que foi descrito por Hore *et al*. (2006), que substitui de dados reais por um conjunto de centróides com a mesma distribuição estatística dos dados. Assim, uma vez que apenas os vetores referência e um vetor de índices são enviados à unidade central e que nenhuma informação real do conjunto de entrada é transferida, a segurança é mantida.

A redução do volume de dados transferidos entre as unidades remotas e a unidade central é outro ponto forte da arquitetura partSOM. Supondo uma matriz de dados de tamanho $n \times p$, onde n representa a quantidade de instâncias e p representa a cardinalidade do conjunto. O *codebook* relacionado a esse conjunto de dados pode ser representado por uma matriz de dimensão $k \times p$, onde k representa o número de *codewords* do *codebook*. O vetor de índices pode ser representado por um vetor de dimensão $n \times 1$. Uma vez que apenas o *codebook* e o vetor de índices são enviados à unidade central e que, normalmente, $k \ll n$, a estratégia proposta pela arquitetura partSOM consegue reduzir sensivelmente o tráfego de dados entre as unidades remotas e a unidade central.

Uma última vantagem dessa abordagem é a facilidade de implementação, uma vez que a arquitetura partSOM pressupõe a utilização de algoritmos simples e tradicionais de análise de agrupamentos, disponíveis nas principais ferramentas de mineração de dados existentes no mercado. Todos os experimentos desenvolvidos nesse trabalho utilizaram apenas os algoritmos SOM e *k*-médias, além dos algoritmos desenvolvidos para a arquitetura que foram descritos neste capítulo.

5.4.2 Algumas Limitações da Arquitetura partSOM

Uma desvantagem da arquitetura partSOM é a necessidade de mais de uma etapa de agrupamento de dados, o que aumenta a complexidade do processo. Em um procedimento tradicional, os dados são transferidos à unidade central e segmentados uma única vez. Em um procedimento que utilize a abordagem proposta pela arquitetura partSOM, cada subconjunto dos dados é analisado uma vez e um conjunto inteiro dos dados é analisado novamente. Além disso, o processo de análise final somente poderá ser realizado após a conclusão das análises individuais. Como uma forma de agilizar o processo, o treinamento das unidades locais pode ser realizado em paralelo.

5.4.3 Trabalhos Relacionados à Arquitetura partSOM

A arquitetura partSOM é resultado de diversas pesquisas envolvendo mapas auto-organizáveis, comitês de agrupamento e segmentação de mercado, desenvolvidas junto ao Laboratório de Sistemas Adaptativos (LSA) da Universidade Federal do Rio Grande do Norte. São apresentados a seguir os resultados de alguns trabalhos parciais publicados em conjunto com outros membros da equipe e que foram desenvolvidos durante o período.

Gorgônio e Costa (2007) introduzem a proposta inicial da arquitetura partSOM, mostrando a possibilidades de analisar agrupamentos de forma distribuída através de múltiplos mapas auto-organizáveis. Nesse trabalho foram apresentados alguns resultados iniciais obtidos com as bases de dados Iris, Wine e Breast Cancer Wisconsin.

Gorgônio e Costa (2008a) aprofundam a proposta de mineração de dados distribuída, discutindo alguns trabalhos relacionados ao tema, mais particularmente, à análise de agrupamentos distribuída e demonstram a importância de se considerar outros fatores tais como segurança e privacidade dos dados analisados.

Gorgônio e Costa (2008b) estendem a arquitetura partSOM incluindo uma etapa de pós-processamento, realizada através da segmentação dos mapas auto-organizáveis com o algoritmo *k*-médias. A abordagem foi utilizada para facilitar a análise dos resultados obtidos nos trabalhos anteriores.

Gorgônio e Costa (2008c) apresentam uma revisão bibliográfica mais profunda da área, incluindo os resultados de novos testes com a arquitetura partSOM aplicada a outras bases de dados, como a Wages e Mushroom. Também é apresentada uma avaliação

preliminar sobre a redução de dados transferidos entre as unidades remotas e a unidade central, mostrando a eficiência da estratégia nesse sentido.

Gorgônio *et al*. (2008) demonstram a validade da estratégia de análise de agrupamentos embutida na arquitetura partSOM, mostrando que é possível utilizar outros algoritmos de análise de agrupamento. São apresentados resultados de experimentos realizados com os algoritmos SOM e *k*-médias, utilizando as bases de dados Iris, Wine e Mushroom.

Gorgônio (2008) apresenta uma revisão sobre alguns trabalhos relacionados à utilização de mapas auto-organizáveis em tarefas de segmentação de mercado. Por fim, Gorgônio e Costa (2010) incluem modificações no algoritmo original da arquitetura partSOM com a introdução da matriz de covariância como parte dos descritores dos sub-conjuntos de dados locais, de forma a assegurar que os dados remontados sejam ainda mais próximos do conjunto original.

5.5 Considerações sobre o Processo de Seleção de Atributos

Um dos fatores de maior influência na qualidade do processo de análise de agrupamentos é a correta seleção dos atributos a serem considerados. Na verdade, a escolha dos atributos é uma decisão extremamente importante em qualquer processo de descoberta de conhecimento, uma vez que os passos seguintes do processo, assim como os resultados obtidos, passam a depender dessa decisão.

A correta seleção dos atributos que irão compor um processo de análise de agrupamento não depende exclusivamente do problema e dos resultados desejados. Idealmente, devem-se escolher atributos que representem bem as características dos agrupamentos existentes e que apresentem pouca variância entre objetos de um mesmo agrupamento e grandes diferenças entre agrupamentos vizinhos [Costa, 1999].

O problema torna-se ainda maior quando se considera o uso de comitês de agrupamento que operam sobre diferentes conjuntos de atributos, uma vez que é necessário selecionar vários subconjuntos de atributos, um para cada membro do comitê. De fato, existem várias pesquisas associadas à seleção de atributos com aplicações em comitês de agrupamento [Opitz, 1999] [Klepaczko e Materka, 2004] [Santana *et al*., 2007] [Jian e Bangzhu, 2007] [Vale *et al.*, 2008].

Uma abordagem bastante utilizada em bases de dados de alta dimensionalidade é realizar a análise a partir de um subconjunto dos atributos, ao invés de considerar todas as variáveis simultaneamente. Essa abordagem é conhecida como seleção de atributos ou seleção de variáveis (*feature selection*). Apesar de eficiente, uma dificuldade óbvia dessa abordagem é identificar quais variáveis são mais importantes no processo de identificação dos grupos. A utilização de métodos estatísticos, tais como Análise de Componentes Principais (PCA) e Análise Fatorial, é empregada com frequência na realização nessa tarefa [Hair Jr. *et al.*, 2005].

Dentro da proposta de redução do conjunto de variáveis, Friedman e Meulman (2004) apresentam um algoritmo denominado COSA (*Clustering Objects on Subsets of Attributes*) que não apenas realiza análise de agrupamentos a partir de um subconjunto dos atributos, mas identifica que variáveis são mais importantes em cada grupo detectado. Em um trabalho mais recente, Damian *et al*. (2007) demonstram as vantagens da aplicabilidade do algoritmo COSA na análise de sistemas biológicos, que normalmente possuem alta dimensionalidade.

Laine (2002) demonstra como o usuário pode identificar as variáveis mais importantes em um processo de mineração de dados através da utilização de mapas auto-organizáveis para a tarefa de análise de agrupamentos. Para isso, o autor descreve um método que analisa o mapa SOM treinado e identifica o conjunto de variáveis que melhor separa os grupos.

Considerando-se que quanto maior o volume de dados, maiores as dificuldades de atender a essas restrições, em bases de dados que possuam grande número de registros e que cada registro possua uma grande quantidade de atributos, aumentam as chances de que o processo de identificação de agrupamentos não seja tão bem-sucedido.

Uma alternativa frequentemente argumentada pelos defensores do uso de comitês é a utilização de diferentes algoritmos sobre diferente subconjunto de dados, correspondendo a partições horizontais ou verticais. Porém, como a combinação de possibilidades é exponencial, não há como garantir que a combinação de conjuntos de atributos e algoritmos escolhidos será a mais apropriada.

Por isso, a arquitetura partSOM realiza o processo de agrupamento uma única vez, durante a etapa final. As etapas locais servem apenas para escolher um subconjunto de representantes dos dados que represente bem aquela partição. Além disso, para tentar minimizar essas dificuldades, propõe-se o uso da arquitetura partSOM com múltiplas instâncias do mesmo algoritmo e com o particionamento do conjunto de entrada que permita uma certa sobreposição entre os conjuntos.

5.6 Resumo do Capítulo

Este capítulo apresentou a arquitetura partSOM, foco central do trabalho proposto, que permite analisar agrupamentos sobre bases de dados distribuídas, sem necessidade de integração dos dados. A arquitetura é uma alternativa viável à abordagem tradicional, onde os dados são agrupados em um único local antes da aplicação dos algoritmos de mineração de dados. Além disso, foram discutidas a motivação biológica para a arquitetura, seu princípio de funcionamento, suas vantagens e algumas limitações.

Em seguida, foram apresentados os algoritmos desenvolvidos para criação dos modelos parciais e codificação do vetor de índices (etapa local), reagrupamento dos dados e remontagem da base (etapa central) e o algoritmo de poda. Depois, foi apresentado um exemplo prático de uso da arquitetura partSOM com uma base de dados ilustrativa.

No final, foram discutidas algumas questões relacionadas ao particionamento dos dados e à distribuição dos atributos em cada partição. No próximo capítulo serão apresentados os experimentos realizados e os resultados obtidos, comprovando a eficácia do modelo proposto.

Capítulo 6
Métodos e Experimentos

Conforme apresentado no capítulo anterior, a arquitetura partSOM é bastante flexível, tanto em função das inúmeras formas de particionamento dos dados, como na possibilidade de utilização de vários algoritmos de agrupamento.

Assim, com o intuito de comprovar a validade do modelo proposto, foi realizada uma série de experimentos com diferentes bases de dados. Os experimentos foram conduzidos utilizando diferentes formas de particionamento das bases de dados e diversas combinações de possibilidades de uso dos algoritmos SOM e *k*-médias.

Este capítulo inicia com uma descrição das bases de dados utilizadas nos experimentos. Em seguida, são descritas a metodologia utilizada, os critérios de avaliação que foram considerados e os detalhes sobre cada um dos experimentos realizados.

6.1 Descrição das Bases de Dados

Para a avaliação da arquitetura partSOM e dos algoritmos propostos, foram selecionadas diversas bases de dados, tanto artificiais quanto reais, detalhadas a seguir.

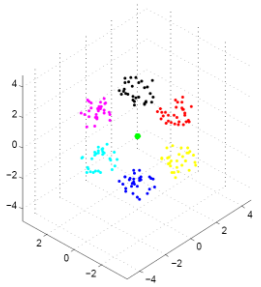
6.1.1 Bases de Dados da FCPS

O conjunto de dados da FCPS (*Fundamental Clustering Problems Suite*) possui 10 bases de dados artificiais relativamente simples e de baixa dimensionalidade (2 e 3 dimensões), mas que demonstram alguns dos principais problemas enfrentados pelos algoritmos de agrupamento mais comuns [Ultsch, 2005].

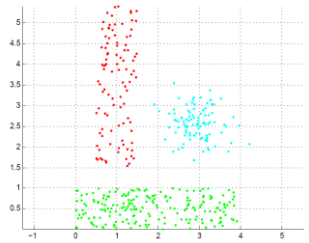
As bases de dados da FCPS servem como conjunto de testes para novos algoritmos de agrupamento. Cada uma das bases ilustra um determinado tipo de dificuldade, como por exemplo, agrupamentos não linearmente separáveis, variações na densidade dos agrupamentos e existência de valores discrepantes (*outliers*), que são comumente enfrentadas por algoritmos tradicionais como Ligação Simples, Ward e *k*-médias. A utilização dessas bases de dados é interessante não apenas pelas dificuldades de agrupamento presentes nos dados, mas principalmente por sua baixa dimensionalidade permitir a avaliação visual dos resultados.

A Figura 6.1 apresenta visualmente as bases de dados que compõem a FCPS. As principais características de cada uma das bases de dados da FCPS estão descritas na Tabela 6.1.

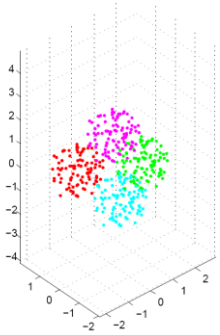
a) Hepta



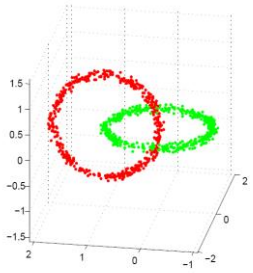
b) Lsun



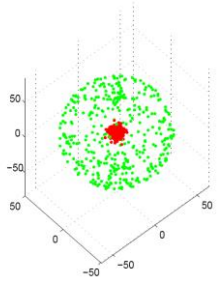
c) Tetra



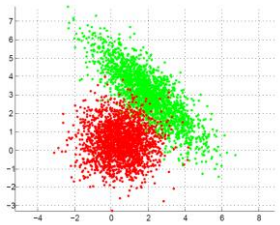
d) Chainlink



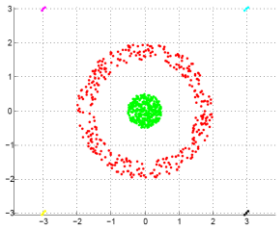
e) Atom



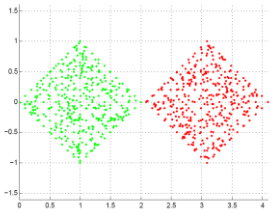
f) EngyTime



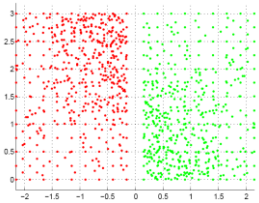
g) Target



h) TwoDiamonds



i) Wingnut



j) Golfball

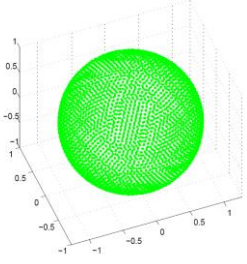


Figura 6.1: Bases de dados da FCPS [Ultsch, 2005].

Tabela 6.1: Descrição das bases de dados da FCPS

Nome	Dificuldade	Tamanho	Dimensão	Classes
a) Hepta	Agrupamentos esféricos bem definidos, mas com variâncias diferentes	212	3	7
b) Lsun	Agrupamentos com diferentes variâncias e distâncias intra-grupos	400	2	3
c) Tetra	Quatro agrupamentos com pequena distância entre eles, que quase se tocam	400	3	4
d) Chainlink	Dois anéis entrelaçados, não linearmente separáveis	1000	3	2
e) Atom	Dois agrupamentos concêntricos, com diferentes variâncias e não linearmente separáveis	800	3	2
f) EngyTime	Dois agrupamentos formados a partir de uma mistura gaussiana, com sobreposição de pontos	4096	2	2
g) Target	Presença de exceções (*outliers*)	760	2	6
h) TwoDiamonds	Dois agrupamentos em forma de losangos que quase se tocam e bordas definidas pela densidade	800	2	2
i) Wingnut	Dois agrupamentos em forma de retângulos, com diferentes densidades intra-grupo	1070	2	2
j) Golfball	Nenhum agrupamento	4002	3	0

6.1.2 Base de Dados Iris

A base de dados Iris é uma das mais utilizadas em testes de reconhecimento de padrões e aprendizado de máquina e faz parte do repositório de dados da Universidade da Califórnia em Irvine, EUA, denominado *Machine Learning Repository* [Newman, 1998].

A base possui 150 instâncias contendo medidas de largura e comprimento de sépalas e pétalas de três espécies da flor Iris: *Setosa*, *Versicolor* e *Virginica*. Cada instância da base possui 4 atributos, correspondentes às medidas das pétalas e sépalas e a base possui 50 registros de cada uma das espécies. Uma pequena amostra do conteúdo dessa base de dados é apresentada na Tabela 6.2.

Tabela 6.2: Estrutura da base de dados Iris

Nº	Comprimento da Sépala	Largura da Sépala	Comprimento da Pétala	Largura da Pétala	Classe
1	5.1	3.5	1.4	0.2	Setosa
2	4.9	3.0	1.4	0.2	Setosa
3	4.7	3.2	1.3	0.2	Setosa
...	...	...	...	...	...
150	5.9	3.0	5.1	1.8	Virginica

Para os experimentos aqui descritos, o atributo de classe foi ignorado, uma vez que o objetivo do algoritmo é conseguir agrupar as espécies semelhantes a partir das suas medidas de pétala e sépala.

Apesar de ser uma base de dados relativamente fácil de ser agrupada, a base Iris possui uma particularidade interessante: a classe *Setosa* pode ser linearmente separada das demais, porém as classes *Versicolor* e *Virginica* não são linearmente separáveis, o que na maioria das vezes é responsável pela ocorrência de erros de atribuição em algoritmos de análise de agrupamentos.

A Figura 6.2 mostra a relação entre cada par de variáveis da base de dados. Cada classe está representada por uma cor diferente, a saber: *Setosa*, em vermelho; *Versicolor*, em azul e *Virginica*, em verde. É importante verificar que a classe *Setosa* é facilmente separável das demais, o que pode ser observado em todos os gráficos. No entanto, dependendo de quais atributos sejam analisados em conjunto, nem sempre é possível separar linearmente as classes *Versicolor* e *Virginica*.

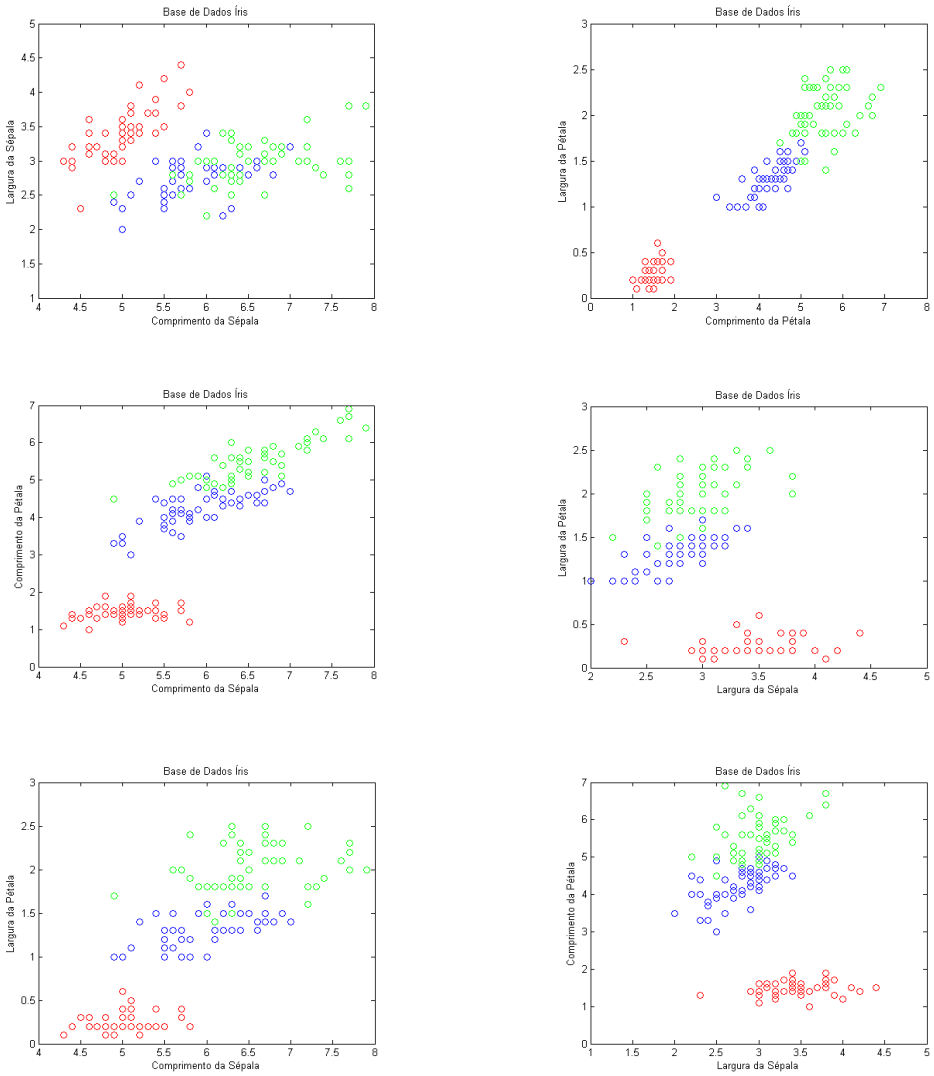


Figura 6.2: Relação entre atributos da base de dados Iris.

A Tabela 6.3 mostra os valores mínimos, máximos e médios de cada um dos atributos, separados por classe. Novamente é possível observar que existem algumas sobreposições nas faixas de valores, principalmente entre as classes *Versicolor* e *Virginica*.

Tabela 6.3: Valores limites dos atributos da base de dados Iris

Classe	Valor	Comp. da Sépala	Larg. da Sépala	Comp. da Pétala	Larg. da Pétala
Setosa	**Mínimo**	4.30	2.30	1.00	0.10
	Médio	5.01	3.42	1.46	0.24
	Máximo	5.80	4.40	1.90	0.60
Versicolor	**Mínimo**	4.90	2.00	3.00	1.00
	Médio	5.94	2.77	4.26	1.33
	Máximo	7.00	3.40	5.10	1.80
Virginica	**Mínimo**	4.90	2.20	4.50	1.40
	Médio	6.59	2.97	5.55	2.03
	Máximo	7.90	3.80	6.90	2.50

6.1.3 Base de Dados Wine

A base de dados Wine contém 178 instâncias, cada uma delas composta por 13 atributos que correspondem aos resultados de análises químicas realizadas em três tipos de vinhos que são produzidos na mesma região da Itália, provenientes de diferentes cultivos. Os atributos incluem teor alcoólico, acidez, alcalinidade, intensidade de cor, entre outros. A base possui 59 instâncias da primeira classe, 71 instâncias da segunda classe e 48 instâncias da terceira. Uma pequena amostra do conteúdo da base de dados Wine é apresentado na Figura 6.3, onde os atributos são representados por $\boldsymbol{a}_1$, $\boldsymbol{a}_2$, ..., $\boldsymbol{a}_{13}$.

	$\boldsymbol{a_1}$	$\boldsymbol{a_2}$	$\boldsymbol{a_3}$	...	$\boldsymbol{a_{13}}$	Classe
1	14.23	1.71	2.43	...	1065	'1'
2	13.20	1.78	2.14	...	1050	'1'
3	13.16	2.36	2.67	...	1185	'1'
...	...	...	...	...	...	...
178	14.13	4.1	2.74	...	560	'3'

Figura 6.3: Estrutura da base de dados Wine.

Como nos demais experimentos aqui realizados, o atributo de classe foi ignorado durante a etapa de treinamento, tendo sido utilizado apenas na averiguação dos resultados. A base de dados Wine é relativamente fácil de ser agrupada, uma vez que possui agrupamentos bem definidos e linearmente separáveis, sendo uma das mais utilizadas em

tarefas de agrupamento. Ela também faz parte do repositório de dados da UCI (*University of California, Irvine*) [Newman, 1998].

6.1.4 Base de Dados Mushroom

A base de dados Mushroom contém a descrição de exemplos hipotéticos de 23 espécies de cogumelos, divididos em dois grupos: comestíveis ou não-comestíveis. A base possui 8.124 instâncias, sendo 4.208 pertencentes ao primeiro grupo e 3.916 pertencentes ao segundo grupo. Cada instância contém 22 atributos, além do atributo da classe, que foi ignorado durante o treinamento. Os dados de cada instância incluem diversas informações sobre a espécie, tais como: o formato, aparência e cor do caule e da parte superior (chapéu), odor característico e local onde é encontrado. Originalmente, a base de dados possui apenas valores categóricos. Ou seja, cada atributo possui um conjunto de possíveis valores, representados por caracteres, conforme apresentado na Tabela 6.4.

Algoritmos como o SOM e *k*-médias não trabalham diretamente com dados categóricos, por isso esses dados precisaram ser convertidos em numéricos (binários), a fim de poderem ser tratados adequadamente, uma vez que o algoritmo calcula a similaridade entre objetos utilizando distância euclidiana. Como a base de dados Mushroom possui apenas dados categóricos, foi necessária a realização de uma etapa de pré-processamento em toda a base, a fim de que os valores categóricos fossem convertidos em valores binários mutuamente excludentes, antes da aplicação dos algoritmos.

A conversão foi realizada da seguinte forma: para cada possível valor categórico existente no atributo, foi criada uma coluna adicional na matriz de dados. O valor de cada instância em cada uma das colunas foi atribuído como sendo 0 ou 1, de acordo com o valor categórico da instância para aquele atributo. Um exemplo do processo de conversão que foi realizado, em relação ao primeiro atributo da base, está ilustrado na Figura 6.4. Após o pré-processamento, o número de colunas da base de dados aumentou consideravelmente, subindo de 22 para 117.

#	valor
1	x
2	x
3	b
...	...
8123	k
8124	x

→

#	b	c	x	f	k	s
1	0	0	1	0	0	0
2	0	0	1	0	0	0
3	1	0	0	0	0	0
...	...	...	...	...	...	...
8123	0	0	0	0	1	0
8124	0	0	1	0	0	0

Figura 6.4: Exemplo de conversão dos atributos da base de dados Mushroom.

A base de dados Mushroom também pode ser encontrada no repositório de dados da UCI [Newman, 1998].

Tabela 6.4: Descrição dos atributos da base de dados Mushroom

Atributo	*Possíveis Valores*
Formato do chapéu	***b*** – sino; ***c*** – cônico; ***x*** – convexo; ***f*** – plano; ***k*** – com saliências; ***s*** – rebaixado
Superfície do chapéu	***f*** – fibroso; ***g*** – ranhuras; ***y*** – escamoso; ***s*** – suave
Cor do chapéu	***n*** – marrom; ***b*** – amarelo claro; ***c*** – canela; ***g*** – cinza; ***r*** – verde; ***p*** – rosa; ***u*** – violeta; ***e*** – vermelho; ***w*** – branco; ***y*** - amarelo
Manchas	***t*** – com manchas; ***f*** – sem manchas
Odor	***a*** – amêndoas; ***l*** – anis; ***c*** – creosoto; ***y*** – peixe; ***f*** – mau cheiro; ***m*** – mofo; ***n*** – nenhum; ***p*** – picante; ***s*** - temperado
Ligação das brânquias	***a*** – ligada; ***d*** – descendente; ***f*** – livre; ***n*** – fendida
Espaçamento entre as brânquias	***c*** – fechado; ***w*** – preenchido; ***d*** – distante
Tamanho das brânquias	***b*** – larga; ***n*** – estreita
Cor das brânquias	***k*** - preto; ***n*** – marrom; ***b*** – amarelo claro; ***h*** – chocolate; ***g*** – cinza; ***r*** – verde; ***o*** – laranja; ***p*** – rosa; ***u*** – violeta; ***e*** – vermelho; ***w*** – branco; ***y*** - amarelo
Formato do caule	***e*** – alargado; ***t*** – estreitado
Raiz do caule	***b*** – bulboso; ***c*** – clava; ***u*** – copo; ***e*** – uniforme; ***z*** – forma de talo; ***r*** – enraizada; ***?*** – valor ausente
Superfície do caule acima do anel	***f*** – fibroso; ***y*** – escamoso; ***k*** – sedoso; ***s*** – suave
Superfície do caule abaixo do anel	***f*** – fibroso; ***y*** – escamoso; ***k*** – sedoso; ***s*** – suave
Cor do caule acima do anel	***n*** – marrom; ***b*** – amarelo claro; ***c*** – canela; ***g*** – cinza; ***o*** – laranja; ***p*** – rosa; ***e*** – vermelho; ***w*** – branco; ***y*** - amarelo
Cor do caule abaixo do anel	***n*** – marrom; ***b*** – amarelo claro; ***c*** – canela; ***g*** – cinza; ***o*** – laranja; ***p*** – rosa; ***e*** – vermelho; ***w*** – branco; ***y*** - amarelo
Tipo de véu	***p*** – parcial; ***u*** – total
Cor do véu	***n*** – marrom; ***o*** – laranja; ***w*** – branco; ***y*** - amarelo
Número de anéis	***n*** – nenhum; ***o*** – um; ***t*** – dois
Tipo de anel	***c*** – teia de aranha; ***e*** – ofuscado; ***f*** – brilhante; ***l*** – largo; ***n*** – nenhum; ***p*** – pendente; ***s*** – forrado; ***z*** – zona
Cor do esporão	***k*** - preto; ***n*** – marrom; ***b*** – amarelo claro; ***h*** – chocolate; ***r*** – verde; ***o*** – laranja; ***u*** – violeta; ***w*** – branco; ***y*** - amarelo
População	***a*** – abundante; ***c*** – agrupada; ***n*** – numerosa; ***s*** – espalhada; ***v*** – alguns; ***y*** – solitário
Habitat	***g*** – grama; ***l*** – folhas; ***m*** – pasto; ***p*** – estrada; ***u*** – urbano; ***w*** – lixo; ***d*** – madeira

6.1.5 Base de Dados Wages

A base de dados Wages consiste de um subconjunto extraído aleatoriamente a partir dos dados da pesquisa *Current Population Survey*, realizada nos Estados Unidos, em 1985. A base possui 534 instâncias, contendo 11 atributos cada uma. Não há atributo de classe, por não se tratar de uma base de dados destinada a tarefas de classificação. Os

atributos incluem informações sobre salário, sexo, número de anos de formação acadêmica, número de anos de experiência, ocupação e região onde reside, entre outros, conforme mostra a Tabela 6.5. Também estão representados os valores médios para os atributos numéricos e a distribuição estatística para os atributos categóricos.

Essa base possui tanto dados categóricos quanto numéricos. Da mesma maneira que foi feito no exemplo anterior, dados categóricos foram convertidos em binários mutuamente excludentes a fim de poderem ser tratados adequadamente, uma vez que os algoritmos utilizados neste trabalho calculam a similaridade entre objetos utilizando distância euclidiana. Por isso, foi necessário realizar uma etapa de pré-processamento dos dados nessa base de dados antes de aplicar o algoritmo sobre as mesmas.

Por exemplo, o atributo 'occupation' é categórico e refere-se ao tipo de trabalho desempenhado pelo empregado. O atributo possui originalmente seis valores possíveis. Após o pré-processamento, o atributo foi substituído por seis novas colunas com valores binários (e mutuamente excludentes), que indicam a qual setor o empregado pertence. Após o pré-processamento aplicado à base de dados para conversão dos dados categóricos em dados numéricos, o número de colunas aumentou para 20.

Tabela 6.5: Descrição dos atributos da base de dados Wages

Atributo	*Descrição*	*Valores*	*Valor Médio*
1. Education	Número de anos de estudo	inteiro	13.0
2. South	Indica se o indivíduo reside da região Sul dos EUA	**1** – morador do sul **0** – de outra região	29% 71%
3. Sex	Indica o sexo do indivíduo	**1** – feminino **0** – masculino	46% 54%
4. Experience	Anos de experiência profissional	inteiro	17.8
5. Union	Indica se o indivíduo é membro de alguma associação/sindicato	**1** – membro **0** – não membro	18% 82%
6. Wage	Salário (em US$/hora)	real	US$ 9.02
7. Age	Idade do indivíduo	inteiro	36.8
8. Race	Raça do indivíduo	**1** – outros **2** – origem hispânica **3** – branco	13% 5% 82%
9. Occupation	Ocupação do indivíduo	**1** – administrativo **2** – vendas **3** – religioso **4** – serviços **5** – profissional liberal **6** – outros	10% 7% 18% 16% 20% 29%
10. Setor	Setor onde o indivíduo trabalha	**1** – outros **2** – indústria/produção **3** – construção civil	77% 19% 4%
11. Married	Estado civil	**0** – não casado **1** – casado	34% 66%

A base de dados Wages está disponível no repositório de dados da UCI [Newman, 1998].

6.1.6 Base de Dados Biscoitos

A base de dados Biscoitos contém dados referentes a uma pesquisa de mercado realizada no Estado de São Paulo, durante o primeiro semestre de 2004, com o objetivo de avaliar, por meio de atributos comportamentais, hábitos e atitudes de consumidores de biscoitos daquela população. A base inclui consumidores de ambos os sexos, com idades entre 13 a 60 anos. Um estudo sobre segmentação desta base de dados foi originalmente publicado em [Marques, 2008].

A base contém 750 instâncias, cada uma delas contendo 17 atributos, sendo 13 atributos numéricos e 4 categóricos. A pesquisa foi conduzida na forma de entrevistas, onde os consumidores respondiam a 13 afirmações, conforme apresentado na Tabela 6.6, através de um grau de concordância ou discordância, de acordo com a escala unidimensional de Likert [Likert, 1932], usualmente aplicada em mensuração de pesquisas de opinião e atitudes de consumidores. A pesquisa utilizou 5 pontos na escala: (1 – discordo totalmente; 2 – discordo em parte; 3 – não concordo nem discordo; 4 – concordo em parte; 5 – concordo totalmente).

Tabela 6.6: Variáveis utilizadas na pesquisa da base de dados Biscoito

Atributo	Afirmação Relacionada
a_1	Sempre compro produtos em promoção, mesmo que nunca tenha experimentado
a_2	Faço qualquer sacrifício para manter um bom visual
a_3	Sempre consumo produtos *light* e/ou *diet*
a_4	Não compro produtos *light* e/ou *diet* porque são muito caros
a_5	Não me importo em pagar mais por produtos de qualidade
a_6	Busco sempre fazer refeições saudáveis
a_7	Sou viciado em *snacks*, não passo um dia sem comer biscoitinhos e salgadinhos
a_8	Como somente alimentos naturais
a_9	Sempre procuro alimentos enriquecidos com vitaminas
a_{10}	Sempre leio artigos sobre a saúde
a_{11}	Não me importo em gastar a maior parte da minha renda com alimentação
a_{12}	Prefiro fazer ginástica a comer menos
a_{13}	Para mim, saúde e alimentação estão intimamente ligadas

Tabela 6.7: Variáveis categóricas da base de dados Biscoito

Atributo	Variável	Possíveis Valores
a_{14}	Idade	1 – 13 a 17 anos (jovens) 2 – 18 a 25 anos (jovem/adulto) 3 – 26 a 37 anos (adulto) 4 – 38 a 48 anos (adulto/idoso) 5 – 49 a 60 anos (idoso)
a_{15}	Classe	1 – classes A e B 2 – classe C 3 – classe D
a_{16}	Frequência de Consumo	1 – muito (3 a 7 vezes por semana) 2 – médio (1 a 2 vezes por semana) 3 – pouco (no máximo, 3 vezes por mês)
a_{17}	Perfil	1 – mães com filhos de 3 a 12 anos 2 – mulheres sem filhos de 3 a 12 anos 3 – homens

As quatro últimas questões da pesquisa possuíam respostas categóricas, dentro de determinadas fixas pré-fixadas. A Tabela 6.7 mostra as variáveis restantes e as faixas de valores consideradas para cada uma delas.

6.1.7 Base de Dados CAPES

A Coordenação de Aperfeiçoamento de Pessoal de Nível Superior (CAPES) é o órgão responsável pela avaliação dos cursos de pós-graduação *stricto sensu* (mestrado e doutorado) e da pesquisa científica e tecnológica no Brasil. A CAPES cumpre um papel de fundamental importância para o desenvolvimento da pós-graduação no País, através de ações que incluem a avaliação dos cursos de pós-graduação; o acesso e divulgação da produção científica; investimentos na formação de recursos humanos de alto nível no país e exterior e promoção da cooperação científica internacional [CAPES, 2009].

A avaliação dos cursos de pós-graduação é realizada inicialmente por uma Comissão de Avaliação (CA), que produz um relatório contendo um conjunto de conceitos para cada um dos itens avaliados. Os conceitos variam dentro de uma escala de valores: MB – Muito Bom, B – Bom, R – Regular, F – Fraco, D – Deficiente e NA – Não Aplicável. Além disso, o relatório inclui uma justificativa para cada conceito atribuído. Em uma segunda avaliação, realizada posteriormente, o Comitê Técnico Científico (CTC) analisa o relatório produzido pela CA e valida o parecer proposto, podendo inclusive, aumentar ou diminuir o conceito que foi atribuído.

A coleta dos dados é feita todos os anos, porém os resultados são consolidados e publicados a cada três anos, quando é feita a renovação de reconhecimento dos cursos. Assim, os dados publicados no relatório trienal 2007 e que foram utilizados nos experimentos, são referentes ao período de 2004 a 2006. A nota final atribuída ao programa é expressa através de um valor que varia entre 1 e 7, mas apenas os cursos com nota entre 3 e 7 são recomendados pela CAPES. As notas fornecidas pela CA e pelo CTC não foram consideradas durante o processo de análise dos agrupamentos da base.

A base de dados considerada nos experimentos realizados neste trabalho inclui apenas os resultados individuais dos cursos de pós-graduação da área de Ciência de Computação. Foram analisadas 39 instituições, sendo considerados 21 atributos de cada uma delas. Os atributos são divididos em 5 grupos (ou quesitos), que possuem diferentes pesos na avaliação final, conforme descritos na Tabela 6.8. A Tabela 6.9 detalha as notas obtidas pelas instituições em cada quesito avaliado.

Durante a fase de pré-processamento dos dados, houve a necessidade de conversão dos valores categóricos em valores numéricos. Como os valores categóricos estão distribuídos em uma escala ordenada, a conversão é feita simplesmente substituindo cada valor categórico por um valor inteiro, obedecendo à escala original. Assim, os valores finais após a conversão são: Muito Bom – 5, Bom – 4, Regular – 3, Fraco – 2, Deficiente – 1. Onde havia o conceito NA – Não Aplicável, a instituição foi desconsiderada na análise total, tendo sido considerada apenas nas análises parciais.

Tabela 6.8: Descrição e peso dos atributos da base de dados CAPES

Quesitos e Atributos	***Peso***
Proposta do programa	**0%**
a_1: áreas de concentração, linhas de pesquisa, projetos	0%
a_2: estrutura curricular	0%
a_3: infra-estrutura para ensino, pesquisa e extensão	0%
Corpo docente	**25%**
a_4: formação do corpo docente (formação, experiência, etc.)	25%
a_5: adequação do corpo docente permanente	20%
a_6: perfil e compatibilidade do corpo docente com a proposta do programa	15%
a_7: atividade docente e distribuição da carga letiva	10%
a_8: participação dos docentes no ensino e pesquisa da graduação	10%
a_9: participação dos docentes em pesquisas e desenvolvimento de projetos	20%
Corpo discente, teses e dissertações	**30%**
a_{10}: orientação de teses e dissertações concluídas	25%
a_{11}: relação orientador/discente	10%
a_{12}: participação discente na produção científica	10%
a_{13}: qualidade das teses e dissertações em relação à publicações	25%
a_{14}: outros indicadores de qualidade das teses e dissertações	20%
a_{15}: eficiência do programa (tempo de formação e percentual de bolsistas)	10%
Produção intelectual	**35%**
a_{16}: publicações qualificadas por docente permanente	75%
a_{17}: distribuição das publicações qualificadas em relação ao corpo docente	20%
a_{18}: outras produções consideradas relevantes	5%
Inserção social	**10%**
a_{19}: inserção e impacto regional/nacional	40%
a_{20}: integração e cooperação com outros programas	30%
a_{21}: visibilidade do programa	30%

Tabela 6.9: Avaliação dos cursos de pós-graduação da base de dados CAPES

Instituição	Proposta			Corpo Docente						Corpo Discente						Prod. Intelectual			Inserção Social			Conceito	
	a_1	a_2	a_3	a_4	a_5	a_6	a_7	a_8	a_9	a_{10}	a_{11}	a_{12}	a_{13}	a_{14}	a_{15}	a_{16}	a_{17}	a_{18}	a_{19}	a_{20}	a_{21}	CA	CTC
FEESR	R	B	R	B	B	B	B	B	R	R	B	R	R	B	R	F	R	R	R	F	B	3	3
FESP/UPE	D	D	R	R	B	B	B	MB	B	NA	B	NA	NA	NA	NA	R	R	F	NA	R	B	3	3
IME	B	R	B	R	R	R	B	B	R	R	R	B	R	R	B	R	R	R	R	R	R	3	3
PUC/RJ	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	7	7
PUC/MG	R	B	B	R	B	B	B	R	R	NA	MB	B	NA	NA	NA	R	F	F	NA	F	R	3	3
PUC/PR	B	B	B	B	B	B	B	B	B	R	B	B	B	B	R	B	R	B	B	B	B	4	4
PUC/RS	B	B	MB	MB	B	B	B	B	B	B	B	B	B	B	MB	B	B	B	B	B	B	4	4
UCPEL	B	B	R	R	B	R	R	B	B	NA	R	NA	NA	NA	NA	R	R	R	R	R	R	3	3
UEM	B	B	B	B	R	B	B	B	F	R	B	F	R	B	R	R	R	R	B	R	R	3	3
UFAM	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	B	4	4
UFC	B	B	B	B	B	B	B	B	B	R	B	B	B	B	R	B	B	B	B	B	B	4	4
UFCG	MB	B	MB	B	MB	MB	MB	B	B	MB	B	B	MB	B	MB	B	B	B	MB	MB	B	5	4
UFES	B	B	B	B	B	B	R	B	B	B	MB	B	R	B	B	R	R	R	B	R	R	4	3
UFF	B	MB	MB	MB	MB	MB	B	R	MB	MB	MB	B	B	B	B	MB	B	B	B	MB	MB	5	5
UFG	B	B	B	B	B	MB	B	F	B	B	D	R	R	R	B	R	R	R	B	F	B	3	3
UFMG	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	6	6
UFMS	R	B	B	B	B	B	B	B	MB	R	B	B	B	R	R	B	R	F	B	B	B	4	4
UFPA	R	R	R	B	R	B	R	R	R	NA	NA	NA	NA	NA	NA	F	R	R	B	R	R	3	3
UFPB	B	B	MB	B	R	B	F	R	B	NA	B	NA	NA	NA	NA	R	R	NA	B	R	B	3	3
UFPE	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	6	6
UFPR	B	B	B	B	B	B	B	B	B	MB	B	B	B	B	B	B	B	B	B	B	B	4	4
UFRGS	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	6	6
UFRJ	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	7	7
UFRJ	B	MB	MB	MB	MB	MB	B	R	MB	MB	MB	F	R	B	R	MB	B	F	R	B	B	5	4
UFRN	B	B	B	B	B	B	B	B	B	MB	B	B	B	B	B	B	B	B	B	B	B	4	4
UFSC	B	B	MB	B	B	B	B	B	MB	R	B	B	B	B	R	R	R	R	B	B	R	4	3
UFSCAR	MB	MB	MB	B	MB	MB	MB	B	MB	MB	MB	B	MB	B	B	B	MB	MB	B	B	MB	4	4
UFU	B	B	B	B	B	B	B	B	R	B	B	B	R	B	B	R	R	R	B	B	B	4	3
UFV	R	B	B	R	B	B	R	R	F	R	B	B	R	B	R	F	R	D	B	F	R	3	3
UNB	B	B	B	B	R	B	R	B	R	B	B	B	B	B	R	R	R	R	R	R	R	3	3
UNESP/SJRP	B	B	B	F	R	F	B	B	R	NA	R	NA	NA	NA	NA	F	R	R	B	R	R	3	3
UNICAMP	MB	MB	MB	MB	MB	MB	MB	MB	B	B	MB	R	B	MB	B	MB	MB	B	MB	MB	MB	6	5
UNIFACS	R	B	R	B	R	B	F	B	R	NA	R	NA	NA	NA	NA	F	F	R	B	B	B	3	3
UNIFOR	B	MB	MB	B	B	B	MB	MB	MB	B	B	B	MB	B	F	B	MB	B	B	B	B	4	4
UNIMEP	R	R	B	B	R	B	R	B	R	B	B	R	R	R	B	R	R	R	B	B	B	3	3
UNISANTOS	B	MB	R	B	R	B	R	F	B	R	B	R	F	F	R	B	R	B	B	R	R	3	3
UNISINOS	B	B	B	B	B	B	R	B	B	B	B	B	B	B	B	B	R	B	B	R	B	4	4
USP	MB	MB	B	MB	MB	MB	B	R	B	R	B	B	MB	MB	R	MB	B	MB	MB	MB	B	6	5
USP/SC	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	MB	5	5

6.1.8 Base de Dados HiperCubo

A base de dados HiperCubo é composta por um conjunto de pontos que formam um hipercubo no espaço 4D. Um hipercubo é um polígono com 16 vértices e 32 arestas no $\Re^4$. A base de dados é composta por 1600 pontos, distribuídos ao longo dos 16 vértices que formam o polígono, sendo que cada ponto consiste em um vetor com 4 dimensões. A Figura 6.5 apresenta uma projeção de um hipercubo sobre o espaço tridimensional, a partir de um ponto exterior.

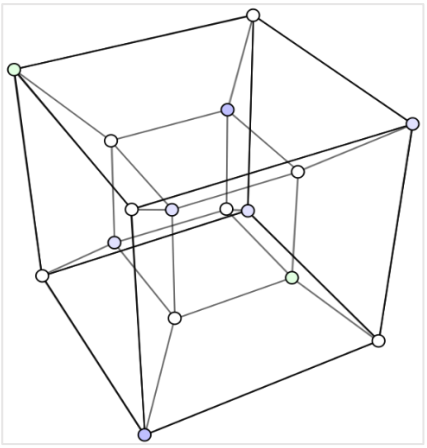


Figura 6.5: Hipercubo projetado sobre um espaço tridimensional.

6.2 Metodologia Utilizada nos Experimentos

A fim de verificar a precisão da arquitetura proposta em tarefas de análise de agrupamentos sobre bases de dados distribuídas, foi realizada uma série de experimentos com diferentes conjuntos de dados, formas de partição e combinação de algoritmos. A proposta foi comparar os resultados obtidos com a arquitetura partSOM em relação aos que seriam obtidos pelas abordagens tradicionais, em que os dados são centralizados antes de serem agrupados e, assim, comprovar a eficácia da abordagem proposta.

Os experimentos iniciais foram realizados utilizando-se bases de dados didáticas, amplamente conhecidas na literatura e bastante utilizadas em tarefas de aprendizado de máquina, sobre as quais é possível encontrar valores de referência para diversos algoritmos, o que facilitou a avaliação dos resultados obtidos. Porém, como o foco do trabalho é a segmentação de mercado, os demais experimentos utilizaram bases de dados reais de mercado, como uma forma de mensurar e demonstrar as vantagens obtidas com a utilização da arquitetura sobre este domínio.

Em todos os testes realizados, foram utilizados diversos critérios comparativos para validar a arquitetura partSOM. Foram considerados três índices de validação de agrupamentos comumente utilizados na literatura: o erro de quantização médio, o erro topológico dos mapas auto-organizáveis e o índice de Davies-Bouldin. A eficácia da arquitetura também foi medida através da taxa de erros em tarefas de classificação.

Além disso, foram considerados ainda outros critérios comparativos, como a representação gráfica das bases de dados em baixa dimensionalidade e a comparação visual dos mapas treinados e da matriz-U, sempre considerando os algoritmos tradicionais e a abordagem proposta.

Em relação aos índices de validação, foram utilizados os pacotes *Cluster validation toolbox for estimating the number of clusters* (CVT-NC), versão 2.0 [Wang, 2007a], disponível em: http://www.mathworks.com/matlabcentral/fileexchange/13916 e *Cluster Validity Analysis Platform* (CVAP), versão 3.5 [Wang, 2007b], disponível em: http://www.mathworks.com/matlabcentral/fileexchange/14620. Todos os experimentos foram efetuados utilizando-se o pacote SOM Toolbox 2.0 [Vesanto, 2000], disponível em: http://www.cis.hut.fi/projects/somtoolbox/download/.

Os algoritmos SOM e *k*-médias são não determinísticos, o que significa que cada execução do algoritmo produz um resultado diferente, mesmo que sejam utilizados os mesmos parâmetros e o mesmo conjunto de dados. Isso acontece devido à inicialização aleatória dos valores iniciais dos centróides que compõem o *codebook*.

No entanto, Kohonen (2001) sugere inicializar o mapa distribuindo os neurônios de forma ordenada ao longo de um subespaço linear utilizando-se os k componentes principais da matriz de autocorrelação do conjunto de entrada. Assim, se os demais parâmetros forem mantidos, essa forma de inicialização permite obter os mesmos resultados a cada execução do algoritmo, por utilizar sempre os mesmos valores que são calculados a partir dos dados de entrada. Nos experimentos aqui realizados, foi utilizada a inicialização linear em todas as execuções do algoritmo SOM.

Vrusias *et al*. (2007) apresentam diversas considerações a respeito do tamanho dos mapas, parâmetros de treinamento e tamanho dos conjuntos de dados no treinamento distribuído de redes SOM. Tais conclusões foram consideradas durante o particionamento dos dados e na construção dos mapas.

A matriz-U permite visualizar os resultados obtidos em um processo de análise de agrupamentos utilizando o algoritmo SOM. No entanto, a matriz-U é uma imagem bidimensional composta a partir de uma escala de níveis de cinza que indicam a distância entre os neurônios do mapa e a interpretação dos seus resultados em aplicações reais pode ser uma tarefa complexa, uma vez que não há uma delineação clara da fronteira entre os agrupamentos [Costa, 1999]. Por isso, em alguns experimentos aplicou-se o algoritmo *k*-médias sobre os mapas treinados a fim de detectar com mais facilidade os agrupamentos existentes e melhorar a visualização dos resultados, conforme proposto em [Vesanto, 2000].

6.3 Experimentos Realizados

Os experimentos realizados com a arquitetura partSOM visam avaliar o seu desempenho, comparando os resultados obtidos com as abordagens tradicionais. A seguir, são descritos os resultados individuais para cada um dos experimentos realizados.

6.3.1 Bases de Dados da FCPS

Dentre as bases de dados que compõem a FCPS, foram selecionadas a Hepta e a EngyTime para serem utilizadas nos experimentos. A primeira, pela maior quantidade de agrupamentos e a segunda, pela maior quantidade de instâncias. Em virtude da baixa dimensionalidade, as bases da FCPS foram utilizadas apenas nos experimentos com particionamento horizontal dos dados, uma vez que tais bases possuem apenas 2 ou 3 atributos.

A base de dados Hepta possui 212 pontos em 3 dimensões. Os pontos formam 7 agrupamentos esféricos bem definidos, mas com variâncias diferentes, sendo relativamente fácil de ser agrupada. Os experimentos aqui realizados consideraram apenas o particionamento horizontal da base, uma vez que verticalmente a base possui apenas 3 dimensões.

Inicialmente, a base foi analisada com o algoritmo SOM tradicional, considerando os 212 pontos simultaneamente. Foi utilizado um mapa plano de tamanho 9 x 8, totalizando 72 neurônios, dispostos em formato hexagonal. O SOM Toolbox, ferramenta utilizada nos experimentos, realiza o treinamento do SOM em duas fases, uma fase de aproximação e um ajuste fino. Em cada fase, a valor da vizinhança varia entre $\sigma_{inicial}$ e σ_{final}. Além disso, foi utilizada inicialização linear do mapa e treinamento em lote (*batch*), o que torna o algoritmo mais estável (determinístico) para diferentes execuções com o mesmo conjunto de dados. O tamanho do mapa e os parâmetros de treinamento foram automaticamente definidos pela ferramenta, a partir de uma análise estatística dos dados de entrada. Um resumo dos parâmetros utilizados no treinamento dos mapas do SOM tradicional e da arquitetura partSOM é apresentado na Tabela 6.10.

Tabela 6.10: Parâmetros utilizados no treinamento da base de dados Hepta

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	9 x 8	2	1	4	1	1	14
partSOM mapa 1/2	8 x 6	1	1	5	1	1	19
partSOM mapa 2/2	8 x 6	1	1	5	1	1	19
partSOM mapa final	9 x 8	2	1	4	1	1	14

Em seguida, a base foi aleatoriamente dividida em dois subconjuntos, cada um com metade dos dados. O algoritmo partSOM local foi aplicado a cada um dos subconjuntos, criando dois mapas parciais. Posteriormente, o algoritmo partSOM central foi aplicado sobre os mapas parciais. Nas etapas parciais, foram utilizados dois mapas planos, de tamanho 8 x 6, totalizando 48 neurônios, dispostos em formato hexagonal. Na etapa de centralização dos resultados foi utilizado um mapa plano de tamanho 9 x 8, também em formato hexagonal. Um resumo dos parâmetros utilizados no treinamento dos mapas da arquitetura partSOM é apresentado na Tabela 6.10

Os resultados obtidos com a arquitetura partSOM ficaram muito próximos ao método tradicional, onde todas as variáveis são analisadas simultaneamente. Ambas as abordagens conseguiram identificar corretamente os 7 agrupamentos do conjunto de dados Hepta. A Figura 6.5 apresenta os mapas obtidos com o algoritmo SOM tradicional e com a arquitetura proposta, devidamente segmentados com ajuda do algoritmo *k*-médias. Além disso, conforme pode ser observado na Figura 6.6, os mapas parciais conseguem capturar adequadamente a distribuição estatística dos dados do conjunto de entrada.

Foram utilizados ainda, como parâmetros de comparação entre as duas abordagens, alguns índices de validação de agrupamento: o erro de quantização médio (*qe*), medido entre cada elemento do conjunto de entrada e o seu *bmu* correspondente; o erro topológico médio (*te*), que mede a preservação da topologia do mapa e o índice de Davies-Bouldin (*db*), bastante utilizado para medir a compacticidade em processos de análise de agrupamento. Os valores médios das taxas de erro *qe* e *te* e do índice *db*, obtidas em 20 execuções dos algoritmos, juntamente com o desvio padrão (σ) dos valores obtidos nas medições estão apresentados na Tabela 6.11.

Tabela 6.11: Índices de validação para a base de dados Hepta

Algoritmo	**qe**		**te**		**db**	
	médio	**σ**	**médio**	**σ**	**médio**	**σ**
Som tradicional	0.2774	0.00%	0.0189	0.00%	0.9677	2.07%
Arquitetura partSOM	0.1132	1.33%	0.0202	2.10%	0.9202	3.24%

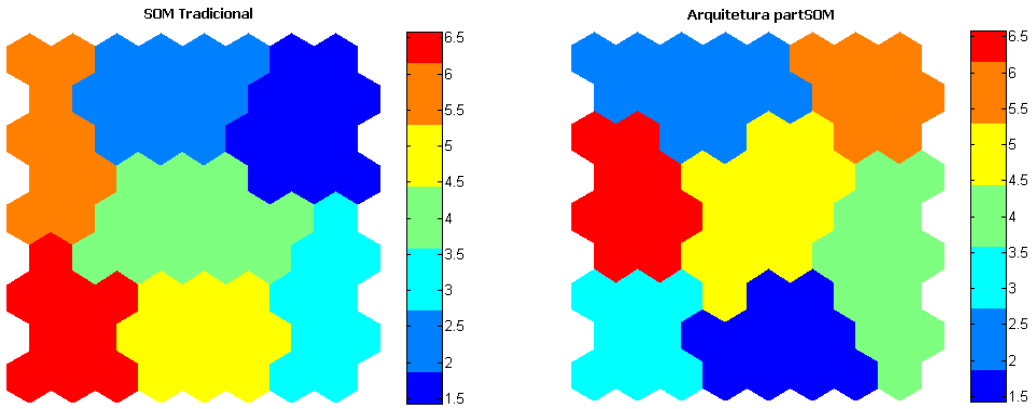


Figura 6.6: Mapas segmentados da base de dados Hepta.

A base de dados EngyTime possui 4096 pontos em 2 dimensões. A base possui 2 agrupamentos formados a partir de uma mistura gaussiana, com sobreposição de pontos, sendo mais difícil de ser agrupada do que a base do exemplo anterior. Os experimentos

aqui realizados consideraram apenas o particionamento horizontal da base, uma vez que verticalmente a mesma possui apenas 2 dimensões.

Inicialmente, a base foi analisada com o algoritmo SOM tradicional, considerando os 4096 pontos simultaneamente. Foi utilizado um mapa plano de tamanho 20 x 16, totalizando 320 neurônios, dispostos em formato hexagonal. Como no experimento anterior, foi utilizada inicialização linear do mapa e treinamento em lote (*batch*). O tamanho do mapa e os parâmetros de treinamento também foram definidos a partir da análise estatística dos dados de entrada. Um resumo dos parâmetros utilizados no treinamento dos mapas do SOM tradicional e da arquitetura partSOM é apresentado na Tabela 6.12.

Tabela 6.12: Parâmetros utilizados no treinamento da base de dados EngyTime

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	20 x 16	3	1	1	1	1	4
partSOM mapa 1/2	16 x 14	2	1	2	1	1	5
partSOM mapa 2/2	17 x 13	3	1	2	1	1	5
partSOM mapa final	20 x 16	3	1	1	1	1	4

Em seguida, a base foi dividida em dois subconjuntos, cada um com metade dos dados selecionados de forma aleatória, sem reposição. O algoritmo partSOM local foi aplicado a cada um dos subconjuntos, criando dois mapas hexagonais planos, com 224 e 221 neurônios respectivamente. Na etapa de centralização dos resultados, onde o algoritmo partSOM central foi aplicado sobre os mapas parciais, foi utilizado um mapa hexagonal plano de 320 neurônios, mesmo tamanho do utilizado na abordagem tradicional. Um resumo dos parâmetros utilizados no treinamento dos mapas da arquitetura partSOM é apresentado na Tabela 6.12.

Novamente, os resultados obtidos com a arquitetura partSOM ficaram muito próximos ao método tradicional, onde todas as variáveis são analisadas simultaneamente. Ambas as abordagens conseguiram identificar corretamente os 2 agrupamentos do conjunto de dados EngyTime.

A Figura 6.7 apresenta a dispersão dos pontos no espaço bidimensional. O gráfico da esquerda mostra os pontos originais e o gráfico da direita mostra os pontos do conjunto remontado a partir dos *codebooks* recebidos. Mesmo com um número menor de pontos distintos, em conseqüência do processo de quantização vetorial, é possível verificar que a distribuição estatística dos dados do conjunto de entrada é mantida.

Como parâmetros de comparação entre as duas abordagens, foram utilizados o erro de quantização médio (*qe*), o erro topológico médio (*te*) e o índice de Davies-Bouldin (*db*). Os valores médios de *qe* e *te* e do índice *db*, obtidas em 20 execuções dos algoritmos, juntamente com o desvio padrão (σ) calculado a partir dos valores obtidos nas medições estão apresentados na Tabela 6.13.

Tabela 6.13: Índices de validação para a base de dados EngyTime

Algoritmo	qe		te		db	
	médio	σ	médio	σ	médio	σ
Som tradicional	0.1090	0.00%	0.0327	0.00%	0.9809	0.00%
Arquitetura partSOM	0.0819	0.20%	0.0148	0.57%	0.9438	4.47%

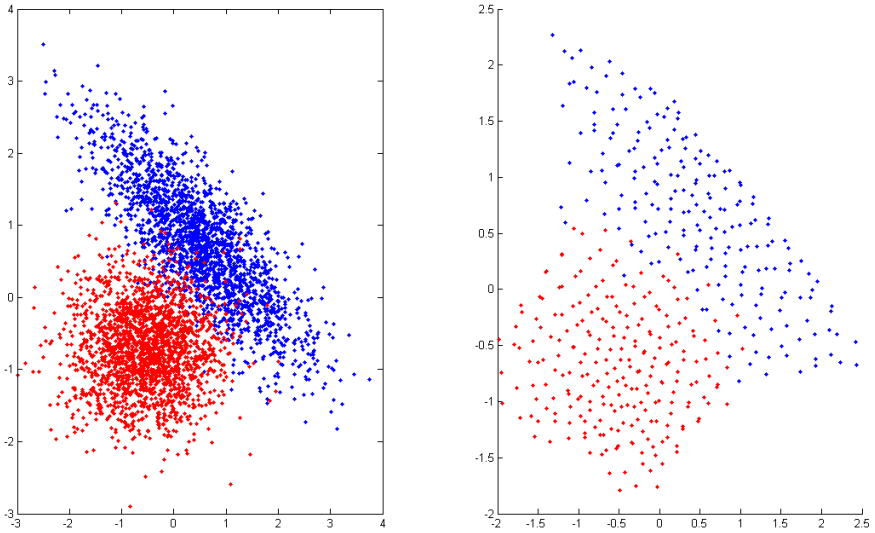


Figura 6.7: Dispersão dos pontos da base de dados Engytime.

6.3.2 Base de Dados Iris

Os quatro experimentos realizados com a base de dados Iris objetivaram analisar o desempenho da arquitetura partSOM tanto em relação ao particionamento horizontal quanto vertical. O particionamento horizontal foi realizado de duas formas distintas, com e sem sobreposição dos dados. Nos experimentos com particionamento vertical, analisou-se inicialmente uma abordagem utilizando o algoritmo SOM nas duas etapas e, posteriormente, uma combinação do algoritmo SOM na primeira etapa com o algoritmo *k*-médias na segunda etapa. Os resultados são apresentados em cada uma das seções a seguir.

6.3.2.1 Particionamento Horizontal sem Sobreposição

No primeiro experimento, a base de dados Iris foi horizontalmente dividida em três subconjuntos mutuamente exclusivos, cada um com 50 elementos aleatoriamente selecionados, sobre os quais foi aplicado o algoritmo SOM conforme a estratégia proposta. Na etapa central, foi aplicado o algoritmo SOM sobre os dados remontados. Para comparar os resultados, a base também foi analisada com o algoritmo SOM tradicional, considerando os 150 registros.

Nas duas abordagens foram utilizados mapas planos, com neurônios dispostos em formato hexagonal. Foi utilizada inicialização linear dos mapas e treinamento em lote. O tamanho do mapa e os parâmetros de treinamento estão apresentados na Tabela 6.14 e foram definidos pelo SOM Toolbox a partir de análises estatísticas dos dados.

Tabela 6.14: Parâmetros utilizados no treinamento da base de dados Iris

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	11 x 6	2	1	5	1	1	18
partSOM mapa 1/3	11 x 6	2	1	14	1	1	53
partSOM mapa 2/3	11 x 6	2	1	14	1	1	53
partSOM mapa 3/3	11 x 6	2	1	14	1	1	53
partSOM mapa final	11 x 6	2	1	5	1	1	18

Apesar de cada mapa parcial analisar apenas um terço dos dados da base original, cada um deles consegue extrair a distribuição estatística dos dados no espaço de entrada relativa aos elementos analisados. A Figura 6.8 mostra os mapas parciais obtidos após a etapa local de treinamento. É possível identificar certa separação entre as classes em cada um deles, embora esta separação ainda não seja boa o suficientemente para desconsiderar a fase seguinte do algoritmo.

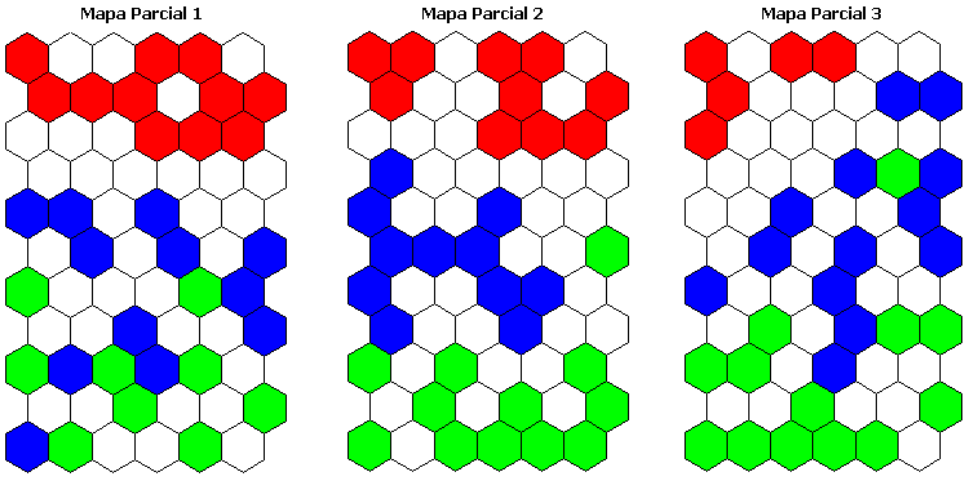


Figura 6.8: Mapas parciais da base de dados Iris.

A coloração dos mapas foi realizada a partir dos rótulos das classes existentes na base de dados original, apenas para ilustrar a segmentação dos mapas parciais a partir dos seus respectivos subconjuntos de dados. Cada elemento do conjunto de entrada é associado ao seu neurônio mais semelhante (*bmu*) dentro do *codebook*. Em seguida, para cada neurônio do mapa são selecionados os rótulos pertencentes aos elementos associados

a ele e é feita uma eleição para identificar qual a classe com maior representatividade, que é então associada ao neurônio.

A Figura 6.9 apresenta as matrizes-U obtidas com a abordagem tradicional e com a arquitetura partSOM. A Figura 6.10 apresenta os mapas finais, devidamente segmentados, onde foi feita a rotulagem dos neurônios a partir dos rótulos do conjunto de entrada, a fim de facilitar a visualização dos resultados. É possível identificar uma pequena diferença entre um mapa e outro, com relação a dois neurônios da região de fronteira entre os agrupamentos.

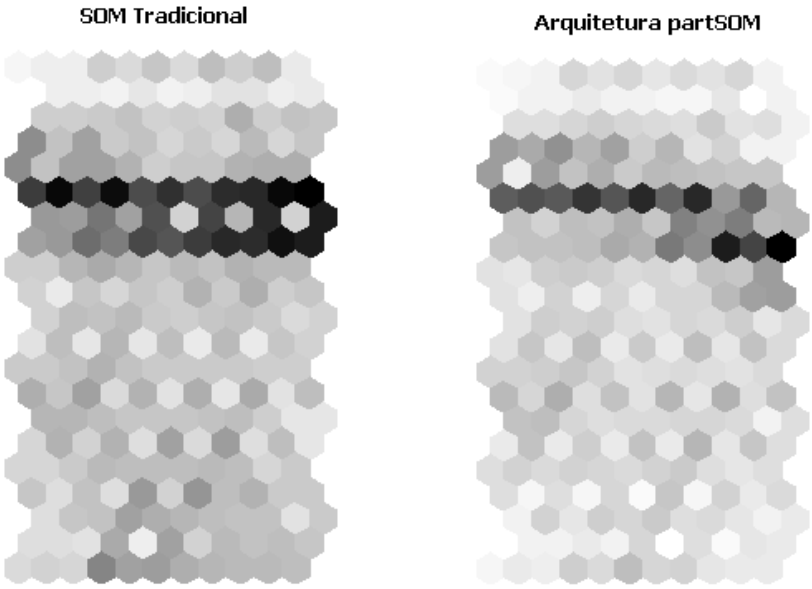


Figura 6.9: Matriz-U original e remontada da base de dados Iris.

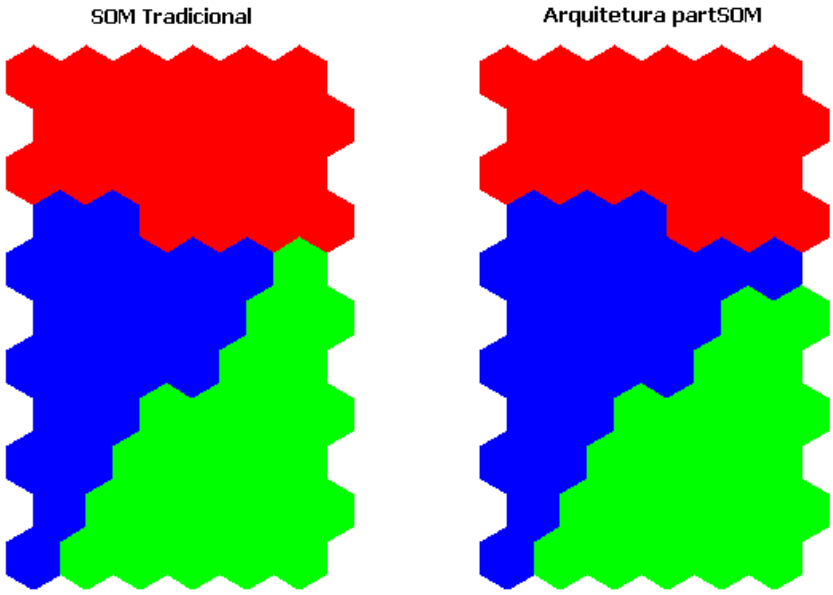


Figura 6.10: Mapas segmentados (original e remontado) da base de dados Iris.

A comparação entre a abordagem tradicional e a arquitetura proposta foi efetuada também através dos seguintes índices quantitativos: erro de quantização médio (*qe*), o erro topológico médio (*te*) e o índice de Davies-Bouldin (*db*). A Tabela 6.15 apresenta os valores médios e o desvio padrão (σ) calculados para os índices de validação obtidos após 20 execuções dos algoritmos.

Tabela 6.15: Índices de validação para a base de dados Iris

Algoritmo	qe		te		db	
	médio	σ	médio	σ	médio	σ
Som tradicional	0.3927	0.00%	0.0133	0.00%	0.7198	2.12%
Arquitetura partSOM	0.2306	1.13%	0.0267	1.42%	0.6664	2.62%

6.3.2.2 Particionamento Horizontal com Sobreposição

No segundo experimento, a base de dados Iris foi horizontalmente dividida em quatro subconjuntos parcialmente sobrepostos, cada um com 75 elementos aleatoriamente selecionados, sobre os quais foi aplicado o algoritmo SOM conforme a estratégia proposta. Na etapa central, foi aplicado o algoritmo SOM sobre os dados remontados. Os resultados foram comparados com a abordagem tradicional do algoritmo SOM, considerando os 150 registros.

Nas duas abordagens foram utilizados mapas planos, com neurônios dispostos em formato hexagonal. Foi utilizada inicialização linear dos mapas e treinamento em lote. O tamanho do mapa e os parâmetros de treinamento estão apresentados na Tabela 6.16 e foram definidos pelo SOM Toolbox a partir de análises estatísticas dos dados.

Tabela 6.16: Parâmetros utilizados no treinamento da base de dados Iris

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	11 x 6	2	1	5	1	1	18
partSOM mapa 1/4	8 x 5	1	1	6	1	1	22
partSOM mapa 2/4	8 x 5	1	1	6	1	1	22
partSOM mapa 3/4	11 x 4	2	1	6	1	1	24
partSOM mapa 4/4	11 x 4	2	1	6	1	1	24
partSOM mapa final	15 x 6	2	1	3	1	1	12

A comparação entre a abordagem tradicional e a arquitetura proposta também foi efetuada através mesmos índices quantitativos utilizados no exemplo anterior: erro de quantização médio (*qe*), erro topológico médio (*te*) e o índice de Davies-Bouldin (*db*). A Tabela 6.17 apresenta os valores médios e o desvio padrão (σ) calculados para os índices de validação obtidos após 20 execuções dos algoritmos.

Tabela 6.17: Índices de validação para a base de dados Iris

Algoritmo	qe		te		db	
	médio	σ	médio	σ	médio	σ
Som tradicional	0.3927	0.00%	0.0133	0.00%	0.7352	0.00%
Arquitetura partSOM	0.2081	3.98%	0.0320	2.88%	0.5981	3.04%

A Figura 6.11 apresenta uma projeção dos dados de entrada sobre a rede SOM treinada a partir da base de dados Iris. A imagem da esquerda apresenta a rede treinada com a abordagem original e a imagem da direita apresenta a rede treinada com a arquitetura partSOM. Foi utilizada uma redução de dimensionalidade através de PCA para permitir a visualização em três dimensões. É possível observar ainda que as classes 'Versicolor' e 'Virginica' apresentam-se mais linearmente separadas na rede à direita, comprovando a eficiência da abordagem proposta.

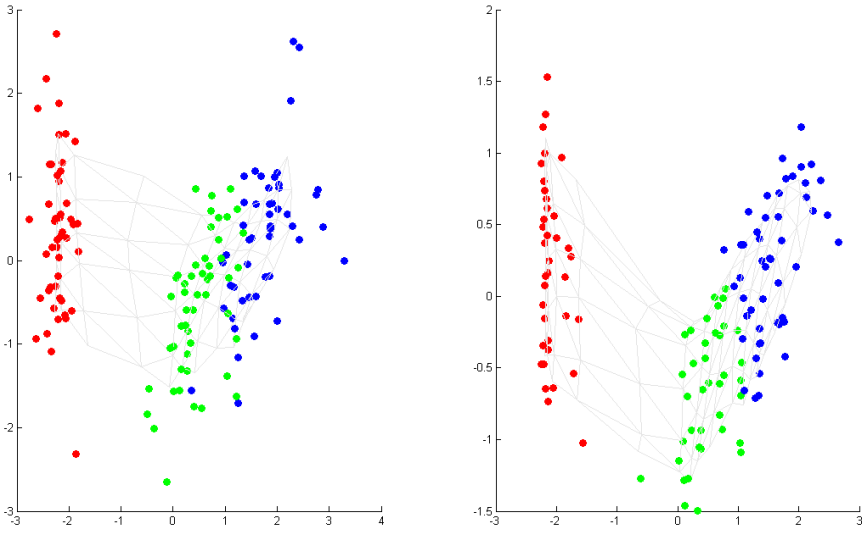


Figura 6.11: Projeção dos dados de entrada sobre a rede SOM treinada a partir da base de dados Iris.

6.3.2.3 Particionamento Vertical com SOM nas Duas Etapas

No terceiro experimento da série, a base de dados Iris foi dividida verticalmente em dois subconjuntos, um contendo os dois primeiros atributos e o outro contendo os dois últimos, sobre os quais foi aplicado o algoritmo SOM conforme a estratégia proposta. Na etapa central, foi aplicado o algoritmo SOM novamente sobre os dados remontados. Os resultados foram comparados com a abordagem tradicional do algoritmo SOM, considerando os 4 atributos simultaneamente.

Nas duas abordagens foram utilizados mapas planos, com neurônios dispostos em formato hexagonal. Foi utilizada inicialização linear dos mapas e treinamento em lote. O

tamanho do mapa e os parâmetros de treinamento estão apresentados na Tabela 6.18 e foram definidos pelo SOM Toolbox a partir de análises estatísticas dos dados.

Tabela 6.18: Parâmetros utilizados no treinamento da base de dados Iris

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	11 x 6	2	1	5	1	1	18
partSOM mapa 1/2	9 x 7	2	1	6	1	1	5
partSOM mapa 2/2	21 x 3	3	1	6	1	1	17
partSOM mapa final	11 x 6	2	1	5	1	1	18

A comparação entre a abordagem tradicional e a arquitetura proposta foi efetuada através dos índices de erro de quantização médio (*qe*) e do erro topológico médio (*te*). Alem disso, foram utilizados os rótulos existentes na base de dados de entrada para verificar a precisão do algoritmo em uma tarefa de classificação. A Tabela 6.19 apresenta os valores médios obtidos para os índices de validação propostos em 20 execuções do algoritmo.

Tabela 6.19: Índices de validação para a base de dados Iris

Algoritmo	qe	te	erros	% acertos
Som tradicional	0.3927	0.0133	6	96.0%
Arquitetura partSOM	0.2081	0.0067	5	96.7%

6.3.2.4 Particionamento Vertical com SOM e *k*-Médias

No quarto e último experimento da série, a base de dados Iris foi dividida verticalmente em dois subconjuntos, como no experimento anterior. Porém, a fim de verificar a flexibilidade da arquitetura partSOM com outros algoritmos, foi utilizada uma combinação de SOM na etapa local e *k*-médias na etapa central. O algoritmo SOM foi aplicado nos dois subconjuntos separadamente, para eleger os representantes dos dados que seriam enviados à unidade central. Em seguida, o algoritmo *k*-médias foi aplicado sobre a base remontada.

O tamanho dos mapas, formato, disposição dos neurônios e parâmetros de treinamento usados na etapa local deste experimento foram exatamente iguais aos utilizados no experimento anterior. Na unidade central, o *k*-médias foi executado com o parâmetro $k = 3$, número de classes existentes do conjunto de entrada.

A precisão da arquitetura proposta, usando a combinação de SOM + *k*-médias foi medida através da comparação com o algoritmo *k*-médias tradicional, em uma tarefa de classificação, como no experimento anterior. A Tabela 6.20 apresenta o valor médio da taxa de acertos do algoritmo para 20 execuções do algoritmo.

Tabela 6.20: Precisão da combinação SOM e *k*-médias para a base de dados Iris

Algoritmo	erros	% acertos
k-médias tradicional	27	82.0%
Arquitetura com SOM + *k*-médias	25	83.3%

6.3.3 Base de Dados Wine

Os experimentos realizados com a base de dados Wine tiveram como objetivo analisar o desempenho da arquitetura partSOM em relação à forma de distribuição das variáveis durante a etapa de particionamento vertical. A base de dados Wine possui 13 atributos. Durante os testes, esses atributos foram agrupados de seis formas distintas, para medir a influência de alguns fatores, tais como correlação entre variáveis e partições com número reduzido de dimensões. Os resultados das várias formas de particionamento – escolhidas aleatoriamente a partir do agrupamento de atributos adjacentes na base de dados original – foram comparados com a abordagem tradicional, com todas as variáveis sendo consideradas simultaneamente.

Inicialmente, a base de dados Wine foi analisada com o algoritmo SOM tradicional. Em seguida, a base de dados foi particionada de diversas formas, conforme apresentado na Tabela 6.21. O particionamento foi realizado sempre agrupando atributos vizinhos, conforme originalmente distribuídos na base.

Tabela 6.21: Formas de particionamento da base de dados Wine

Experimento	Número de Partições	Atributos por Partição					
		1	2	3	4	5	6
1	1	13	-	-	-	-	-
2	2	6	7	-	-	-	-
3	3	4	5	4	-	-	-
4	4	3	4	3	3	-	-
5	4	3	3	4	3	-	-
6	5	2	3	3	3	2	-
7	6	3	2	2	2	2	2

Durante os testes com a abordagem proposta, o algoritmo SOM foi utilizado nas duas etapas, conforme exemplos anteriores. A comparação entre a abordagem tradicional e a arquitetura proposta foi efetuada através dos índices de erro de quantização médio (*qe*) e do erro topológico médio (*te*). Alem disso, foram utilizados os rótulos existentes na base de dados de entrada para verificar a precisão do algoritmo em uma tarefa de classificação. Cada experimento foi repetido 10 vezes e os valores médios obtidos para os índices de validação propostos estão apresentados na Tabela 6.22.

Tabela 6.22: Índices de validação para a base de dados Wine

Experimento	qe	te	erros	% acertos
1	1.8833	0.0169	4	97.75%
2	2.0967	0.0674	7	96.07%
3	2.0280	0.0225	3	98.31%
4	2.0523	0.0281	5	96.63%
5	2.0292	0.0225	4	97.75%
6	2.0032	0.0337	3	98.31%
7	1.9929	0.0506	4	97.75%

6.3.4 Base de Dados Mushroom

Os experimentos realizados com a base de dados Mushroom tiveram como objetivo estimar o desempenho da arquitetura partSOM em relação à redução de dados transferidos entre as unidades remotas e a unidade central, que é uma das principais vantagens da abordagem proposta em relação às abordagens tradicionais, onde os dados são centralizados antes do processo de análise de agrupamentos.

A base de dados Mushroom possui um conjunto de 8124 instâncias, cada uma com 22 atributos. Como foi necessária a conversão dos dados categóricos em dados numéricos durante a etapa de pré-processamento, a dimensionalidade da base de dados a ser analisada aumentou para 117 atributos (ver seção 6.1.4). Foram feitas algumas considerações durante a realização dos testes. Primeiro, foi considerado que a base de dados Mushroom encontrava-se distribuída entre várias unidades remotas. Segundo, considerou-se 1 byte para cada atributo armazenado, uma vez que após a conversão, todos os atributos passaram a ser binários.

Conforme realizado nos experimentos anteriores, inicialmente a base de dados foi analisada com o algoritmo SOM tradicional, considerando-se todos os 117 atributos simultaneamente. Em todas as comparações, foi utilizado um mapa plano de tamanho 23 x 19, totalizando 437 neurônios, dispostos em formato hexagonal. Os parâmetros de treinamento utilizados nesta configuração foram $\sigma_{inicial} = 3$, $\sigma_{final} = 1$ e épocas = 1 na fase de aproximação e $\sigma_{inicial} = \sigma_{final} = 1$ e épocas = 3 na fase de ajuste fino, conforme descrito nas Tabelas 6.24 a 6.26.

Em seguida, a base de dados foi verticalmente particionada de três maneiras diferentes, considerando-se 2, 3 e 4 partições. Durante a distribuição dos dados nos mapas, buscou-se manter agrupados aquelas colunas relacionadas ao mesmo atributo na base de dados original, de forma a manter juntos atributos relacionados com as mesmas características da espécie, como por exemplo, o formato, cor e aspecto do caule. A forma de distribuição dos atributos nas unidades remotas está descrita na Tabela 6.23. Da mesma forma que no SOM tradicional, em todos os experimentos foram utilizados mapas planos, no formato hexagonal. Detalhes adicionais sobre o tamanho dos mapas e parâmetros de treinamento dos algoritmos estão apresentados nas Tabelas 6.24 a 6.26.

Tabela 6.23: Formas de particionamento da base de dados Mushroom

Experimento	Número de Partições	Atributos por Partição			
		1	2	3	4
1	1	117	-	-	-
2	2	49	68	-	-
3	3	49	33	35	-
4	4	31	18	33	35

Tabela 6.24: Parâmetros utilizados no treinamento da base de dados Mushroom com 2 partições

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	23 x 19	3	1	1	1	1	3
partSOM mapa 1/2	25 x 18	4	1	1	4	1	2
partSOM mapa 2/2	23 x 19	3	1	1	3	1	2
partSOM mapa final	23 x 19	1	1	1	1	1	2

Tabela 6.25: Parâmetros utilizados no treinamento da base de dados Mushroom com 3 partições

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	23 x 19	3	1	1	1	1	3
partSOM mapa 1/3	25 x 18	2	1	1	2	1	1
partSOM mapa 2/3	25 x 18	2	1	1	2	1	1
partSOM mapa 3/3	23 x 19	2	1	1	2	1	1
partSOM mapa final	25 x 18	1	1	1	1	1	3

Tabela 6.26: Parâmetros utilizados no treinamento da base de dados Mushroom com 4 partições

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	23 x 19	3	1	1	1	1	3
partSOM mapa 1/4	25 x 18	4	1	1	4	1	1
partSOM mapa 2/4	23 x 19	4	1	3	4	1	3
partSOM mapa 3/4	25 x 18	4	1	1	4	1	1
partSOM mapa 4/4	23 x 19	4	1	3	4	1	3
partSOM mapa final	23 x 19	4	1	3	4	1	3

A comparação dos resultados obtidos entre a abordagem tradicional e a arquitetura proposta foi efetuada através dos índices de erro de quantização médio (*qe*), do erro topológico médio (*te*) e de Davies-Bouldin (*db*). Alem disso, foram utilizados os rótulos existentes na base de dados de entrada para verificar a precisão do algoritmo em uma tarefa de classificação. Cada experimento foi repetido 20 vezes e os valores médios e desvio padrão (σ) obtidos para os índices de validação propostos estão apresentados na

Tabela 6.27. Os índices *qe* e *te* são calculados sobre um SOM inicializado de forma linear, com os mesmos atributos, por isso não apresentam variação. O índice *db* apresenta variação entre uma execução e outra por utilizar os rótulos obtidos pelo *k*-médias, que é um algoritmo não-determinístico.

Tabela 6.27: Índices de validação para a base de dados Mushroom

Experimento	qe		te		db		erros	% acertos
	médio	σ	médio	σ	médio	σ		
1	1.8339	0.00%	0.0720	0.00%	1.1437	4.49%	295	96.37%
2	1.8962	0.00%	0.0529	0.00%	1.0872	3.99%	292	96.41%
3	2.6561	0.00%	0.1350	0.00%	1.1543	1.46%	331	95.93%
4	1.8695	0.00%	0.0512	0.00%	1.1130	5.76%	292	96.41%

Para avaliar a redução de dados transferidos entre as unidades remotas e a unidade central, foram calculados os tamanhos de cada *codebook* e dos vetores de índices, considerando-se 1 byte para cada atributo armazenado. Os resultados estão apresentados nas Tabelas 6.28 a 6.30. Em todos os casos, o volume de dados transferidos entre as unidades remotas e a unidade central obteve uma redução bastante significativa.

Tabela 6.28: Volume de dados transferido (em bytes) entre as unidades remotas e a unidade central para a base de dados Mushroom com 2 partições

Algoritmo	Unidade 1	Unidade 2	Total
Som Tradicional	398076	552432	950508
Arquitetura partSOM	30174	37840	68014
Redução (em bytes)	367902	514592	882494
Redução (em %)	92.42%	93.15%	92.84%

Tabela 6.29: Volume de dados transferido (em bytes) entre as unidades remotas e a unidade central para a base de dados Mushroom com 3 partições

Algoritmo	Unidade 1	Unidade 2	Unidade 3	Total
Som Tradicional	398076	268092	284340	950508
Arquitetura partSOM	30174	22974	23419	76567
Redução (em bytes)	367902	245118	260921	873941
Redução (em %)	92.42%	91.43%	91.76%	91.94%

Tabela 6.30: Volume de dados transferido (em bytes) entre as unidades remotas e a unidade central para a base de dados Mushroom com 4 partições

Algoritmo	Unidade 1	Unidade 2	Unidade 3	Unidade 4	Total
Som Tradicional	251844	146232	268092	284340	950508
Arquitetura partSOM	22074	15990	22974	23419	84457
Redução (em bytes)	229770	130242	245118	260921	866051
Redução (em %)	91.24%	89.07%	91.43%	91.76%	91.11%

6.3.5 Base de Dados Wages

Os experimentos realizados com a base de dados Wages tiveram como objetivo verificar o desempenho da arquitetura partSOM na análise de dados reais de uma população. Neste experimento, buscou-se segmentar a base de dados em agrupamentos distintos, de acordo com o perfil dos indivíduos.

Em geral, bases de dados reais são mais difíceis de segmentar que bases de dados sintéticas, uma vez que nem sempre essas bases possuem agrupamentos bem definidos. Além disso, o número de agrupamentos detectados normalmente é grande e exige uma etapa bem detalhada de análise estatística (segmentação *a posteriori*) para identificação de características comuns aos indivíduos que compõem cada um dos agrupamentos detectados durante a etapa de pós-processamento. Outra dificuldade das bases de dados reais é a inexistência de rótulos pré-definidos que possam ser utilizados para verificação dos resultados, que é bastante comum em bases de dados sintéticas.

Originalmente, a base de dados Wages possui um conjunto de 534 indivíduos, cada um com 11 atributos. Como foi necessária a conversão dos dados categóricos em dados numéricos durante a etapa de pré-processamento, a dimensionalidade da base de dados aumentou para 20 atributos.

Inicialmente, a base de dados foi analisada com o algoritmo SOM tradicional, considerando os 20 atributos simultaneamente. Foi utilizado um mapa plano de tamanho 12 x 10, totalizando 120 neurônios, dispostos em formato hexagonal. Na etapa seguinte, a arquitetura partSOM foi aplicada sobre duas partições verticais dos dados e, posteriormente, sobre os dados recombinados. Os parâmetros de configuração e treinamento utilizados em ambas as etapas estão apresentados na Tabela 6.31.

Tabela 6.31: Parâmetros utilizados no treinamento da base de dados Wages

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	12 x 10	2	1	3	1	1	9
partSOM mapa 1/2	12 x 10	2	1	3	1	1	9
partSOM mapa 2/2	13 x 9	2	1	3	1	1	9
partSOM mapa final	12 x 10	2	1	3	1	1	9

Durante o processo de análise da base Wages, o algoritmo SOM tradicional foi aplicado sobre o conjunto de dados e, em seguida, o algoritmo *k*-médias foi aplicado sobre o mapa treinado para obter a segmentação. Essa estratégia executa o algoritmo *k*-médias com diferentes valores de *k*, sobre o mesmo conjunto de dados (o mapa treinado) e identifica a melhor partição através do uso do índice de Davies-Bouldin [Vesanto, 2000]. Essa abordagem identificou a existência de 10 agrupamentos de indivíduos com características semelhantes na base de dados Wages. A Figura 6.12 apresenta a matriz-U e o mapa treinado e segmentado com a ajuda do algoritmo *k*-médias.

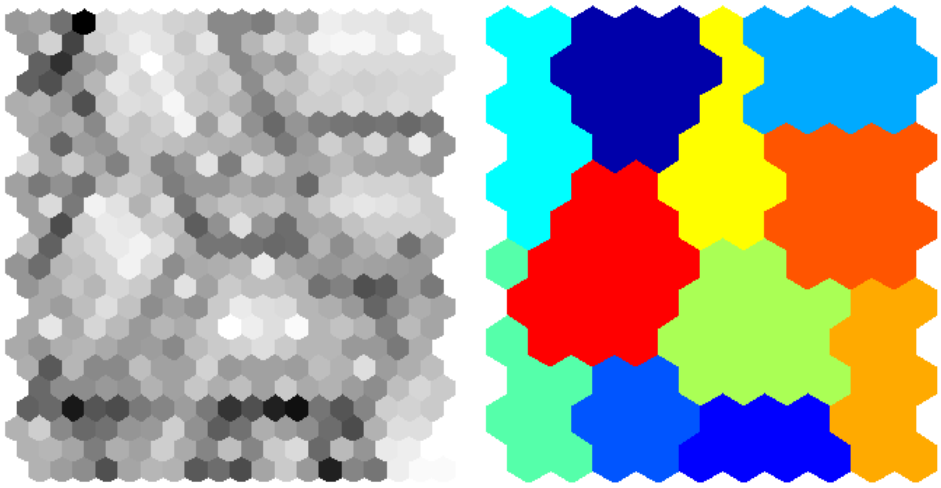


Figura 6.12: Matriz-U e mapa segmentado com o algoritmo SOM tradicional usando a base de dados Wages.

A Tabela 6.32 apresenta o perfil dos indivíduos que compõem cada um dos agrupamentos detectados. São mostrados os valores médios para os atributos numéricos e percentuais para os atributos categóricos, destacando-se os valores mais significativos.

Tabela 6.32: Análise estatística dos agrupamentos detectados na base de dados Wages

Agrupamentos	Gr1	Gr2	Gr3	Gr4	Gr5	Gr6	Gr7	Gr8	Gr9	Gr10
Educação (anos)	12.12	14.71	11.60	9.42	15.72	11.42	13.16	10.94	13.38	12.83
Sulistas (%)	21%	18%	28%	54%	22%	42%	47%	50%	11%	35%
Mulheres (%)	8%	37%	65%	42%	54%	8%	49%	22%	38%	67%
Experiência (anos)	11.96	18.41	22.62	39.13	15.05	22.67	13.55	16.17	14.36	17.49
Sindicalizado (%)	27%	4%	22%	33%	25%	33%	8%	44%	13%	8%
Salário (US$)	8.41	13.09	6.59	8.46	11.58	9.22	7.28	6.31	10.54	7.51
Idade (anos)	30.08	39.12	40.22	54.54	36.72	40.08	32.71	33.11	33.74	36.32
Raça - Outros (%)	0%	10%	20%	0%	1%	13%	55%	89%	0%	0%
Raça - Hispânica (%)	0%	0%	1%	8%	0%	0%	45%	11%	0%	0%
Raça - Branco (%)	100%	90%	78%	92%	99%	88%	0%	0%	100%	100%
Profissão – Administrativo (%)	0%	100%	0%	0%	0%	0%	8%	0%	0%	0%
Profissão – Vendas (%)	0%	0%	0%	0%	0%	0%	8%	0%	3%	32%
Profissão – Religioso (%)	0%	0%	0%	8%	0%	8%	41%	0%	8%	68%
Profissão – Serviços (%)	0%	0%	96%	0%	0%	0%	20%	0%	3%	0%
Profissão – Profiss. liberal (%)	0%	0%	3%	0%	100%	8%	16%	6%	18%	0%
Profissão – Outras profissões (%)	100%	0%	1%	92%	0%	83%	6%	94%	67%	0%
Setor – Outros setores (%)	100%	88%	100%	33%	100%	0%	100%	22%	0%	98%
Setor – Indústria / produção (%)	0%	12%	0%	67%	0%	0%	0%	78%	100%	2%
Setor – Construção civil (%)	0%	0%	0%	0%	0%	100%	0%	0%	0%	0%
Casado (%)	58%	69%	64%	83%	73%	75%	51%	61%	66%	65%

A Tabela 6.33 resume as principais características de cada agrupamento detectado.

Tabela 6.33: Principais características dos agrupamentos da base de dados Wages

Agrupamento	Amostra	Descrição das Principais Características
Gr1	52	Não sulistas; homens; brancos; prof: outros; setor: outros
Gr2	51	Não sulistas; brancos; setor administrativo; salário alto; não sindic
Gr3	74	Mais experiência; casados; profissão: serviços; setor: outros
Gr4	24	Sulistas; mais velhos; casados; profissão: outros
Gr5	81	Brancos; mais educação; casados; prof. liberais; setor: outros
Gr6	24	Mais experiência; casados; setor: construção civil
Gr7	49	Hispânicos; pouca experiência; setor: outros; salário baixo; não sindic
Gr8	18	Homens, raça: outros; sindicalizados; salário baixo; profissão: outros
Gr9	61	Não sulistas; brancos, casados; setor: indústria/produção
Gr10	100	Brancos, mulheres; casados; prof: religiosos e vendas; setor: outros

Uma análise estatística mais detalhada dos indivíduos que compõem os agrupamentos detectados permite verificar que a maioria dos grupos é composta por indivíduos com apenas um ou dois atributos em comum, conforme pode ser verificado na Tabela 6.33. Esse fato acontece em função de se tratar de uma base de dados real, obtida a partir de uma amostra aleatória de uma população, onde a dispersão dos indivíduos no espaço *p*-dimensional não segue nenhuma distribuição estatística pré-definida e os grupos de indivíduos realmente semelhantes possuem tamanho bem reduzido.

A forte influência de alguns atributos, particularmente os categóricos, é determinante no resultado obtido. Atributos como ‘raça’, ‘ocupação’ e ‘setor’ são decisivos na identificação de cada um dos agrupamentos detectados e praticamente dominam o processo de segmentação, o que dificulta a análise desse conjunto de dados em relação aos atributos menos favorecidos.

Ao contrário da abordagem tradicional, a arquitetura partSOM considera que a segmentação obtida a partir de um conjunto menor de atributos que sejam, de certa forma, relacionados entre si, aumenta as chances de se obter agrupamentos mais homogêneos e com maior número de indivíduos em cada grupo. Assim, as diferentes visões sob diferentes aspectos que a arquitetura proporciona, ajudam a entender melhor como os dados estão distribuídos no espaço *p*-dimensional.

Para comprovar as vantagens da abordagem proposta, a base de dados Wages foi dividida verticalmente em dois subconjuntos, respectivamente denominados de Wages1 e Wages2. Nessa divisão, tentou-se manter juntos atributos que estivessem relacionados entre si, ao mesmo tempo em que se buscou garantir certo equilíbrio no número de atributos de cada subconjunto. Assim, o primeiro mapa analisa os aspectos relacionados à vida profissional do indivíduo (anos de estudo; experiência profissional; vinculação a associações/sindicatos; salário; ocupação e setor onde trabalha) e o segundo mapa analisa

os aspectos relacionados à vida pessoal do indivíduo (região onde reside; sexo; idade; raça e estado civil).

A Figura 6.13 apresenta os mapas parciais obtidos com a aplicação da arquitetura partSOM sobre os conjuntos de dados Wages1 e Wages2. Uma análise mais detalhada dos mapas parciais ajuda a compreender o resultado obtido. Por exemplo, é possível verificar que o mapa à esquerda, que correspondente aos aspectos profissionais dos indivíduos, possui 10 agrupamentos de indivíduos, conforme detectado pela análise simultânea de todas as variáveis; enquanto que o mapa à direita, correspondente aos aspectos pessoais dos indivíduos, apresenta apenas 3 agrupamentos de indivíduos. Essa diferença entre os dois mapas demonstra que os atributos pertencentes ao aspecto pessoal possuem menos dispersão que os pertencentes ao aspecto profissional.

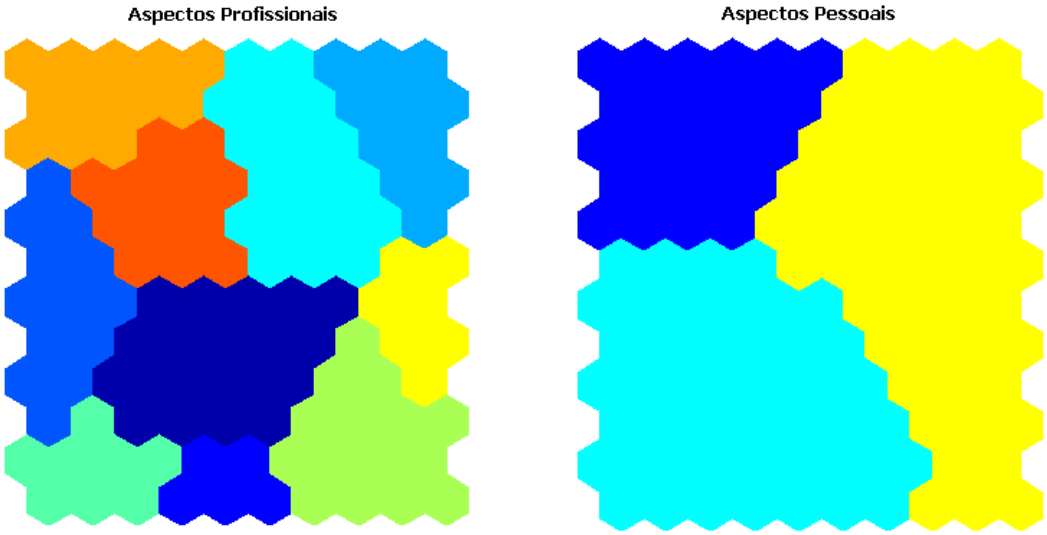


Figura 6.13: Mapas parciais obtidos com a arquitetura partSOM usando a base de dados Wages.

Uma análise estatística mais detalhada dos agrupamentos detectados pelo primeiro conjunto de atributos (aspectos profissionais) é apresentada nas Tabelas 6.34 e 6.35. É possível verificar que o atributo 'ocupação' é o mais determinante na separação dos grupos do primeiro conjunto de atributos. Por serem resultados de análises independentes, é importante destacar que os números de identificação dos agrupamentos nas Tabelas 6.32 e 6.33 não correspondem necessariamente aos mesmos nas Tabelas 6.34 e 6.35.

A mesma análise é feita para o segundo conjunto de atributos (aspectos pessoais) e os resultados são apresentados nas Tabelas 6.36 e 6.37. É possível verificar que o atributo 'raça' e 'sexo' são os mais determinantes na separação dos grupos do segundo conjunto de atributos.

A comparação dos resultados obtidos entre a abordagem tradicional e a arquitetura proposta foi efetuada através dos índices de erro de quantização médio (*qe*), do erro topológico médio (*te*) e de Davies-Bouldin (*db*), conforme Tabela 6.38.

Tabela 6.34: Análise estatística dos agrupamentos detectados na base de dados Wages1

Agrupamentos	Gr1	Gr2	Gr3	Gr4	Gr5	Gr6	Gr7	Gr8	Gr9	Gr10
Educação (anos)	12.03	11.42	11.51	15.67	14.51	8.16	12.21	15.95	12.92	13.21
Experiência (anos)	13.74	22.67	20.75	15.21	18.16	40.11	15.81	15.42	16.19	18.45
Sindicalizado (%)	28%	33%	19%	22%	4%	37%	24%	11%	9%	3%
Salário (US$)	8.86	9.22	6.30	11.47	12.60	6.23	8.65	15.18	7.23	7.59
Profissão – Administrativo (%)	0%	0%	0%	0%	100%	0%	0%	32%	0%	0%
Profissão – Vendas (%)	0%	0%	0%	0%	0%	0%	0%	0%	0%	100%
Profissão – Religioso (%)	0%	8%	0%	0%	0%	0%	11%	0%	100%	0%
Profissão – Serviços (%)	2%	0%	100%	0%	0%	0%	2%	5%	0%	0%
Profissão – Profiss. liberal (%)	0%	8%	0%	100%	0%	5%	0%	63%	0%	0%
Profissão – Outras profissões (%)	98%	83%	0%	0%	0%	95%	87%	0%	0%	0%
Setor – Outros setores (%)	100%	0%	100%	100%	100%	26%	0%	0%	100%	89%
Setor – Indústria / produção (%)	0%	0%	0%	0%	0%	74%	100%	100%	0%	11%
Setor – Construção civil (%)	0%	100%	0%	0%	0%	0%	0%	0%	0%	0%

Tabela 6.35: Principais características dos agrupamentos da base de dados Wages1

Agrupamento	Amostra	Descrição das Principais Características
Gr1	65	Profissão: outros; setor: outros; pouco experiência; salário baixo
Gr2	24	Setor: construção civil; pouca formação; sindicalizados
Gr3	80	Profissão: serviços; setor: outros; pouca formação; salário baixo
Gr4	90	Prof. liberais; setor: outros; boa formação
Gr5	49	Profissão: administradores; setor: outros; boa formação; não sindic
Gr6	19	Profissão: outras; pouca formação; experiência; salário baixo; sindic
Gr7	62	Setor: indústria; profissão: outros; sindicalizados
Gr8	19	Setor: indústria; alta remuneração; boa formação
Gr9	88	Profissão: religiosos; setor: outros
Gr10	38	Profissão: vendas; setor: outros

Tabela 6.36: Análise estatística dos agrupamentos detectados na base de dados Wages2

Agrupamentos	Gr1	Gr2	Gr3
Sulistas (%)	29%	40%	24%
Mulheres (%)	0%	45%	100%
Idade (anos)	36.36	36.86	37.37
Raça - Outros (%)	0%	73%	0%
Raça - Hispânica (%)	1%	27%	0%
Raça - Branco (%)	99%	0%	100%
Casado (%)	69%	60%	64%

Tabela 6.37: Principais características dos agrupamentos da base de dados Wages2

Agrupamento	Amostra	Descrição das Principais Características
1	238	Homens; brancos; casados; maioria não sulista
2	92	Sulistas; raça: hispânicos e outros
3	204	Mulheres; brancas; casadas; maioria não sulista

Tabela 6.38: Índices de validação para a base de dados Wages

Experimento	qe		te		db	
	médio	σ	médio	σ	médio	σ
Som tradicional	2.4690	0.00%	0.0431	0.00%	1.2459	1.57%
partSOM mapa 1/2	1.2854	0.00%	0.0262	0.00%	1.0920	1.76%
partSOM mapa 2/2	0.1935	0.00%	0.0281	0.00%	0.9698	3.33%
partSOM mapa final	1.6342	0.00%	0.0112	0.00%	0.9928	4.66%

6.3.6 Base de Dados Biscoitos

Assim como no experimento anterior, os testes realizados com a base de dados Biscoito tiveram como objetivo analisar o desempenho da arquitetura partSOM em uma base de dados comercial, com dados reais. A base de dados Biscoito possui 17 atributos, que foram divididos em 2 grupos, conforme apresentado anteriormente nas Tabela 6.6 e 6.7. Na divisão dos atributos, buscou-se manter agrupados aqueles relacionados ao mesmo contexto, de forma a possibilitar uma interpretação dos resultados parciais. Assim, o primeiro conjunto de dados analisou a opinião dos entrevistados, enquanto que no segundo conjunto foram reunidas as informações sobre seu perfil sócio-econômico.

Além disso, conforme descrito em [Marques, 2008], uma avaliação estatística dos dados permite identificar que alguns atributos aparecem sempre com pontuação alta, independente do entrevistado. Tais atributos, embora não sejam úteis na identificação dos agrupamentos, são importantes por estarem relacionados às características que são sempre desejáveis e que qualquer produto deve possuir, independente do público ao qual se destina.

A Figura 6.14 mostra os mapas parciais obtidos com a aplicação da arquitetura partSOM sobre o conjunto de dados Biscoitos. O mapa à esquerda correspondente aos hábitos de consumo dos indivíduos (questões 1 a 13), enquanto que o mapa à direita, correspondente ao perfil sócio-econômico dos indivíduos. Como no exemplo anterior, a diferença entre os dois mapas demonstra que os atributos pertencentes ao aspecto sócio-econômico possuem menos dispersão que os pertencentes ao aspecto profissional.

A Tabela 6.39 apresenta os valores médios de resposta dos indivíduos que compõem cada um dos agrupamentos detectados, para as 13 primeiras questões da pesquisa (hábitos de consumo). A Tabela 6.40 apresenta os valores médios de resposta para as 4 últimas questões da pesquisa (perfil sócio-econômico). A Tabela 6.41 resume as principais características de cada agrupamento detectado.

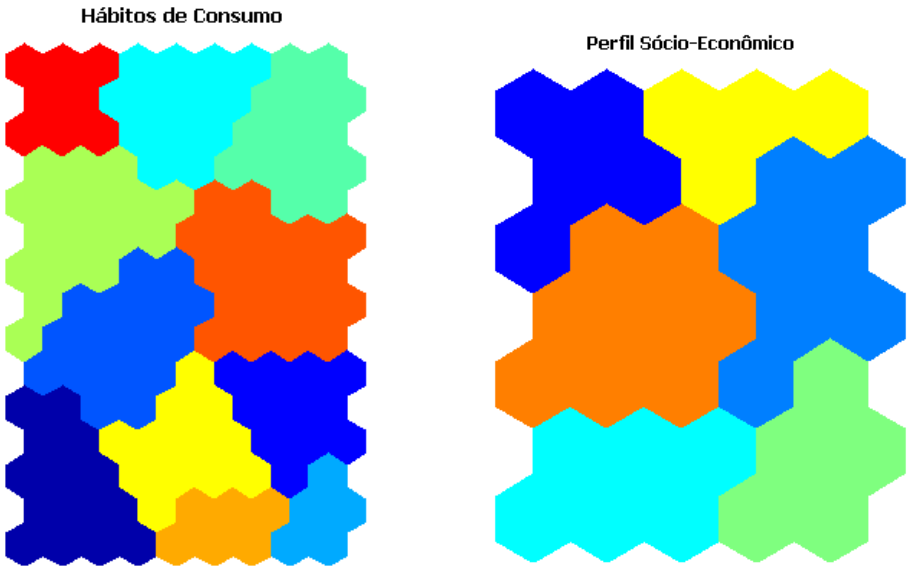


Figura 6.14: Mapas parciais obtidos com a arquitetura partSOM usando a base de dados Biscoitos.

Tabela 6.39: Análise estatística dos agrupamentos detectados na base de dados Biscoitos1

Agrupamentos	Gr1	Gr2	Gr3	Gr4	Gr5	Gr6	Gr7	Gr8	Gr9	Gr10	Gr11
Questão a1	4.61	3.69	3.30	3.65	3.26	3.24	4.38	4.26	2.15	2.42	3.47
Questão a2	4.56	4.17	4.32	4.39	3.70	1.85	3.62	4.08	4.35	3.65	2.80
Questão a3	4.48	2.13	2.02	4.58	1.44	1.37	1.73	1.74	2.15	1.98	1.17
Questão a4	4.25	3.15	1.68	1.52	2.19	1.32	2.00	1.47	2.05	3.50	1.63
Questão a5	4.54	4.63	4.82	4.61	2.60	4.24	4.20	4.53	4.10	4.45	3.00
Questão a6	4.55	4.77	3.60	4.84	3.35	3.86	3.98	4.55	4.65	4.27	2.80
Questão a7	4.25	3.46	3.76	1.77	2.77	2.24	3.62	4.58	3.00	2.21	2.33
Questão a8	4.30	2.58	2.02	3.90	1.91	2.86	3.03	3.79	4.05	3.35	1.93
Questão a9	4.48	4.73	3.48	4.65	4.02	3.90	3.67	4.61	4.45	4.15	2.30
Questão a10	4.45	4.21	3.54	4.58	3.07	3.10	2.57	3.68	4.80	2.89	2.50
Questão a11	4.48	4.71	4.76	4.65	2.49	4.02	3.58	4.74	4.50	3.73	3.10
Questão a12	4.47	2.08	3.80	2.58	2.47	2.12	3.27	3.50	4.85	2.41	2.60
Questão a13	4.63	4.92	4.78	4.84	4.53	4.63	3.77	4.74	4.95	4.47	2.77

Tabela 6.40: Análise estatística dos agrupamentos detectados na base de dados Biscoitos2

Agrupamentos	Gr1	Gr2	Gr3	Gr4	Gr5	Gr6
Idade	3.20	1.42	3.71	2.07	3.73	3.95
Classe social	1.79	1.86	2.54	1.88	1.00	2.36
Freqüência de Consumo	1.02	1.00	2.03	2.04	1.11	1.00
Perfil	1.00	2.57	1.64	2.72	2.49	2.56

Tabela 6.41: Principais características dos agrupamentos da base de dados Biscoitos2

Agrupamento	Amostra	Descrição das Principais Características
1	70	Mães com filhos; adultas; classes A/B e C; consumo alto
2	105	Jovens; solteiros; classes A/B e C; consumo alto
3	78	Mulheres; adultas ou idosas; classes C e D, consumo médio
4	97	Homens e mulheres; jovens e adultos; classes A/B e C; consumo médio
5	75	Adultos e idosos; sem filhos pequenos; classe A/B; consumo alto
6	109	Adultos e idosos; sem filhos pequenos; classes C/D; consumo alto

Os parâmetros de treinamento utilizados durante as duas etapas de análise da base de dados Biscoito estão apresentados na Tabela 6.42. Os índices de validação obtidos em ambas as etapas dos testes estão apresentados na Tabela 6.43

Tabela 6.42: Parâmetros utilizados no treinamento da base de dados Biscoito

Algoritmo	Tamanho do Mapa	Fase de Aproximação			Fase de Ajuste Fino		
		$\sigma_{inicial}$	σ_{final}	épocas	$\sigma_{inicial}$	σ_{final}	épocas
Som tradicional	15 x 9	2	1	2	1	1	8
partSOM mapa 1/2	15 x 9	2	1	2	1	1	8
partSOM mapa 2/2	7 x 5	1	1	1	1	1	2
partSOM mapa final	15 x 8	2	1	3	1	1	9

Tabela 6.43: Índices de validação para a base de dados Biscoitos

Experimento	qe		te		db	
	médio	σ	médio	σ	médio	σ
Som tradicional	2.9099	0.00%	0.0587	0.00%	1.2109	2.53%
partSOM mapa 1/2	2.3042	0.00%	0.0600	0.00%	1.1725	3.84%
partSOM mapa 2/2	0.9373	0.00%	0.0467	0.00%	1.0090	1.94%
partSOM mapa final	1.5140	0.00%	0.0374	0.00%	1.0951	2.96%

6.3.7 Base de Dados CAPES

Os experimentos realizados com a base de dados CAPES tiveram como objetivo demonstrar a importância das análises parciais proporcionadas pela arquitetura partSOM em relação à análise simultânea de todos os atributos. O experimento demonstra que a utilização de mapas parciais, contendo um determinado subconjunto de atributos relacionados entre si, permite uma interpretação semântica mais rica do conjunto de dados analisado.

A base de dados CAPES possui informações sobre a avaliação de 39 cursos de pós-graduação da área de Ciência de Computação realizada pela CAPES, no período de 2004 a 2006 e publicadas no relatório trienal 2007. Cada avaliação é composta por um conjunto de 21 atributos, divididos em 5 grupos: proposta do programa; corpo docente; corpo discente, teses e dissertações; produção intelectual e inserção social.

Durante os testes realizados, a base de dados foi particionada verticalmente, mantendo juntos os atributos inter-relacionados na avaliação original, de forma a dar um conteúdo semântico à interpretação dos mapas. Essa abordagem permitiu comparar as instituições a partir dos mesmos quesitos avaliados pela CAPES e foi possível detectar pontos fortes e pontos fracos das instituições a partir desses critérios. Os resultados foram comparados com a abordagem tradicional, com todas as variáveis consideradas simultaneamente, de forma a apresentar as vantagens de utilização da arquitetura proposta.

A Figura 6.15 apresenta os resultados do processo de análise de agrupamentos utilizando o algoritmo SOM tradicional, utilizando-se simultaneamente todos os atributos. A matriz-U obtida é mostrada à esquerda e o mapa devidamente segmentado com o algoritmo *k*-médias é mostrado à direita. A segmentação do mapa demonstra que o algoritmo SOM consegue interpretar os resultados atribuídos pelos avaliadores, separando adequadamente as instituições de acordo com os conceitos recebidos. Os valores entre parênteses correspondem às notas finais atribuídas pelo CTC a cada instituição. Esses valores, assim como a nota final atribuída pela CA, não foram utilizados durante a análise dos dados pelo algoritmo.

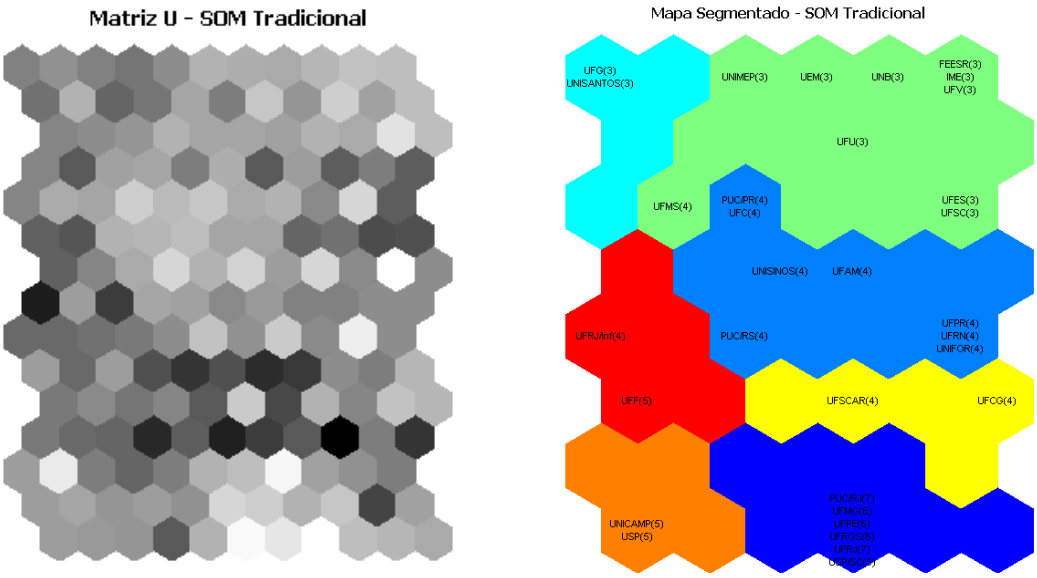


Figura 6.15: Matriz-U e mapa segmentado com o algoritmo SOM tradicional usando a base de dados CAPES.

Apesar de apresentar bons resultados, a abordagem tradicional não permite avaliar os resultados obtidos em cada um dos aspectos analisados. Por exemplo, se a UFF, a UNICAMP, a USP e a USP/SC apresentam o mesmo conceito 5, por que não pertencem ao mesmo agrupamento? O que diferencia o programa de cada uma dessas instituições? Quais seus pontos fortes e pontos fracos? São essas questões que podem ser melhor analisadas com a avaliação parcial proposta pela arquitetura partSOM.

A Figura 6.16 apresenta os resultados da análise da arquitetura partSOM sobre os atributos correspondentes ao quesito ‘Proposta do Programa’.

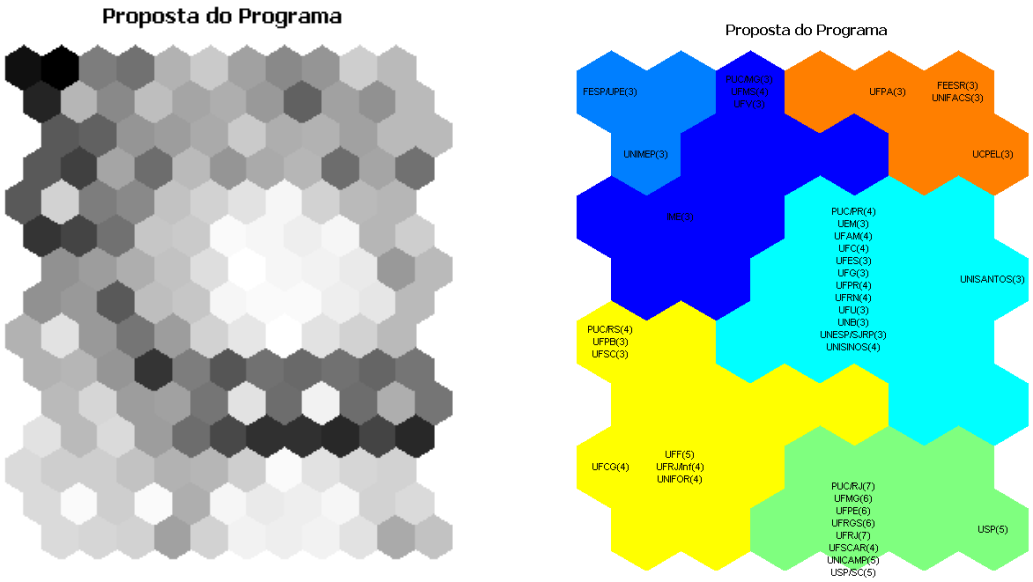


Figura 6.16: Mapas parciais do aspecto Proposta do Programa da base de dados CAPES.

É importante observar que, apesar desse aspecto ter peso 0.0% na atribuição da nota final, há uma clara relação entre as instituições que possuem as melhores propostas e as que foram mais bem avaliadas. Além disso, ele permite identificar como estão distribuídas as linhas de pesquisa da instituição, além da estrutura curricular e da infra-estrutura fornecida pelo programa.

Também é possível observar algumas particularidades do tipo: a UFSCAR, apesar de possuir conceito 4, possui excelente proposta, equivalente às das melhores instituições do país, uma vez que obteve conceito MB em todos os itens. Já a UFMS, apesar de também ter conceito 4, possui desempenho inferior nesse item.

Outro ponto a destacar a favor da arquitetura partSOM é a possibilidade de analisar bases de dados parcialmente incompletas. Conforme pode ser verificado na Tabela 6.9, algumas instituições receberam conceito NA (Não Aplicável) em aspectos como ‘Corpo Discente’, ‘Produção Intelectual’ e ‘Inserção Social’, por terem programas recentes e ainda sem alunos concluintes. Essas instituições foram eliminadas da base de dados durante o processo de análise com o algoritmo SOM tradicional, mas puderam ser consideradas pela arquitetura partSOM durante a avaliação dos demais aspectos. As instituições FESP/UPE, PUC/MG, UCPEL, UFPA, UFPB, UNESP/SJRP e UNIFACS incluem-se nessa categoria e estão representadas na Figura 6.16.

A Figura 6.17 apresenta os resultados da análise da arquitetura partSOM sobre os atributos correspondentes ao quesito ‘Corpo Docente’. Nesse caso, é possível identificar instituições tradicionais que, mesmo recebendo uma boa avaliação por parte da CAPES, foram questionadas no próprio relatório produzido pela CA em relação às deficiências

neste aspecto. Além disso, o resultado parcial ajuda a entender porque instituições como a UFF e a USP, apesar de terem recebido conceito 5 pela comissão de avaliação, os conceitos obtidos no aspecto 'Corpo Docente' as distanciam de outras instituições com nota final equivalente, como a UNICAMP e da USP/SC.

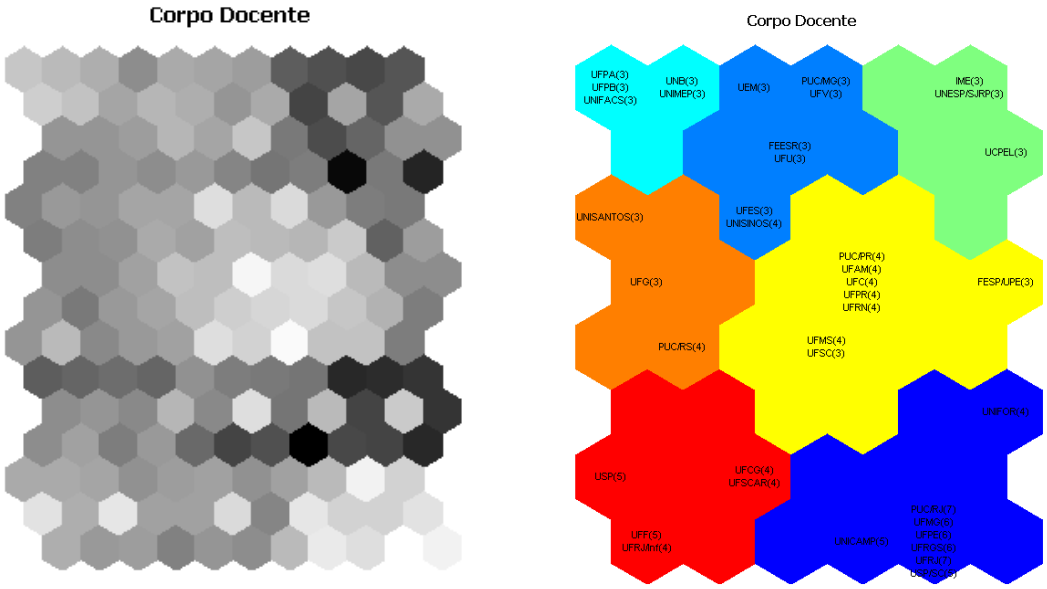


Figura 6.17: Mapas parciais do aspecto Corpo Docente da base de dados CAPES.

Instituições como a UFSC, a UFCG, a UFSCAR, a UFRJ/Inf e a UNICAMP obtiveram destaque neste aspecto, obtendo uma avaliação superior a outras instituições de mesma categoria. É interessante observar que destas, a UFSC, a UFCG, a UFRJ/Inf e a UNICAMP tiveram nota final reduzida pelo CTC em relação à nota publicada pela CA.

A Figura 6.18 apresenta os resultados da análise da arquitetura partSOM sobre os atributos correspondentes ao quesito 'Corpo Discente, Teses e Dissertações'. Nesse aspecto, é possível verificar que as instituições estão representadas de forma mais dispersa ao longo do mapa, o que caracteriza pouca similaridade entre elas e a existência de agrupamentos menos homogêneos.

Há vários casos de instituições com desempenho inferior a outras instituições de mesma categoria (UFRJ/Inf, USP, UFF, PUC/PR, UFC, UFMS) e outros casos de instituições com desempenho superior à categoria (UNB, UNIFOR, UFSC, UFCG, UFSCAR). A UFRJ/Inf, USP, PUC/PR e UFMS foram criticadas neste aspecto em seus respectivos relatórios de avaliação.

A Figura 6.19 apresenta os resultados da análise da arquitetura partSOM sobre os atributos correspondentes ao quesito 'Produção Intelectual'. Nesse aspecto, ao contrário do anterior, as instituições estão melhor agrupadas, embora exista certo grau de heterogeneidade entre os elementos dentro dos três agrupamentos que foram detectados.

Em parte, isso é provocado pelo fato de o primeiro atributo desse aspecto ter peso igual a 0.75, sendo bastante superior aos demais, conforme pode ser visto na Tabela 6.8.

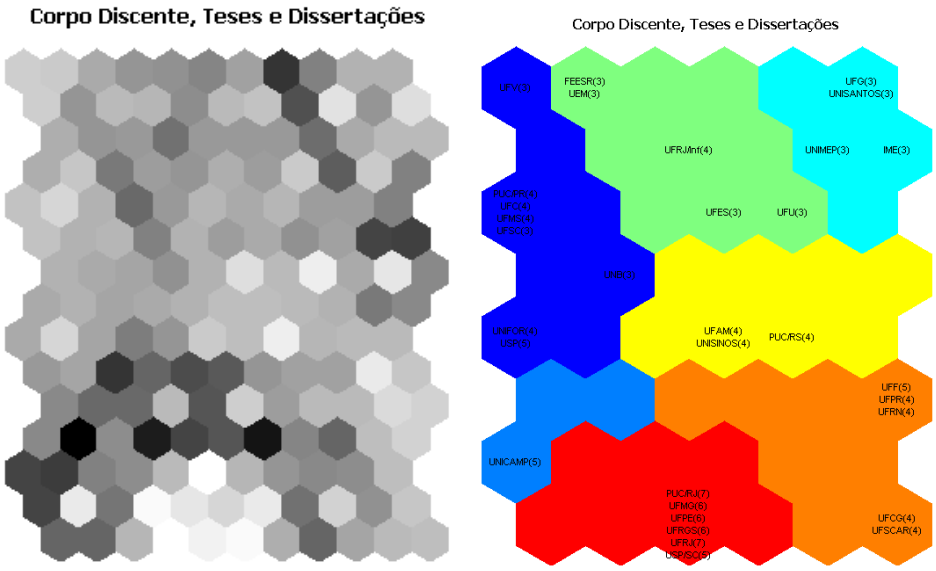


Figura 6.18: Mapas parciais do aspecto Corpo Discente da base de dados CAPES.

O maior agrupamento é representado em amarelo na figura, sendo formado em sua maioria por instituições que receberam nota final igual a 3. A exceção é a UFMS, que obteve conceitos inferiores a outras instituições de mesma categoria e foi bastante criticada em seu relatório de avaliação.

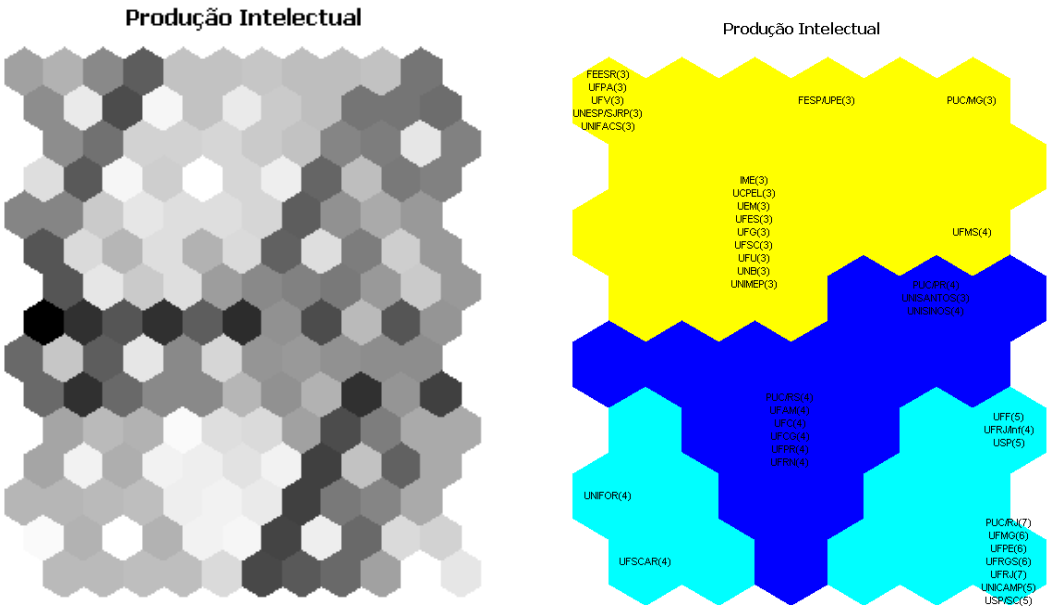


Figura 6.19: Mapas parciais do aspecto Produção Intelectual da base de dados CAPES.

O agrupamento representado em azul escuro comporta a maior parte das instituições com nota final igual a 4, com exceção da UNISANTOS, que ficou acima da média das instituições de sua categoria. O último agrupamento, em azul claro, possui as instituições mais bem avaliadas neste aspecto. Todas as instituições deste agrupamento receberam pelo menos um conceito MB, sendo que as do lado direito receberam conceito MB no primeiro atributo, que possui maior peso.

A Figura 6.20 apresenta os resultados da análise da arquitetura partSOM sobre os atributos correspondentes ao quesito 'Inserção Social'. Esse aspecto mede a importância social e atuação do programa a nível regional, nacional e internacional. Destaque para a UFCG e a USP, que possuem importância social bastante superior às da mesma categoria. Outras instituições como UFU, UNIFACS e UNIMEP também forma bem avaliadas. UNISINOS e UFRJ/Inf precisam melhorar nesse aspecto, em relação a outras instituições de mesma categoria.

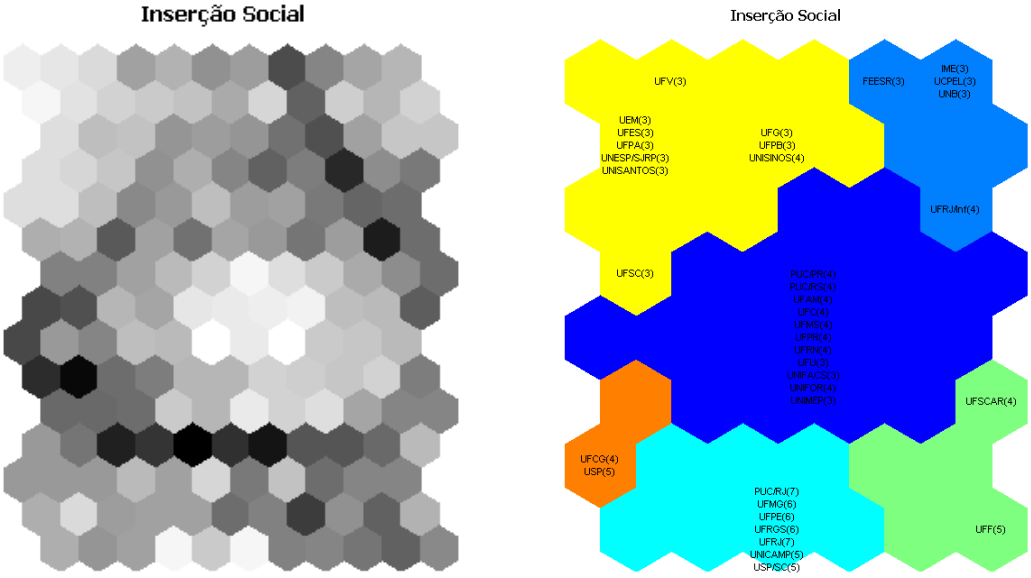


Figura 6.20: Mapas parciais do aspecto Inserção Social da base de dados CAPES.

Como pode ser visto, as avaliações parciais fornecidas pela arquitetura partSOM possibilitam uma análise bastante detalhada e bem superior à abordagem tradicional, onde todos os atributos são considerados simultaneamente. Além disso, a arquitetura ainda possibilita a avaliação final com todos os atributos, omitido nesse experimento por não apresentar nenhuma contribuição adicional.

6.3.8 Base de Dados HiperCubo

Os experimentos realizados com a base de dados Hipercubo tiveram como objetivo analisar o desempenho da arquitetura partSOM em situações onde há sobreposição dos agrupamentos provocadas pelo particionamento do conjunto de dados.

A base de dados que forma um hipercubo no espaço $\mathbb{R}^4$ é composta por um conjunto de 1600 pontos distribuídos ao longo dos 16 vértices que formam o polígono, onde cada ponto consiste em um vetor com 4 dimensões. Ao particionar a base de dados verticalmente, são criados dois subconjuntos de vetores com duas dimensões cada um. A projeção do hipercubo no espaço 2D sobrepõe alguns vértices do polígono sobre os outros, de forma que apenas 4 vértices podem ser visualizados em cada projeção, conforme pode ser visualizado na Figura 6.21.

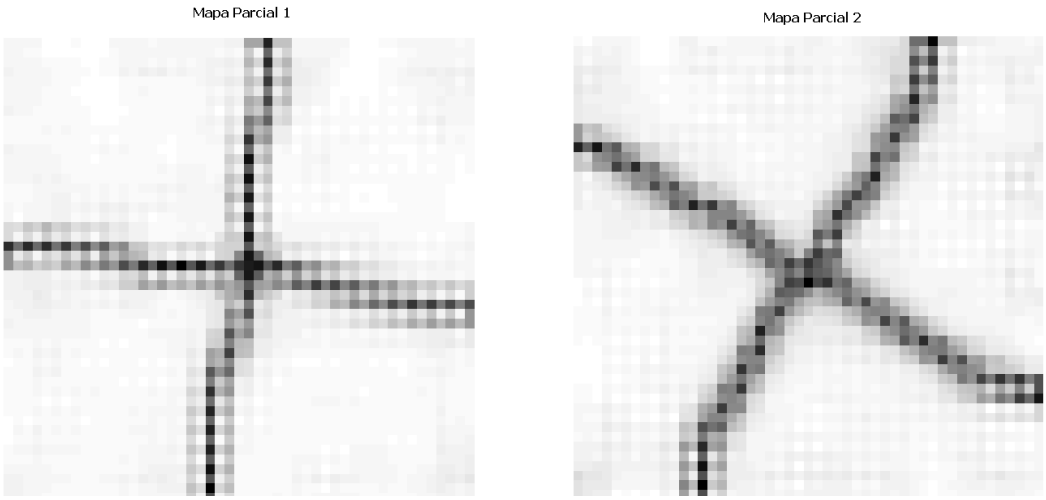


Figura 6.21: Matriz-U parciais da base de dados HiperCubo.

Inicialmente, a base de dados foi analisada com o algoritmo SOM tradicional, considerando os 4 atributos simultaneamente. Foi utilizado um mapa plano de tamanho 25 x 25, totalizando 625 neurônios, dispostos em formato retangular. Os demais parâmetros de treinamento utilizados estão apresentados na Tabela 6.44.

Tabela 6.44: Parâmetros utilizados no treinamento
da base de dados HiperCubo com 2 partições

Algoritmo	**Tamanho do Mapa**	**Fase de Aproximação**			**Fase de Ajuste Fino**		
		$\sigma_{inicial}$	σ_{final}	**épocas**	$\sigma_{inicial}$	σ_{final}	**épocas**
Som tradicional	25 x 25	4	1	4	1	1	16
partSOM mapa 1/2	25 x 25	4	1	4	1	1	16
partSOM mapa 2/2	25 x 25	4	1	4	1	1	16
partSOM mapa final	25 x 25	4	1	4	1	1	16

A Figura 6.22 apresenta a análise do conjunto de pontos pelo algoritmo SOM tradicional, onde é possível identificar os 16 agrupamentos existentes, que correspondem a cada um dos vértices do polígono. A Figura 6.23 apresenta os dados remontados pela estratégia descrita na arquitetura partSOM, onde também é possível identificar os 16 vértices do polígono.

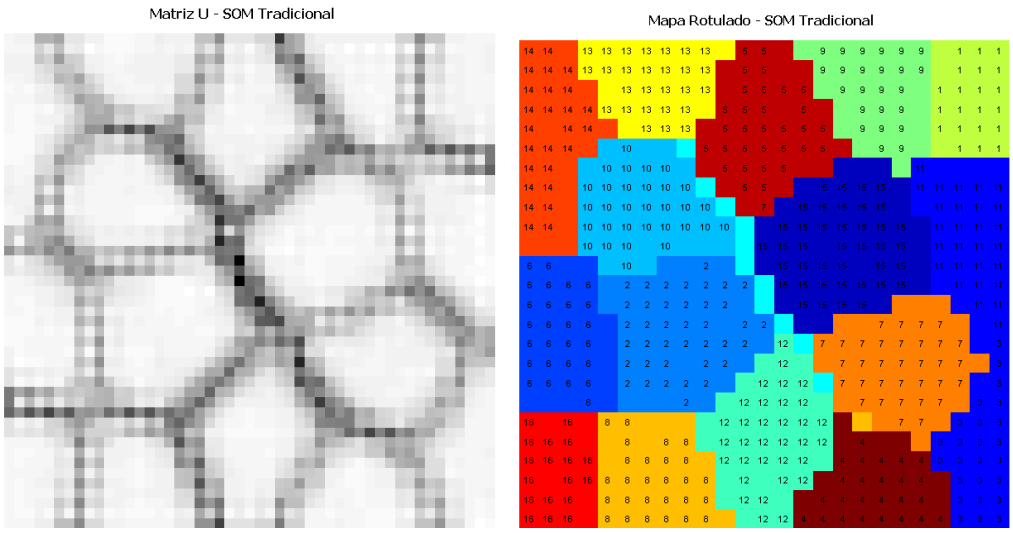


Figura 6.22: Matriz-U e mapa rotulado obtidos com o algoritmo SOM tradicional usando a base de dados HiperCubo.

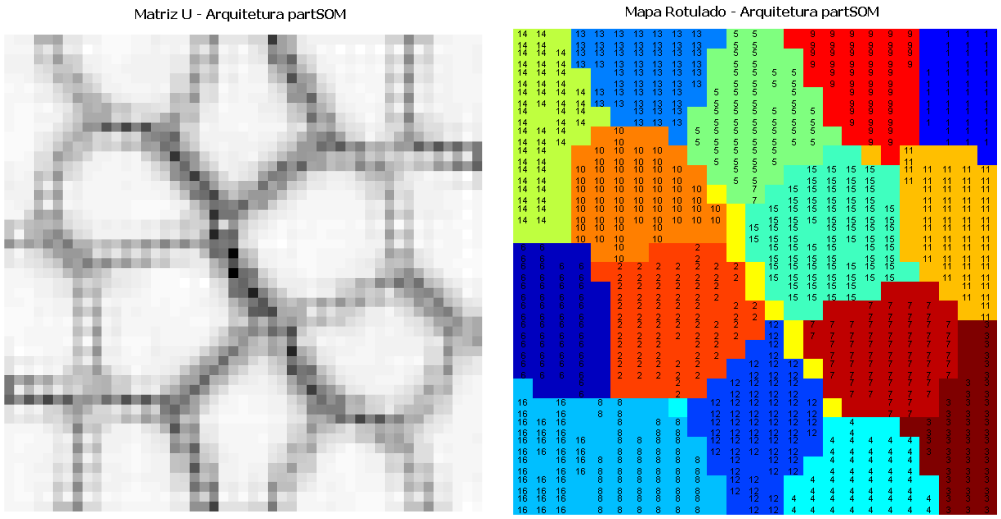


Figura 6.23: Matriz-U e mapa rotulado obtidos com a arquitetura partSOM usando a base de dados HiperCubo.

6.4 Resumo do Capítulo

Este capítulo apresentou uma série de experimentos realizados para validação da arquitetura partSOM, além da descrição da metodologia utilizada em cada experimento realizado e dos parâmetros de treinamento utilizados em cada algoritmo. Foram considerados os algoritmos SOM e *k*-médias para o agrupamento dos dados.

No próximo capítulo, que encerra esta tese, estão descritas a avaliação final da arquitetura partSOM, as conclusões obtidas com o desenvolvimento da arquitetura e algumas propostas de trabalhos futuros.

Capítulo 7
Conclusões e Trabalhos Futuros

Este capítulo apresenta uma avaliação final da arquitetura partSOM, de acordo com os resultados obtidos no capítulo anterior, além de apresentar algumas sugestões de trabalhos futuros que podem ser derivados a partir das propostas aqui apresentadas.

7.1 Considerações Finais

Os recentes avanços na área de processamento inteligente da informação, particularmente na área de descoberta de conhecimento e mineração de dados, têm motivado o acúmulo de dados por parte de empresas e organizações, que buscam na Tecnologia da Informação o desenvolvimento de ferramentas computacionais que automatizem o processo de análise, auxiliando a torná-las mais ágeis na tomada de decisões e, consequentemente, mais competitivas perante o mercado.

Atualmente, a tarefa de análise de dados requer mecanismos que permitam lidar com bases de dados cada vez maiores e, muitas vezes, distribuídas entre várias unidades. Em virtude disso, uma categoria de algoritmos tem recebido uma atenção especial, os algoritmos que realizam análise de agrupamentos distribuída, aplicados em domínios onde é necessário operar de forma descentralizada, pois não é possível (ou viável) agrupar os dados em um único local.

A análise de agrupamentos distribuída oferece diversas vantagens sobre a abordagem centralizada, uma vez que reduz sensivelmente a quantidade de dados transferidos entre as unidades distribuídas; as atualizações efetuadas nas unidades distribuídas não precisam ser replicadas na unidade central e a segurança e a privacidade dos dados podem ser mantidas, pois somente os resultados parciais de cada unidade são enviados à unidade central.

Esta tese propõe uma arquitetura para a realização de análise de agrupamentos sobre bases de dados distribuídas. A arquitetura partSOM é composta por um conjunto de algoritmos e uma estratégia de aplicação dos algoritmos sobre as várias partições do conjunto de dados que permite efetuar agrupamento sobre bases de dados distribuídas com a mesma eficácia que seria obtida utilizando-se algoritmos tradicionais de agrupamento centralizado. Posteriormente, os resultados parciais são enviados à unidade central, onde são recombinados e novamente analisados para formar o resultado final.

As principais contribuições da arquitetura proposta são:

i) Possibilidade de realizar agrupamento de dados em ambiente distribuído, tanto sobre bases de dados particionadas horizontalmente quanto verticalmente, evitando a necessidade de centralização dos dados em um único local;
ii) Uma vez que apenas um pequeno subconjunto dos dados é enviado à unidade central, obtém-se uma sensível redução na quantidade de dados transferidos entre as unidades remotas e a unidade central;
iii) Apenas os representantes dos dados são enviados à unidade central, ao invés dos dados reais, como é comum nos métodos tradicionais, o que contribui para manter a privacidade e segurança dos dados analisados;
iv) Possibilidade de realizar avaliações parciais, sob diferentes aspectos, em contrapartida aos métodos tradicionais que consideram todos os atributos simultaneamente.

A arquitetura partSOM utiliza um processo baseado em quantização vetorial para extrair um subconjunto de representantes do conjunto original de dados. Esse subconjunto de representantes mantém a distribuição estatística dos dados originais, ao mesmo tempo em que reduz a quantidade de dados a ser transferido para a unidade central. Ademais, como o subconjunto é composto por valores substitutivos dos dados, ao invés de valores reais, a segurança e privacidade dos dados originais são mantidas.

Para aplicações na área de segmentação de mercado, a arquitetura partSOM oferece como vantagem adicional a possibilidade de extrair visões parciais das bases de dados, sob diferentes aspectos, o que contribui para um melhor entendimento dos dados analisados. A arquitetura permite, por exemplo, analisar separadamente dados de clientes de diversas empresas parceiras e, numa etapa posterior, analisar os dados de um grupo de empresas, de forma que nenhuma delas tenha acesso aos dados reais das outras empresas. De maneira similar, é possível analisar aspectos parciais dos dados que estejam relacionados a um subconjunto específico de atributos, como por exemplo, analisar dados do setor de marketing, do setor pessoal, do setor financeiro, etc.

O processo de análise de dados através da arquitetura partSOM é composto por duas etapas de aplicação dos algoritmos de agrupamento de dados: uma etapa local, executada em cada partição do conjunto de dados e uma etapa global, realizada sobre os dados remontados. O custo computacional de utilização da arquitetura é superior à abordagem tradicional, uma vez que somente o custo computacional da etapa global já é equivalente ao da abordagem tradicional. Entretanto, este custo é compensado pelos benefícios que a arquitetura proporcional, seja através da redução nas transferências de dados, da contribuição na segurança dos dados analisados ou pelos resultados adicionais obtidos com os resultados parciais.

Além disso, como a execução dos algoritmos de agrupamento sobre as partições da base de dados pode ser realizada de forma independente, o processo pode ser

paralelizado, quando realizado em uma única unidade, ou executado em uma grade computacional, quando realizado em um ambiente distribuído. Dessa forma, é possível minimizar o custo computacional do algoritmo, tornando-o mais eficiente.

Os resultados obtidos nos experimentos demonstram que a arquitetura partSOM consegue obter resultados semelhantes e, em alguns casos, até superiores aos obtidos com a transferência de todas as bases de dados para uma unidade central e a posterior aplicação de algoritmos de mineração de dados convencionais. No entanto, o principal objetivo da arquitetura proposta não é competir com os métodos tradicionais, mas ser uma alternativa viável para analisar agrupamentos em ambientes onde não seja possível dispor de todos os dados de forma centralizada, seja por razões de segurança ou pelo custo relativo às transferências de dados.

7.2 Propostas de Trabalhos Futuros

Um trabalho de pesquisa é importante não apenas pelos resultados que obtêm, mas principalmente pelas possibilidades de expansão e surgimento de novas pesquisas a partir dele. A seguir, são apresentadas algumas sugestões de trabalhos futuros baseados nesta tese:

Aplicação da arquitetura partSOM a outras tarefas de mineração de dados: análise de agrupamento é apenas uma dentre várias tarefas de mineração de dados, que da mesma forma, necessitam ser realizadas em bases de dados distribuídas. Uma expansão natural deste trabalho é a aplicação da arquitetura partSOM em tarefas de mineração de dados, como classificação, regressão, sumarização, etc.;

Incremento na segurança dos dados compartilhados: Como a arquitetura partSOM é direcionada à segmentação de mercado, é muito importante considerar questões relacionadas à segurança e privacidade dos dados analisados. As técnicas aqui apresentadas permitem evitar a identificação individual dos usuários através da substituição dos dados reais por centroides. Em bases de dados muito pequenas ou com pouca variância, a privacidade pode ser comprometida, uma vez que os centroides podem ser muito próximos dos valores reais. Uma expansão natural deste trabalho é a aplicação de outras técnicas de segurança e privacidade à arquitetura partSOM;

Desenvolvimento de novos algoritmos de fusão de resultados: os algoritmos de fusão de resultados apresentados neste trabalho foram especificamente desenvolvidos para efetuar análise de agrupamentos a partir da utilização dos algoritmos *k*-médias e SOM. A possibilidade de expansão da arquitetura partSOM para utilização de outros algoritmos de agrupamento irá requerer o desenvolvimento de novos algoritmos de fusão de resultados, até mesmo para outros algoritmos que não sejam baseados em protótipos;

Análise de novos métodos de particionamento dos dados: apesar da importância que a forma de particionar os dados exerce sobre os resultados obtidos com a arquitetura proposta, essa questão não foi exaustivamente explorada neste trabalho. De certa forma, foi considerado que em aplicações reais os dados já se encontram distribuídos

geograficamente entre várias unidades e o algoritmo deve ser capaz de lidar com essa restrição, mesmo que não seja a melhor distribuição para análise. No entanto, existem situações onde a divisão dos dados pode ser realizada de acordo com a necessidade e podem-se obter ganhos significativos se forem levadas em consideração questões importantes como índices de correlação entre variáveis e significância das variáveis que compõem cada partição.

Novas aplicações para a arquitetura partSOM: Neste trabalho, a arquitetura partSOM foi analisada principalmente no contexto de segmentação de mercado. Outras aplicações para a arquitetura incluem: a análise e avaliação de produtos sob diferentes aspectos (por exemplo: design, recursos tecnológicos, dimensões, etc); agrupamento de dados em redes de sensores onde, em função do grande número de dispositivos normalmente existentes, o volume de dados transferido entre as unidades é um fator crítico; segmentação de imagens sob diferentes aspectos (cor, textura, posição, etc); entre outras.

Por fim, espera-se que a arquitetura partSOM contribua para aumentar o interesse e a realização de mais estudos e pesquisas na área de análise de agrupamentos sobre bases de dados distribuídas.

Referências Bibliográficas

Agrawal, R.; Srikant, R. (2000) *Privacy-preserving data mining*. ACM SIGMOD Record. ACM Press, v. 29, n. 2, June, pp. 439-450.

Agrawal, D.; Aggarwal, C. C. (2001) *On the design and quantification of privacy preserving data mining algorithms*. Proceedings of the Symposium on Principles of Database Systems, Santa Barbara, CA, May, pp. 247-255.

Albuquerque, M. A.; Ferreira, R. L. C.; Silva, J. A. A.; Santos, E. S.; Stosic, B.; Souza, A. L. (2006) *Estabilidade em análise de agrupamento: estudo de caso em ciência florestal*. Revista Árvore, v. 30, n. 2, abril, Viçosa, MG.

Añaña, E. S.; Vieira, L. M. M.; Petroll, M. M.; Petersen-Wagner, R. e Costa, R. S. (2008) *As comunidades virtuais e a segmentação de mercado: uma abordagem exploratória, utilizando redes neurais e dados da comunidade virtual Orkut*. Revista de Administração Contemporânea, v. 12, edição especial, Curitiba, PR, pp. 41-63.

Arroyave, G.; Lobo, O. O.; Marín, A. (2002) *A parallel implementation of the SOM algorithm for visualizing textual documents in a 2D plane*. Encuentro de Investigación sobre Tecnologías de Información Aplicadas a la Solución de Problemas (EITI2002), Medellín, Colombia.

Azevedo, F. M.; Brasil, L. M.; Oliveira, R. C. L. (2000) *Redes neurais com aplicações em controle e em sistemas especialistas*. Florianópolis: Bookstore.

Bakker, B.; Heskes, T. (2003) *Clustering ensembles of neural network models*. Neural Networks, v. 16, n. 2, March, pp. 261-269.

Berkhin, P. (2006) *A survey of clustering data mining techniques*. In: Kogan, J.; Teboulle, M.; Nicholas, C. (eds.), *Grouping multidimensional data: recent advances in clustering*, pp. 25–72, Heidelberg: Springer-Verlag.

Bhaduri, K.; Liu, K.; Kargupta, H.; Ryan, J. (2006) *Distributed data mining bibliography*. Department of Computer Science and Electrical Engineering, University of Maryland, USA. Disponível em: <http://www.cs.umbc.edu/~hillol/ DDMBIB/>. Acesso em: 07 nov. 2008.

Braga, A. P.; Carvalho, A. C. P. L. F.; Ludermir, T. B. (2007) *Redes neurais artificiais: teoria e aplicações*, 2ª edição. Rio de Janeiro: LTC.

Breiman, L. (1996) *Bagging predictors*. Machine Learning, v. 24, n. 2, pp. 123-140.

Brookshear, J. G. (2000) *Ciência da computação: uma visão abrangente*, 5ª. Edição. Porto Alegre: Bookman.

Burgaleta, J. V. C. (1992) *Influencia de los factores ecológicos en la decisión de compra de bienes de consumo repetitivo: una revisión*. Estudios sobre Consumo, año IX, num. 23, abril, pp 37-48.

Burgaleta, J. V. C. (1995) *Influencia de los factores ambientales en la decisión de compra de bienes de consumo*. ESIC Market, n. 89, julio-septiembre, pp. 125-154.

Bussab, W. O.; Miazaki, E. S.; Andrade, D. F. (1990) *Introdução à análise de agrupamento*. Anais do 9º Simpósio Brasileiro de Probabilidade e Estatística, São Paulo: IME-USP.

Calvert, D.; Guan, J. (2005) *Distributed artificial neural network architectures*. 19th International Symposium on High Performance Computing Systems and Applications, HPCS 2005, pp. 2-10.

CAPES (2009) Coordenação de Aperfeiçoamento de Pessoal de Nível Superior. Disponível em: <http://www.capes.gov.br/avaliacao/avaliacao-da-pos-graduacao>. Acesso em: 26 Jan. 2009.

Carvalho, A. C. P. L. F. *et al*. (2006), Grandes Desafios da Pesquisa em Computação no Brasil 2006 – 2016: Relatório sobre o seminário realizado em 8 a 9 de maio de 2006. Porto Alegre: Sociedade Brasileira de Computação, SBC.

Castro, A. M. G.; Lima, S. M. V.; Carvalho, J. R. P. (1999), *Planejamento de C&T: Sistemas de informação gerencial*. Brasília: Embrapa.

Chi, S.-C.; Kuo, R.-J.; Teng, P.-W. (2000) *A fuzzy self-organizing map neural network for market segmentation of credit card*. IEEE International Conference on Systems, Man, and Cybernetics, CSMC'2000, v. 5, pp. 3617-3622.

Chiang, M. M.-T.; Mirkin, B. (2007) *Experiments for the number of clusters in K-means*. LNCS, Progress in Artificial Intelligence, Berlin: Springer- Heidelberg, pp. 395-405.

Chiavenato, I. (2004) *Introdução à teoria geral da administração: Edição compacta*, 3ª. Edição. Rio de Janeiro: Elsevier.

Clifton, C.; Marks, D. (1996) *Security and privacy implications of data mining*. Proceedings of the ACM SIGMOD Workshop on Data Mining and Knowledge Discovery, Montreal, Canada, June, pp.15-19.

Clifton, C.; Kantarcioglu, M.; Vaidya, J.; Lin, X.; Zhu, M. Y. (2002), *Tools for privacy preserving distributed data mining*. SIGKDD Explorations, v. 4, n. 2, December, pp. 28-34.

Coelho, L. S.; Raittz, R. T.; Trezub, M. (2006), *FCONTROL: Sistema inteligente inovador para detecção de fraudes em operações de comércio eletrônico*. Gestão e Produção, v. 13, n. 1, UFSCar, São Carlos, SP, pp. 129-140.

Costa, J. A. F. (1999) *Classificação automática e análise de dados por redes neurais auto-organizáveis*. Tese de Doutorado, Departamento de Engenharia de Computação e Automação Industrial, Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, Campinas, SP.

Costa, J. A. F.; Andrade Netto, M. L. (1999) *Estimating the number of clusters in multivariate data by self-organizing maps*. International Journal of Neural Systems, v. 9, n. 3, pp. 195-202.

Costa, J. A. F.; Andrade Netto, M. L. (2001a) *A new tree-structured self-organizing map for data analysis*. Proceedings of the IEEE International Joint Conference on Neural Networks (IJCNN'2001), Washington, DC, July, pp. 1931-1936.

Costa, J. A. F.; Andrade Netto, M. L. (2001b) *Clustering of complex shaped data sets via Kohonen maps and mathematical morphology*. Proceedings of the SPIE, Data Mining and Knowledge Discovery, v. 4384, pp. 16-27.

Costa, J. A. F.; Andrade Netto, M. L. (2003) *Segmentação do SOM baseada em particionamento de grafos*. Anais do VI Congresso Brasileiro de Redes Neurais, São Paulo, junho, pp. 451-456.

Costa, J. A. F.; Dória Neto, A. D.; Andrade Netto, M. L. (2003) *A new structured self-organizing map with dynamic growth applied to image compression*. Anais do VI Congresso Brasileiro de Redes Neurais, São Paulo, junho, pp. 457-462.

Costa, J. A. F. (2005) *Segmentação do SOM por métodos de agrupamentos hierárquicos com conectividade restrita*. VII Congresso Brasileiro de Redes Neurais, CBRN'2005, outubro, Natal, RN.

Costa, J. A. F.; Andrade Netto, M. L. (2007) *Segmentação de mapas auto-organizáveis com espaço de saída 3-D*. SBA: Controle & Automação, v. 18, n. 2, pp. 150-162.

Damian, D.; Orešič, M.; Verheij, E.; Meulman, J.; Friedman, J.; Adourian, A.; Morel, N.; Smilde, A.; van der Greef, J. (2007) *Applications of a new subspace clustering algorithm (COSA) in medical systems biology*. Metabolomics Journal, v. 3, n. 1, March, pp. 69-77.

Davies, D. L.; Bouldin, D. W. (1979) *A cluster separation measure*. IEEE Transactions on Pattern Analysis and Machine Intelligence, v. 1, n. 2, pp. 224-227.

Daza, E. F. (2005) *Reflexiones en torno a la Responsabilidad Social de las Empresas, sus políticas de promoción y la economía social*. Revista de Economía Pública, Social y Cooperativa, n. 053, Valencia, España, noviembre, pp. 261-283.

DeWitt, D. J.; Gray, J. (1992) *Parallel database systems: the future of high performance database processing*. Communications of the ACM, v. 36, n. 6, June, pp. 85-98.

Dias, F. S. (2006) *Avaliação de sistemas de informação: revisão de publicações científicas no período de 1985-2005*. Dissertação de Mestrado, Escola de Ciência da Informação, UFMG, Belo Horizonte, MG.

Dietterich, T. G. (2000) *Ensemble methods in machine learning*. Proceedings of the First International Workshop on Multiple Classifier Systems, LNCS, v. 1857, London: Springer-Verlag, pp. 1-15.

Dimitriadou, E.; Weingessel, A.; Hornik, K. (2001) *Voting-merging: an ensemble method for clustering*. Proceedings of the International Conference on Artificial Neural Networks, LNCS, v. 2130, London: Springer-Verlag, pp. 217-224.

Du, W.; Atallah, Mikhail J. (2001) *Secure multi-party computation problems and their applications: A review and open problems*. New Security Paradigms Workshop, Cloudcroft, New Mexico, September, pp. 11-20.

D'Urso, P.; de Giovanni, L. (2008) *Temporal self-organizing maps for telecommunications market segmentation*. Neurocomputing, v. 71, n. 13-15, August, pp. 2880-2892.

Estivill-Castro, V. (2004) *Private representative-based clustering for vertically partitioned data*. Proceedings of the Fifth Mexican International Conference in Computer Science, ENC'2004, September, pp. 160-167.

Evfimievski, A.; Srikant, R.; Agrawal, R.; Gehrke, J. (2002) *Privacy preserving mining of association rules*. Proceedings of the 8th International Conference on Knowledge Discovery in Databases and Data Mining, Canada, July, pp. 217-228.

Faceli, K. (2006) *Um framework para análise de agrupamento baseado na combinação multi-objetivo de algoritmos de agrupamento*. Tese de Doutorado, Instituto de Ciências Matemáticas e de Computação (ICMC), USP, São Paulo, SP.

Farkas, C.; Jajodia, S. (2002) *The inference problem: a survey*. ACM *SIGKDD Explorations Newsletters*, v. 4, n. 2, December, pp. 6-11.

Fausett, L. (1994) *Fundamentals of neural networks: architectures, algorithms and applications*. New Jersey: Prentice Hall.

Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. (1996a) *From data mining to knowledge discovery: an overview*. In: *Advances in knowledge discovery and data mining*. Eds. American Association for Artificial Intelligence, Menlo Park, CA, pp 1-34.

Fayyad, U. M.; Piatetsky-Shapiro, G.; Smyth, P. (1996b) *The KDD process for extracting useful knowledge from volumes of data*. Communications of the ACM, v. 39, n. 11, November, pp. 27-34.

Fern, X. Z.; Lin, W. (2008) *Cluster Ensemble Selection*. Statistical Analysis and Data Mining, v. 1, n. 3, November, pp. 128-141.

Ferneda, E. (2006) *Redes neurais e sua aplicação em sistemas de recuperação de informação*. Revista Ciência da Informação, v. 35, n. 1, Janeiro/Abril, pp. 25-30.

Fleck, E. M. (2004) *Agrupamento e visualização de dados sísmicos através de quantização vetorial*. Tese de Doutorado, Departamento de Engenharia Elétrica, Pontifícia Universidade Católica do Rio de Janeiro, Rio de Janeiro, RJ.

Forman, G.; Zhang, B. (2000) *Distributed data clustering can be efficient and exact*. ACM SIGKDD Explorations Newsletter, v. 2, n. 2, December, pp. 34-38.

Frawley, W. J.; Piatetsky-Shapiro, G.; Matheus, C. J. (1992) *Knowledge discovery in databases: an overview*. AI Magazine, v. 13, n. 3.

Frei, F. (2006) *Introdução à análise de agrupamentos: teoria e prática*. São Paulo: Editora UNESP.

Freund, Y.; Schapire, R. E. (1999) *A short introduction to boosting*. Journal of Japanese Society for AI, vol. 14, no. 5, pp. 771-780.

Friedman, J. H.; Meulman, J. J. (2004) *Clustering objects on subsets of attributes*. Journal of the Royal Statistical Society: Series B (Statistical Methodology), v. 66, n. 4, pp. 815-849.

Frossyniotis, D.; Likas, A.; Stafylopatis, A. (2004) *A clustering method based on boosting*. Pattern Recognition Letters, v. 25, pp. 641-654.

Fung, B.; Wang, K.; Wang, L.; Debbabi, M. (2008) *A framework for privacy-preserving cluster analysis*. Proceedings of the IEEE International Conference on Intelligence and Security Informatics, June, pp. 46-51.

Garcia, A. C. B.; Varejão, F. M.; Ferraz, I. N. (2003) *Aquisição de Conhecimento*. In: Rezende, S. *Sistemas Inteligentes: Fundamentos e aplicações*. Barueri, SP: Manole.

Gazzaniga, M.; Ivry, R.; Mangun, G. (2006) *Neurociência cognitiva:a biologia da mente*, 2ª ed. Porto Alegre: Artmed Editora.

Georgakis, A.; Li, H.; Gordan, M. (2005) *An ensemble of SOM networks for document organization and retrieval*. Proc. of Int. Conf. on Adaptive Knowledge Representation and Reasoning, AKRR'05, Espoo, Finland, June, pp. 141-147.

Georgakis, A.; Li, H. (2006) *Content based image retrieval using a bootstrapped SOM network*. LNCS, Springer-Verlag, n. 3972, pp. 595-601.

Goldreich, O.; Micali, S.; Wigderson, A. (1987) *How to play any mental game – a completeness theorem for protocols with honest majority*. Proceedings of the 19th ACM Symposium on the Theory of Computing, pp. 218-222.

Goldschmidt, R.; Passos, E. (2005) *Data mining: um guia prático*. Rio de Janeiro: Elsevier.

Gonçalves, M. L.; Netto, M. L. A.; Costa, J. A. F.; Zullo Jr., J. (2005) *Análise de agrupamentos usando Mapas de Kohonen segmentados por morfologia matemática e índice de validação*. VII Congresso Brasileiro de Redes Neurais, Outubro, Natal, RN.

Gonçalves, M. L.; Netto, M. L. A.; Costa, J. A. F.; Zullo Jr., J. (2006) *Data clustering using self-organizing maps segmented by mathematic morphology and simplified cluster validity indexes*. Proceedings of the IEEE International Joint Conference on Neural Networks, Vancouver, July, pp. 8854-8861.

Gonçalves, M. L.; Netto, M. L. A.; Costa, J. A. F. (2007) *Explorando as propriedades do mapa auto-organizável de Kohonen na classificação de imagens de satélite*. IV Encontro Nacional de Inteligência Artificial, Julho, Rio de Janeiro, RJ.

Gonçalves, M. L.; Netto, M. L. A.; Zullo Jr., J.; Costa, J. A. F. (2008) *A new method for unsupervised classification of remotely sensed images using Kohonen self-organizing maps and agglomerative hierarchical clustering methods*. International Journal of Remote Sensing, v. 29, n. 11, June, pp. 3171-3207.

Gorgônio, F. L.; Costa, J. A. F. (2007) *Análise de agrupamentos distribuída através de múltiplos mapas auto-organizáveis*. Anais do XXII Simpósio Brasileiro de Banco de Dados, WAAMD/SBBD'07, João Pessoa, PB.

Gorgônio, F. L. (2008) *Mapas auto-organizáveis aplicados à segmentação de mercado*. I Seminário de Informática da UFRN no Seridó, UFRN/CERES, Novembro, Caicó, RN.

Gorgônio, F. L.; Santos, S. P.; Costa, J. A. F. (2008) *Uma estratégia para análise de agrupamentos em bases de dados verticalmente distribuídas*. Proceedings of the Workshop on Computational Intelligence, WCI'2008, International Joint Conference SBIA/SBRN, Salvador, BA.

Gorgônio, F. L.; Costa, J. A. F. (2008a) *Parallel self-organizing maps with application in clustering distributed data*. Proceedings of the International Joint Conference on Neural Networks, IJCNN'2008, Hong-Kong, v. 1, pp. 420.

Gorgônio, F. L.; Costa, J. A. F. (2008b) *Combining parallel self-organizing maps and K-means to cluster distributed data*. Proceedings of the 11th IEEE International Conference on Computational Science and Engineering - CSE'2008, 2008, São Paulo, SP, pp. 53-58.

Gorgônio, F. L.; Costa, J. A. F. (2008c) *PartSOM: An architecture for distributed data clustering based on multiple self-organizing maps*. Proceedings of the International

Conference on Information Systems and Technology Management, CONTECSI'08, São Paulo, SP.

Gorgônio, F. L.; Costa, J. A. F. (2010) *PartSOM: A Framework for Distributed Data Clustering Using SOM and K-Means*. In: Matsopoulos, George K. (ed.), *Self-Organizing Maps*, Vienna, Austria: InTech Education and Publishing. (Aceito para Publicação).

Gray, H. (1918) *Anatomy of the Human Body*. Philadelphia: Lea & Febiger.

Greenfield, S. A. (2000) *O cérebro humano: uma visita guiada*. Rio de Janeiro: Rocco.

Guha, S.; Rastogi, R.; Shim, K. (1998) *CURE: an efficient clustering algorithm for large databases*. Proceedings of the ACM SIGMOD Conference on Management of Data, Seattle, Washington, USA, June, pp. 73-84.

Hair Jr., J. F.; Anderson, R. E.; Tatham, R. L.; Black, W. C. (2005) *Análise Multivariada de Dados*, 5ª. edição. Porto Alegre: Bookman.

Halkidi, M.; Vazirgiannis, M. (2002) *Clustering validity assessment using multi representatives*. Proceedings of SETN Conference, Thessaloniki, Grécia.

Hämäläinen, T. D. (2002) *Parallel implementation of self-organizing maps*. In: Seiffert, U. and Jain, L. C. (eds.), *Self-Organizing Neural Networks: Recent Advances and Applications*, vol. 78, New York: Springer-Verlag, pp. 245-278.

Han, J.; Kamber, M. (2006) *Data mining: concepts and techniques*, 2nd edition. San Francisco: Morgan Kaufmann.

Haykin, S. (2001) *Redes neurais: princípios e prática*, 2ª. Edição. Porto Alegre: Bookman.

He, Z.; Xu, X.; Deng, S. (2005) *Clustering mixed numeric and categorical data: a cluster ensemble approach*. Technical report. Disponível em: <http://aps.arxiv.org/ftp/cs/papers/0509/0509011.pdf >. Acesso em: 07 jun. 2007.

Ho, T. K. (2000) *Complexity of classification problems and comparative advantages of combined classifiers*. Proceedings of the 1st International Workshop on Multiple Classifier Systems, LNCS, v. 1857, London: Springer-Verlag, pp. 97-106.

Ho, T. K. (2002) *Multiple classifier combination: lessons and the next steps*. In: Kandel, A. and Bunke, H. (editors), *Hybrid methods in pattern recognition*, World Scientific Publishing, pp. 171-198.

Hore, P.; Hall, L.; Goldgof, D. (2006) *A cluster ensemble framework for large data sets*. IEEE International Conference on Systems, Man and Cybernetics, SMC'06, October, vol. 4, pp. 3342-3347.

Hsieh, N.-C. (2004) *An integrated data mining and behavioral scoring model for analyzing bank customers*. Expert Systems with Applications, v. 27, n. 4, November, pp. 623-633.

Hung, C.; Tsai, C.-F. (2008) *Market segmentation based on hierarchical self-organizing map for markets of multimedia on demand*. Expert Systems with Applications, v. 34, n. 1, January, pp. 780-787.

İnan, A.; Saygın, Y.; Savaş, E.; Hintoğlu, A. A.; Levi, A. (2006) *Privacy preserving clustering on horizontally partitioned data*. Proceedings of the 22nd International Conference on Data Engineering Workshops, pp. 95-103.

İnan, A.; Kaya, S. V.; Saygın, Y.; Savaş, E.; Hintoğlu, A. A.; Levi, A. (2007) *Privacy preserving clustering on horizontally partitioned data*. Data & Knowledge Engineering, v. 63, n. 3, December, pp. 646-666.

Izaguirre, J.; Vicente, A.; Tamayo, U. (2007) *Medio ambiente y competitividad ¿obstáculo u oportunidad?: una aproximación a partir de la evidencia empírica*. XIX Congreso anual y XV Congreso Hispano Francés de AEDEM, v. 1, pag. 40.

Jagannathan, G.; Wright, R. N. (2005) *Privacy-preserving distributed k-means clustering over arbitrarily partitioned data*. Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining, KDD'05, pp. 593-599.

Jagannathan, G.; Pillaipakkamnatt, K.; Wright, R. N. (2006) *A new privacy-preserving distributed k-clustering algorithm*. Proceedings of the 2006 SIAM International Conference on Data Mining (SDM), pp. 492-496.

Jain, A. K.; Murty, M. N.; Flynn, P. J. (1999) *Data clustering: a review*. ACM Computing Surveys, v. 31, n. 3, September, pp. 264-323.

Jha, S.; Kruger, L.; McDaniel, P. (2005) *Privacy Preserving Clustering*. Proceedings of the 10th European Symposium on Research in Computer Security, pp. 397-417.

Jian, L.; Bangzhu, Z. (2007) *Neural network ensemble based on feature selection*. IEEE International Conference on Control and Automation, ICCA'2007, May-June, pp. 1844-1847.

Jiang, Y.; Zhou, Z.-H. (2004) *SOM ensemble-based image segmentation*. Neural Processing Letters, v. 20, n. 3, November, pp. 171-178.

Johnson, S. (2003) *Emergência: a vida integrada de formigas, cérebros, cidades e softwares*. Rio de Janeiro: Jorge Zahar Editor.

Kamimura, R. (2003) *Competitive learning by information maximization: eliminating dead neurons in competitive learning*. Proceedings of the Joint International Conference ICANN/ICONIP, LNCS, Berlin: Springer, pp. 99-106.

Kandel, E.; Schwartz, J.; Jessell, T. (1997) *Fundamentos da Neurociência e do Comportamento*. Rio de Janeiro: Guanabara Koogan.

Kantarcioglu, M.; Vaidya, J. (2002) *An architecture for privacy-preserving mining of client information*. In Clifton, C. and Estivill-Castro, V. (eds.), *ACM International Conference Proceeding Series*, v. 144, Darlinghurst: Australian Computer Society, pp. 37-42.

Kantarcioglu, M. (2008) *A survey of privacy-preserving methods across horizontally partitioned data*. In: Aggarwal, C. C. and Yu, P. S., *Privacy-preserving data mining*, Springer, pp. 313-336.

Kantardzic, M. (2002) *Data mining: concepts, models, methods, and algorithms*. New York: Wiley-IEEE Press.

Kapoor, V.; Poncelet, P.; Trousset, F.; Teisseire, M. (2006) *Privacy preserving sequential pattern mining in distributed databases.* Proceedings of the 15th ACM International Conference on Information and Knowledge Management, CIKM '06, New York, NY, pp. 758-767.

Kapoor, V.; Poncelet, P.; Trousset, F.; Teisseire, M. (2007) *Préservation de la vie privée: recherche de motifs séquentiels dans des bases de données distribuées*. Revue Ingénierie des Systèmes d'Information (ISI), v.12, n. 1, Décembre, France, pp. 85-107.

Kargupta, H.; Park, B. H.; Hershberger, D.; Johnson, E. L. (2000) *Collective data mining: a new perspective toward distributed data mining*. In Kargupta, H. and Chan, P. (editors), *Advances in Distributed and Parallel Knowledge Discovery*, MIT/AAAI Press, pp. 131-178.

Kargupta, H.; Huang, W.; Sivakumar, K.; Johnson, E. L. (2001) *Distributed clustering using collective principal component analysis*. Knowledge and Information Systems, v. 3, n. 4, pp. 422-448.

Kasznar, I. K.; Gonçalves, B. M. L. (2007) *Técnicas de agrupamento: clustering*. EletroRevista: Revista Científica e Tecnológica, Rio de Janeiro: IBCI, v. 6, n. 20.

Kiang, M. Y. (2001) *Extending the Kohonen self-organizing map networks for clustering analysis*. Computational Statistics & Data Analysis, v. 38, n. 2, December, pp. 161-180.

Kiang, M. Y.; Kumar, A. (2001) *An evaluation of self-organizing map networks as a robust alternative to factor analysis in data mining applications*. Information Systems Research, v. 12, n. 2, June, pp. 177-194.

Kiang, M. Y.; Fisher, D. M.; Hu, M. Y.; Chi, R. (2004) *Using an extended self-organizing map network to forecast market segment membership*. In: Zhang, P. (ed). *Neural Networks in Business Forecasting*, Idea Group Inc.

Kiang, M. Y.; Kumar, A. (2004). *A comparative analysis of an extended SOM network and K-means analysis*. International Journal of Knowledge-Based and Intelligent Engineering Systems, v. 8, n. 1, pp. 9-15.

Kiang, M. Y.; Hu, M. Y.; Fisher, D. M.; Chi, R. T. (2005) *The effect of sample size on the extended self-organizing map network for market segmentation*. Proceedings of the 38th Annual Hawaii International Conference on System Sciences (HICSS'05), v. 3, track 3, p. 73.2.

Kiang, M. Y.; Hu, M. Y.; Fisher, D. M. (2006) *An extended self-organizing map network for market segmentation – a telecommunication example*. Decision Support Systems, v. 42, n. 1, October, pp. 36-47.

Kiang, M. Y.; Hu, M. Y.; Fisher, D. M. (2007) *The effect of sample size on the extended self-organizing map network – A market segmentation application*. Computational Statistics & Data Analysis, v. 51, n. 12, August, pp. 5940-5948.

Kiang, M. Y.; Fisher, D. M. (2008) *Selecting the right MBA schools – An application of self-organizing map networks*. Expert Systems with Applications, v. 35, n. 3, October, pp. 946-955.

Kim, J.; Wei, S.; Ruys, H. (2003) *Segmenting the market of West Australian senior tourists using an artificial neural network*, Tourism Management, v. 24, n. 1, pp. 25-34.

Kim, K.-J.; Ahn, H. (2008) *A recommender system using GA K-means clustering in an online shopping market*, Expert Systems with Applications, v. 34, n. 2, pp. 1200-1209.

Kim, M.; Ramakrishna, R. S. (2005) *New indices for cluster validity assessment*. Pattern Recognition Letters, v. 26, n. 5, November, pp. 2353-2363.

Klepaczko, A.; Materka, A. (2004) *Clustering quality based feature selection method*. Machine Graphics & Vision International Journal, v. 13, n. 4, October, pp. 355-376.

Klusch, M.; Lodi, S.; Moro, G. (2003) *Distributed clustering based on sampling local density estimates*. Proceedings of the 19th International Joint Conference on Artificial Intelligence, IJCAI'03, pp. 485-490.

Kohonen, T. (1982a) *Self-organizing formation of topologically correct feature maps*. Biological Cybernetics, v. 43, pp. 59-69.

Kohonen, T. (1982b) *Analysis of a simple self-organizing process*. Biological Cybernetics, v. 44, pp. 135-140.

Kohonen, T. (1984) *Self-organization and associative memory*. Berlin: Springer.

Kohonen, T. (1990) *The self-organizing map*. Proceedings of IEEE, v. 78, n. 9, September, pp. 1464-1480.

Kohonen, T. (2001) *Self-organizing maps*, 3rd edition. Berlin: Springer.

Koikkalainen, P. (1994) *Progress with the tree-structured self-organizing map*. In: Cohn, A. G. (Ed.), 11th European Conference on Artificial Intelligence, European Committee for Artificial Intelligence, New York: Wiley.

Kotler, P.; Keller, K. L. (2006) *Administração de marketing*, 12ª. Edição. São Paulo: Prentice Hall.

Kuncheva, L. I. (2003) *That elusive diversity in classifier ensembles*. Lecture Notes in Computer Science, pp. 1126-1138.

Kuncheva, L. I. (2004) *Combining pattern classifiers: methods and algorithms*. New Jersey: John Wiley & Sons.

Kuo, R. J.; An, Y. L.; Wang, H. S.; Chung, W. J. (2006) *Integration of self-organizing feature maps neural network and genetic K-means algorithm for market segmentation*. Expert Systems with Applications. v. 30, n. 2, February, pp. 313-324.

Laaksonen, J.; Koskela, M.; Oja, E. (1999) *PicSOM: self-organizing maps for content-based image retrieval*. International Joint Conference on Neural Networks, IJCNN '99, v. 4, pp. 2470-2473.

Laaksonen, J.; Koskela, M.; Laakso, S.; Oja, E. (2000) *PicSOM – content-based image retrieval with self-organizing maps*. Pattern Recognition Letters v. 21, n. 13-14, December, pp. 1199-1207.

Laaksonen, J.; Koskela, M.; Oja, E. (2002) *PicSOM – Self-organizing image retrieval with MPEG-7 content descriptors*. IEEE Transactions on Neural Networks, v. 13, n. 4, July, pp. 841-853.

Laine, S. (2002) *Selecting the variables that train a self-organizing map (SOM) which best separates predefined clusters*. Neural Information Processing, 2002. ICONIP '02. Proceedings of the 9th International Conference on, v. 4, n. 18-22, November, pp. 1961-1965.

Lam, C.-M.; Zhang, X.; Cheung, W. K.-W. (2004) *Mining local data sources for learning global cluster models*. Proceedings of the IEEE/WIC/ACM International Conference on Web Intelligence, v. 20, n. 24, September, pp. 748-751.

Larose, D. T. (2005) *Discovering knowledge in data: an introduction to data mining.* Hoboken: John Wiley & Sons, Inc.

Las Casas, A. L. (2006) *Plano de marketing para micro e pequena empresa*, 4ª. edição. São Paulo: Atlas.

Lee, J. H.; Park, S. C. (2005) *Intelligent profitable customers segmentation system based on business intelligence tools*. Expert Systems with Applications, v. 29, n. 1, July, pp. 145-152.

Lee, S.-C.; Xiang, J.-Y.; Jing, L.-B. (2005) *Who are the target customers in Chinese online game market?: segmentation with a two-step approach*. IEEE International Conference on e-Business Engineering, ICEBE'2005, pp. 736-742.

Leisch, F. (1998) *Ensemble methods for neural clustering and classification*. PhD Thesis, Institut für Statistik, Wahrscheinlichkeitstheorie und Versicherungsmathematik, Teschnische Universität Wien, Austria.

León, F. (2008) *La percepción de la responsabilidad social empresarial por parte del consumidor*. Visión Gerencial, año 7, n. 1, enero-junio, pp. 83-95.

Likert, R. (1932), *A technique for the measurement of attitudes*, Archives of Psychology v. 22, n. 140, pp. 1-55.

Lin, X.; Clifton, C.; Zhu, M. (2005) *Privacy-preserving clustering with distributed EM mixture modeling*. Knowledge Information Systems, v. 8, n. 1, July, pp. 68-81.

Lin, T-W.; Yang, H.-E. (2006) *Female consumer behavior in a banking environment*. IEEE International Conference on Service Operations and Logistics, and Informatics, SOLI '06, June, pp. 564-569.

Linde, Y.; Buzo, A.; Gray, R. M. (1980) *An algorithm for vector quantizer design*. IEEE Transactions on Communications, v. 28, n. 1, January, pp. 84-95.

Lindell, Y.; Pinkas, B. (2000) *Privacy preserving data mining*. Proceedings of the 20th Annual International Cryptology Conference on Advances in Cryptology, August, pp. 36-54.

Luo, H.-L.; Xie, X.-B.; Li, K. (2007) *A new method for constructing clustering ensembles*. Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition, ICWAPR '07, pp. 874-878.

MacQueen, J. B. (1967) *Some methods for classification and analysis of multivariate observations*. Proc. 5th Berkeley Symposium on Mathematical Statistics and Probability, Berkeley, University of California Press, v. 1, pp. 281-297.

Mañas, A. V. (2005) *Administração de sistemas de informação*, 6ª. edição. São Paulo: Érica.

Marques, M. C. (2008) *Comparação entre os métodos de agrupamentos K-means e mapa de Kohonen (SOM) em análise de pesquisa de mercado*. Revista de Inteligência Computacional Aplicada, PUC-Rio, Rio de Janeiro, RJ, Abril, v. 1, n. 1.

McCulloch, W. S.; Pitts, W. (1943) *A logical calculus of the ideas immanent in nervous activity*. The Bulletin of Mathematical Biophysics, v. 5, n. 4, pp. 115-133.

Merugu S.; Ghosh, J. (2003) *Privacy-preserving distributed clustering using generative models*. Proceedings of the 3rd IEEE International Conference on Data Mining ICDM'03, pp. 211-218.

Mirkin, B. (2005) *Clustering for data mining: a data recovery approach*. Computer science and data analysis series, Boca Raton: Chapman and Hall/CRC.

Moresi, E. A. D. (2000) *Delineando o valor do sistema de informação de uma organização*. Ci. Inf., Brasília, v. 29, n. 1, Janeiro/Abril, pp. 14-24.

Moutarde, F.; Ultsch, A. (2005) *U*F Clustering: a new performant cluster-mining method based on segmentation of self-organizing maps*. Proceedings of the Workshop on Self-Organizing Maps, WSOM'2005, Paris, France, pp. 25-32.

Murtagh, F. (1995) *Interpreting the Kohonen self-organizing feature map using contiguity-constrained clustering*. Pattern Recognition Letters, v. 16, pp. 399-408.

Myers, L. (2000) *Marketing de e-business: o caminho para o sucesso*. Redwood Shores: Oracle Corporation.

Nakagawa, M. (1987) *Estudo de Alguns Aspectos de Controladoria que Contribuem para a Eficácia Gerencial*. Tese de Doutorado. Faculdade de Economia, Administração e Contabilidade, Universidade de São Paulo, SP.

Neagoe, V.-E.; Ropot, A.-D. (2001) *Concurrent self-organizing maps for automatic face recognition*. Proceedings of the 29th International Conference of the Romanian Technical Military Academy, published by Technical Military Academy, Bucharest, Romania, November, pp. 35-40.

Neagoe, V.-E.; Ropot, A.-D. (2002) *Concurrent Self-organizing maps for pattern classification*. Proceedings of First IEEE International Conference on Cognitive Informatics (ICCI'02), pp. 304.

Neagoe, V.-E.; Ropot, A.-D. (2004) *Concurrent self-organizing maps – a powerful artificial neural tool for biometric technology*. Proceedings of IEEE World Automation Congress (WAC'04), v. 17, Seville.

Newman, D. J.; Hettich, S.; Blake, C. L.; Merz, C. J. (1998) *UCI repository of machine learning databases*. Department of Information and Computer Science, University of California, Irvine, CA, USA. Disponível em: <http://www.ics.uci.edu/ ~mlearn/MLRepository.html>. Acesso em: 07 Ago. 2008.

O'Leary, D. E. (1991) *Knowledge discovery as a threat to database security*. In: Piatetsky-Shapiro, G. and Frawley, W. J. (eds.), *Knowledge discovery in databases*, Menlo Park: AAAI/MIT Press, pp. 507-516.

Oliveira, S. R. M.; Zaïane, O. R. (2003) *Privacy preserving clustering by data transformation*. Proceedings of the 18th Brazilian Symposium on Databases, pp. 304-318.

Oliveira, S. R. M.; Zaïane, O. R. (2004) *Privacy preservation when sharing data for clustering*. Proceedings of the International Workshop on Secure Data Management in a Connected World, Toronto, pp. 67-82.

Oliveira, S. R. M.; Zaiane, O. R. (2007) *A privacy-preserving clustering approach toward secure and effective data analysis for business collaboration*. Computers & Security, v. 26, p. 81-93.

Opitz, D. W. (1999) *Feature selection for ensembles*. Proceedings of the 16th National Conference on Artificial Intelligence, Orlando, USA, July, pp. 379-384.

Opolon, D.; Moutarde, F. (2004) *Fast semi-automatic segmentation algorithm for self-organizing maps*. Proceedings of the European Symposium on Artificial Neural Networks, ESANN'2004, Bruges, Belgium, Avril, pp. 507-512.

Oza, N. C.; Tumer, K. (2008) *Classifier ensembles: select real-world applications*. Information Fusion, v. 9, n. 1, January, pp. 4-20.

Paço, A. M. F. (2002) *Segmentação comportamental da população universitária relativamente ao consumo de vestuário e suas implicações para o comércio retalhista*. Dissertação de Mestrado, Mestrado em Gestão, Universidade da Beira Interior, Covilhã, Portugal.

Paço, A. M. F. (2003) *Análise das atitudes de compra dos consumidores líderes e dos consumidores seguidores no caso do vestuário*. XIII Jornadas Hispano-Lusas de Gestión Cientifica, Universidade de Lugo, España.

Paço, A. M. F. (2005) *O perfil dos consumidores de vestuário – um estudo de segmentação de mercado*. In: Martín, E. e Cossío, F. J. (eds.), *XV Spanish-Portuguese Meeting of Scientific Management - Cities in Competition*, New Trends in Marketing Management, Sevilha, España, pp. 693-707.

Paço, A. M. F. (2007) *O consumidor de moda: um estudo de segmentação de mercado*. Revista Convergências, Universidade da Beira Interior, Covilhã, Portugal. Disponível em: <http://convergencias.esart.ipcb.pt/artigos_tema/ Design-de-Moda>. Acesso em: 21 Ago. 2008.

Pedrycz, W. (2005) *Knowledge-based clustering: from data to information granules*. Hoboken: John Wiley & Sons, Inc.

Piatetsky-Shapiro, G. (1995) *Knowledge discovery in personal data vs. privacy: a mini-symposium*. IEEE Expert: Intelligent Systems and Their Applications v. 10, n. 2, April, pp. 46-47.

Pinker, S. (1998) *Como a mente funciona*. São Paulo: Companhia das Letras.

Pölzlbauer, G. (2004) *Survey and comparison of quality measures for self-organizing maps*. Proceedings of the Fifth Workshop on Data Analysis (WDA'04), June, pp. 67-82.

Punj, G.; Stewart, D. W. (1983) *Cluster analysis in marketing research: review and suggestions for application*. Journal of Marketing Research, v. 20, n. 2, May, pp.134-48.

Rezende, S. O. (2003) *Sistemas inteligentes: fundamentos e aplicações*. Barueri, SP: Manole.

Roli, F.; Giacinto, G.; Vernazza, G. (2001) *Methods for designing multiple classifier systems*. Proceedings of the 2nd International Workshop on Multiple Classifier Systems, LNCS, v. 2096, London: Springer-Verlag, pp. 78-87.

Rosario, G. E.; Rundensteiner, E. A.; Brown, D. C.; Ward, M. O. (2004) *Mapping nominal values to numbers for effective visualization*. Information Visualization, v. 3, n. 2, June, pp. 80-95.

Rutkowski, L. (2005) *Computational intelligence: methods and techniques*. Berlin: Springer-Verlag.

Sabbatini, R. M. E.; Cardoso, S. H. (2002), *Interdisciplinarity and the Neurosciences*, Interdisciplinary Science Review, v. 27, n. 4, December, pp. 303-311.

Saitta, S.; Raphael, B.; Smith, I. F. (2007) *A bounded index for cluster validity*. Proceedings of the 5th International Conference on Machine Learning and Data Mining in Pattern Recognition, LNCS, v. 4571, Berlin: Springer-Verlag, pp. 174-187.

Salazar, E.; Arroyave, G.; Ortega, O. (2001) *Evaluating several unsupervised class-selection methods*. Encuentro de Investigación sobre Tecnologías de Información Aplicadas a la Solución de Problemas (EITI2001), Medellín, Colombia.

Salazar, E.; Vélez, A. C.; Parra, C. M.; Ortega, O. (2002) *A cluster validity index for comparing non-hierarchical clustering methods*. Encuentro de Investigación sobre Tecnologías de Información Aplicadas a la Solución de Problemas (EITI2002), Medellín, Colombia, 2002.

Santana, L. E. A.; Oliveira, D. F.; Canuto, A. M. P.; Souto, M. C. P. (2007) *A comparative analysis of feature selection methods for ensembles with different combination methods*. Proceedings of IEEE International Joint Conference on Neural Networks (IJCNN'2007), Orlando, USA, pp. 643-648.

Schiffman, L. G.; Kanuk, L. L. (2000) *Comportamento do consumidor*, 6ª. edição. Rio de Janeiro: Editora LTC.

Sebesta, R. W. (2003) *Conceitos de linguagens de programação*, 5ª. edição. Porto Alegre: Bookman.

Serrentino, A. (2006) *Inovações no varejo: decifrando o quebra-cabeça do consumidor*, 2ª. edição. São Paulo: Saraiva.

Setzer, V. W. (1999) *Dado, informação, conhecimento e competência*. DataGramaZero – Revista de Ciência da Informação, Número Zero, dezembro, Rio de Janeiro, RJ.

Shaw, M. J.; Subramaniam, C.; Tan, G. W.; Welge, M. E. (2001) *Knowledge management and data mining for marketing*. Decision Support Systems, v. 31, pp. 127-137.

Shim, Y.; Chung, J.; Choi, I.-C. (2005) *A comparison study of cluster validity indices using a nonhierarchical clustering algorithm*. Proceedings of the International Conference on Computational Intelligence for Modeling, Control and Automation CIMCA, Washington, DC, pp. 199-204.

Silva, J. C.; Klusch, M. (2006) *Inference in distributed data clustering*. Engineering Applications of Artificial Intelligence, vol. 19, pp. 363-369.

Silva, S. C. M. (2006) *Comitês de agrupamento aplicados a dados de expressão gênica*. Dissertação de Mestrado, DIMAP/UFRN, Universidade Federal do Rio Grande do Norte, Natal, RN.

Sneath, P. H. A.; Sokal R. R. (1973) *Numerical taxonomy: The principles and practice of numerical classification*. San Francisco: W. H. Freeman and Company.

Soares, J. (2007) *Pré-processamento em mineração de dados: um estudo comparativo em complementação*. Tese de Doutorado, COPPE/UFRJ, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ.

Son, J. H.; Kim, M. H. (2004) *An adaptable vertical partitioning method in distributed systems*. Journal of Systems and Software, v. 73, n. 3, pp. 551-561.

Sousa, M. S. R. (1998) *Mineração de dados: uma implementação fortemente acoplada a um sistema gerenciador de banco de dados paralelo*. Dissertação de Mestrado, Coordenação dos Programas de Pós-Graduação de Engenharia, Universidade Federal do Rio de Janeiro, Rio de Janeiro, RJ.

Steenkamp, J.-B. E. M.; Hofstede, F. T. (2002) *International market segmentation: issues and perspectives*. International Journal of Research in Marketing, v. 19, n. 3, September, pp. 185-213.

Strehl, A. (2002) *Relationship-based clustering and cluster ensembles for high-dimensional data mining*. PhD Thesis, The University of Texas at Austin, Austin, Texas, USA.

Strehl, A.; Ghosh, J. (2002) *Cluster ensembles – a knowledge reuse framework for combining multiple partitions*. Journal of Machine Learning Research v. 3, March, pp. 583-617.

Tan, P.-N.; Steinbach, M.; Kumar, V. (2006) *Introduction to data mining*. Boston: Pearson Addison Wesley.

Teixeira, J. F. (1998) *Mentes e máquinas: uma introdução à ciência cognitiva*. Porto Alegre: Artes Médicas.

Thomazi, S. M. (2006) *Cluster de turismo: introdução ao estudo de arranjo produtivo local*. São Paulo: Aleph.

Tryon, R. C. (1939) *Cluster analysis.* Ann Arbor: Edwards Brothers.

Tumer, K.; Agogino, A. K. (2008) *Ensemble clustering with voting active clusters*. Pattern Recognition Letters, v. 29, n. 14, October, pp. 1947-1953.

Ultsch, A. (1993) *Knowledge extraction from self-organizing neural networks*. In: Opitz *et al*. (Ed.), *Information and classification*. Berlin: Springer-Verlag.

Ultsch, A. (2002) *Emergent Self-Organizing Feature Maps used for Prediction and Prevention in Mobile Phone Markets*. Jornal of Targeting, v. 10, n./4, Steward, London, pp. 401-425.

Ultsch, A. (2003) *U*-Matrix: a tool to visualize clusters in high dimensional data*. Technical Report, n. 36, Dept. of Mathematics and Computer Science, University of Marburg, Germany.

Ultsch, A. (2005) *Clustering with SOM: U*C*. Proceedings of the Workshop on Self-Organizing Maps, Paris, France, pp. 75-82.

Ultsch, A.; Moerchen, F. (2005) *ESOM-Maps: tools for clustering, visualization, and classification with Emergent SOM*. Technical Report, n. 46, Dept. of Mathematics and Computer Science, University of Marburg, Germany.

Urdaneta, I. P. (1992) *Gestión de la Inteligencia, Aprendizaje Tecnológico y Modernización del Trabajo Informacional: Retos y oportunidades*. Caracas: Universidad Simón Bolivar, 1992.

Vaidya, J.; Clifton, C. (2003) *Privacy-preserving k-means clustering over vertically partitioned data*. In: Proc. of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, pp. 206-215.

Vaidya, J.; Clifton, C.; Zhu, M. Y. (2006) *Privacy preserving data mining*. New York: Springer.

Vaidya, J. (2008) *A survey of privacy-preserving methods across vertically partitioned data*. In: Aggarwal, C. C. and Yu, P. S., *Privacy-preserving data mining*, Springer, pp. 337-358.

Vale, K.; Dias, F.; Canuto, A. M. P.; Souto, M. C. P. (2008) *A class-based feature selection method for ensemble systems*. Proceedings of the Hybrid Intelligent Systems Conference (HIS), Barcelona, Spain, pp. 596-596.

Valentim, M. L. P. (2002) *Inteligência competitiva em organizações: dado, informação e conhecimento*. DataGramaZero – Revista de Ciência da Informação, v. 4, n. 3, agosto, Rio de Janeiro, RJ.

Vellido, A.; Lisboa, P. J. G.; Meehan, K. (1999) *Segmentation of the On-line Shopping Market Using Neural Networks*. Expert Systems with Applications, vol. 17, pp. 303-314.

Vellido, A.; Lisboa, P. J. G.; Meehan, K. (2002) *Characterizing and Segmenting the On-Line Customer Market Using Neural Networks*. In: J. Segovia, P. S. Szczepaniak, M. Niedzwiedzinski (eds.). E-Commerce and Intelligent Methods. Heidelberg: Springer-Verlag, pp. 101-119.

Verykios, V. S.; Bertino, E.; Fovino, I. N.; Provenza, L. P.; Saygin, Y.; Theodoridis, Y. (2004) *State-of-the-art in privacy preserving data mining*. ACM SIGMOD Records, v. 33, n. 1, March, pp. 50-57.

Vesanto, J.; Alhoniemi, E. (2000) *Clustering of the self-organizing map*. IEEE Transactions on Neural Networks, vol. 11, iss. 3, May, pp. 586-600.

Vin, T.; Seng, M. P. Z.; Kuan, N. P.; Haron, F. (2005) *A framework for grid-based neural networks*. Proceedings of the 1st Int. Conf. on Distributed Frameworks for Multimedia Applications, pp. 246-253.

Vrusias, L, Vomvoridis, and Gillam,

Vrusias, B. L.; Vomvoridis, L.; Gillam, L. (2007) *Distributing SOM ensemble training using grid middleware*. Proceedings of the International Joint Conference on Neural Networks, IJCNN 2007, pp. 2712-2717.

Wang, K. J. (2007a) *Cluster validation toolbox for estimating the number of clusters*. Disponível em: <http://www.mathworks.com/matlabcentral/fileexchange/ 13916>. Acesso em: 18 Dez. 2008.

Wang, K. J. (2007b) *Cluster validation toolbox CVAP*. Disponível em: <http:// www.mathworks.com/matlabcentral/fileexchange/14620>. Acesso em: 10 nov. 2008.

Whitby, B. (2004) *Inteligência artificial: um guia para iniciantes*. São Paulo, Madras.

Xiaoguang, W.; Wangliangyusheng; Junping, L. (2007) *Risk Control and Estimation of Online Services Market Segmentation*. International Conference on Service Systems and Service Management, pp. 1-5.

Xu, R.; Wunsch II, D. (2005) *Survey of clustering algorithms*. IEEE Transaction on Neural Networks, v. 16, n. 3, May, pp. 645-678.

Yang, M.-H.; Ahuja, N. (1999) *A data partition method for parallel self-organizing map*. Proceedings of the International Joint Conference on Neural Networks, IJCNN '99, v. 3, pp. 1929-1933.

Yao, A. C. (1986) *How to generate and exchange secrets*. Proceedings of the 27th IEEE Symposium on Foundations of Computer Science, pp. 162-167.

Yin, H. (2002) *ViSOM – A novel method for multivariate data projection and structure visualization*. IEEE Transactions on Neural Networks, v. 13, pp. 237–243.

Yin, H. (2008) *On multidimensional scaling and the embedding of self-organizing maps.* Neural Networks, v. 21, n. 2-3, March/April, pp. 160-169.

Yu, Z.; Zhang, S.; Wong, H-S.; Zhang, J. (2007) *Image segmentation based on cluster ensemble*. The Fourth International Symposium on Neural Networks, China, June, pp. 894-903

Zakrzewska, D.; Murlewski, J. (2005) *Clustering algorithms for bank customer segmentation*. Proceedings of the 5th International Conference on Intelligent Systems Design and Applications, ISDA'05, pp. 197-202.

Zhan, J. (2007) *Privacy preserving K-medoids clustering*. IEEE International Conference on Systems, Man and Cybernetics, ISIC'2007, October, pp.3600-3603.

Zhan, J. (2008) *Privacy-preserving collaborative data mining*. IEEE Computational Intelligent Magazine, May, pp. 31-41.

Zhang, X.; Lam, C-M; Cheung, W. K.-W. (2004) *Mining Local Data Sources for Learning Global Cluster Models Via Model Exchange*. IEEE Intelligent Informatics Bulletin, v. 4, pp. 16-22.

Zhang, S.; Wu, X.; Zhang, C. (2003) *Multi-database mining*. IEEE Computational Intelligence Bulletin, v. 2, n. 1, June, pp. 5-13.

Zhao, Y.; Gao, J.; Yang, X. (2005) *A survey of neural network ensembles*. International Conference on Neural Networks and Brain, ICNN&B'05, October, v. 1, pp. 438-442.

Zhou, Z.-H., Tang, W. (2006) *Clusterer ensemble*. Knowledge-Based Systems, v. 19, n. 1, March, pp. 77-83.