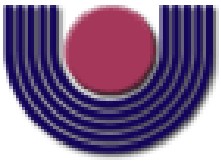


UNIOESTE – Universidade Estadual do Oeste do Paraná

CENTRO DE CIÊNCIAS EXATAS E TECNOLÓGICAS

Colegiado de Ciência da Computação

Curso de Bacharelado em Ciência da Computação

Agrupamento de Dados a partir de Mapas Auto-Organizáveis na Ferramenta YADMT

Thiago Magalhães Faino

CASCAVEL

2013

THIAGO MAGALHÃES FAINO

AGRUPAMENTO DE DADOS A PARTIR DE MAPAS AUTO-ORGANIZÁVEIS NA FERRAMENTA YADMT

Monografia apresentada como requisito parcial para obtenção do grau de Bacharel em Ciência da Computação, do Centro de Ciências Exatas e Tecnológicas da Universidade Estadual do Oeste do Paraná - Campus de Cascavel.

Orientadora: Prof.ª Drª Rosangela Villwock

Co-Orientador: Prof. Dr. Clodis Boscarioli

CASCAVEL

2013

THIAGO MAGALHÃES FAINO

AGRUPAMENTO DE DADOS A PARTIR DE MAPAS AUTO-ORGANIZÁVEIS NA FERRAMENTA YADMT

Monografia apresentada como requisito parcial para obtenção do Título de *Bacharel em Ciência da Computação*, pela Universidade Estadual do Oeste do Paraná, Campus de Cascavel, aprovada pela Comissão formada pelos professores:

Prof.ª Drª Rosangela Villwock (Orientadora)
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Dr. Clodis Boscarioli (Co-Orientador)
Colegiado de Ciência da Computação,
UNIOESTE

Prof. MSc. Carlos José Maria Olguin
Colegiado de Ciência da Computação,
UNIOESTE

Prof. Dr. Jerry Adriani Johann
Centro de Ciências Exatas e Tecnológicas,
UNIOESTE

Cascavel, 2013.

DEDICATÓRIA

Dedico este trabalho aos meus pais, Gilberto da Silva Faino e Veroni Magalhães Faino, que sempre me apoiaram em todas as minhas decisões. Ao meu irmão Gustavo Magalhães Faino por sempre estar por perto para me incentivar. Aos meus amigos e familiares que sempre me deram o total apoio e força pra que eu conseguisse concluir esta difícil etapa na minha vida. E por fim, à minha avó, que aonde quer que esteja, deve estar feliz e orgulhosa por mim agora, Rosa Maria Magalhães (in memoriam).

“Existem três jeitos de fazer as coisas: o jeito certo, o jeito errado, e o meu jeito, que é igual ao jeito errado, só que mais rápido”.

Homer Simpson

AGRADECIMENTOS

Primeiramente agradeço a Deus pela saúde, pelas conquistas e pela força que me deu para superar os momentos difíceis. Agradeço aos meus pais, Gilberto da Silva Faino e Veroni Magalhães Faino, por me proporcionar tudo de melhor, e que sempre me apoiaram e ajudaram em todas as minhas decisões. Agradeço também ao meu irmão Gustavo Magalhães Faino e aos meus familiares que sempre acreditaram em mim e me deram forças para seguir até o final dessa dura caminhada.

Agradeço a todos os meus amigos e companheiros de curso, em especial: Argentino, Astério Junior, Fernando Fernandes, Gustavo Catarino, Gustavo H. Paetzold, Igor S. Lopes, Leandro Maia, Leonardo R. Nardelli, Lucas Szeremeta, Mateus F. Teixeira e Wilson C. Neto, que tenho sorte em ter conhecido e que quero levar essa amizade para a vida toda, são mais que amigos são meus verdadeiros irmãos.

Agradeço de forma especial a minha orientadora Rosangela Villwock, que me orientou em projetos de iniciação científica desde o meu primeiro ano e também neste trabalho de conclusão de curso. Também agradeço ao professor Clodis Boscarioli, que me ajudou muito sendo meu co-orientador neste trabalho e meu orientador no estágio supervisionado. Com eles aprendi muito, e tenho muito a agradecer.

Lista de Figuras

Lista de Quadros

Lista de Abreviatura e Siglas

BMU	*Best Matching Unit* (Neuronal Vencedor)
KDD	*Knowledge Discovery in Databases* (Descoberta de Conhecimento em Bases de Dados)
Matriz-U	Matriz de Distâncias Unificadas
ML	Matriz de Ligações
RNA	Rede Neural Artificial
RNAs	Redes Neurais Aritificiais
SGBD	Sistemas Gerenciadores de Bancos de Dados
SL-*SOM*	*Self-Labeled SOM*
SOM	*Self Organizing Map* (Mapas Auto-organizáveis)
YADMT	*Yet Another Data Mining Tool*

Lista de Símbolos

v_n	Vetor de dados
w	Vetor de pesos do neurônio
w_{bmu}	Vetor de pesos do neurônio vencedor
d	Função de distância
t	Variável de tempo
$\eta(t)$	Função de aprendizagem no tempo t
$\sigma(t)$	Raio de vizinhança topológica no tempo t
h	Função de vizinhança topológica
E_q	Erro de quantização
E_t	Erro topológico

Sumário

Resumo

Com o grande crescimento do volume de dados armazenados, muitas informações e conhecimento podem estar sendo perdidos devido às limitações humanas em analisar e interpretar esta grande quantidade de dados. Desse modo, surgem ferramentas e técnicas para o auxílio na extração de conhecimento dentro de um processo chamado de Descoberta de Conhecimento em Banco de Dados (*Knowledge Discovery in Databases* - KDD). Dentro desse processo técnicas baseadas em Redes Neurais Artificiais (RNAs) vem se destacando quando utilizadas para a tarefa de agrupamento de dados. É neste contexto que este trabalho apresenta uma metodologia de agrupamento de dados a partir de Mapas Auto-Organizáveis ("*Self Organizing Map*" - *SOM*), que foi implementado em um módulo da ferramenta de Mineração de Dados YADMT, que está sendo desenvolvida na UNIOESTE. A metodologia de agrupamento de dados a partir do *SOM* consiste em métodos de visualização e de agrupamentos. Como métodos de visualização foram implementados a Matriz-U e a Matriz de Densidade e também um método pra a visualização em tempo real o aprendizado do mapa *SOM.* Para agrupamento a partir do *SOM* foram implementados ao todo cinco metodologias: agrupamento por Matriz de Densidade, SL-*SOM*, 1D-*SOM*, Metodologia de Vesanto e Alhoniemi (2000) e também foi proposto um método de agrupamento utilizando um algoritmo de extração de componentes conectados. Os métodos de agrupamento implementados foram comparados a outros métodos de agrupamento presentes na ferramenta YADMT, utilizando como métrica a Medida F, o Índice aleatório R e a porcentagem de agrupamento correto. Os resultados obtidos foram satisfatórios, pois na maioria dos experimentos o *SOM* obteve o melhor resultado comparado aos outros métodos de agrupamento. Ainda a metodologia implementada foi aplicada a uma base de dados real e inédita criada a partir dos dados fornecidos pelo IBGE das cidades do Paraná, validando as metodologias implementadas.

Palavras-Chave: Mineração de Dados, Redes Neurais Artificiais, Metodologias de Agrupamento a partir do *SOM*.

Capítulo 1

Introdução

O volume de dados armazenados e manipulados pela maioria das organizações cresce diariamente a uma taxa que ultrapassa a capacidade humana de analisar, sintetizar e extrair conhecimento a partir desses dados. Muitas vezes, esse grande volume de dados contém informações úteis, chamado de "conhecimento", que não está facilmente disponível ou identificado. Analistas humanos podem gastar semanas para descobrir este conhecimento e, por este motivo, alguns bancos de dados grandes nunca recebem uma análise detalhada adequada, o que torna o uso de ferramentas que automatizam o processo de análise de grandes quantidades de dado imprescindível.

Este contexto justifica a existência de uma área de investigação, o *KDD* (*Knowledge Discovery in Databases* ou Descoberta de Conhecimento em Bases de Dados), cujo principal objetivo é extrair o conhecimento a partir de informações "escondidas" nos dados que sejam úteis nas tomadas de decisões, utilizando métodos, algoritmos e técnicas de diferentes áreas científicas, que segundo Tan, Steinbach e Kumar (2005), incluem Estatística, Inteligência Artificial, Aprendizagem de Máquinas e Reconhecimento de Padrões.

De acordo com Fayyad *et al.* (1996), essa área é genericamente definida como "o processo não trivial de identificação de padrões válidos e potencialmente úteis, perceptíveis a partir dos dados". O *KDD* é um processo de várias fases e contêm uma série de passos que auxiliam nas mais diversas decisões a serem tomadas. Cada fase possui uma ligação com as demais, melhorando assim a cada resultado. As fases do KDD são: seleção, pré-processamento, formatação, mineração de dados e interpretação e avaliação do conhecimento, que tem como objetivo a descoberta de padrões válidos, novos, úteis e acessíveis. O núcleo deste processo é a etapa de Mineração de Dados, onde são aplicados os algoritmos para extrair padrões dos dados. A Figura 1.1 ilustra as cinco fases do processo *KDD,* explicadas brevemente a seguir.

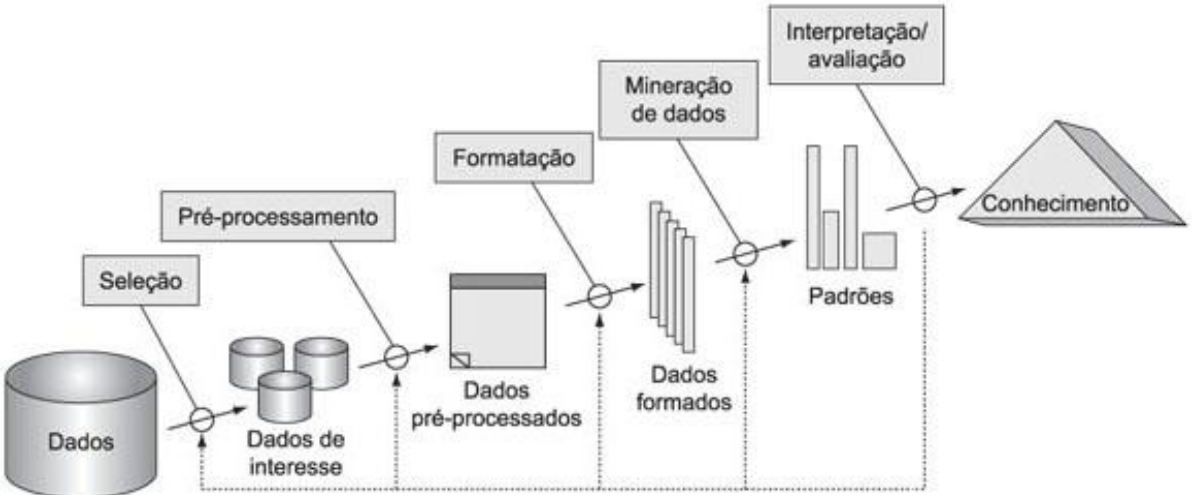


Figura 1.1: Etapas do processo KDD

Fonte: (FAYYAD *et al.* 1996)

- Seleção: Após entender o domínio da aplicação e definir os objetivos a serem atingidos, iniciam-se as fases do *KDD*, começando pela seleção e coletados dados necessários ao trabalho.
- Pré-Processamento: Esta fase determina a qualidade dos dados, verificam-se os dados redundantes, faltantes ou inconsistentes. Também é verificada a possibilidade de diminuir o número de variáveis.
- Formatação: Após serem selecionados e pré-processados os dados necessitam ser armazenados e formatados adequadamente. Neste processo há uma transformação dos dados brutos para um formato apropriado para a aplicação da Mineração de Dados.
- Mineração de Dados: Esta fase é o núcleo do processo *KDD*, onde são aplicados os algoritmos para extrair padrões dos dados. Primeiramente, deve-se escolher o método a ser utilizado.
- Interpretação e Avaliação dos resultados: Nesta fase deve-se interpretar o conhecimento obtido e analisar a eficácia do método aplicado na etapa de Mineração de Dados. Caso o conhecimento não seja válido, o processo deverá ser reiniciado, analisando todas as etapas em busca de melhorar o que for necessário.

Segundo Fayyad *et al.* (1996), o processo *KDD* possui várias formas de interpretação de dados que são denominadas de tarefas. As tarefas de Mineração de Dados podem ser preditivas, que usam variáveis para prever valores futuros ou desconhecidos, ou podem ser

descritivas, que utilizam padrões para descrever os dados. As principais tarefas preditivas são a Predição, que é usada para definir um provável valor para uma ou mais variáveis; e a Classificação, que constrói um modelo que possa ser aplicado a dados não classificados visando categorizar os objetos em classes. Já as principais tarefas descritivas incluem a Regras de Associação, que determina quais fatos ou objetos tendem a ocorrerem juntos num mesmo evento; e o Agrupamento de Dados, foco deste estudo, que visa dividir uma população em subgrupos o mais heterogêneos possível entre si.

1.1. Motivação

Com a grande evolução tecnológica e o aumento de dados produzidos, a análise de dados automatizados vem se tornando uma necessidade para tomada de decisão, principalmente pela dificuldade enfrentada na extração manual de informações e conhecimento a partir de grande quantidade de dados. Desse modo, surgem ferramentas e técnicas para o auxílio na extração de conhecimento dentro de um processo *KDD.* Dentro desse processo técnicas baseadas em Redes Neurais Artificiais (RNAs) vem se destacando quando utilizadas para a tarefa de agrupamento de dados.

Em RNAs, o procedimento para solução de problemas passa inicialmente por uma fase de aprendizagem (treinamento), em que um conjunto de exemplos é apresentado para a Rede Neural, a qual extrai de forma automática as características necessárias para representar a informação fornecida. Os Mapas Auto-Organizáveis (ou *Self Organizing Map - SOM*), também conhecidos como Redes Neurais Artificiais de Kohonen (KOHONEN, 1989), têm sido usados largamente como uma ferramenta de visualização de dados apresentados em dimensões elevadas. O *SOM* define, via treinamento não supervisionado, um mapeamento de um espaço contínuo para um conjunto discreto de neurônios, geralmente dispostos na forma de uma grade. O objetivo principal do treinamento é reduzir dimensionalidade ao mesmo tempo em que se tenta preservar, ao máximo, a topologia do espaço de entrada.

As principais aplicações do *SOM* envolvem a visualização de dados complexos e a criação de abstrações para agrupamento de dados. Para a tarefa de agrupamento de dados o *SOM* normalmente é utilizado apenas como uma das etapas para a realização da tarefa, que deve ser seguida da aplicação de técnicas e metodologias de agrupamentos, como o algoritmo SL-

SOM, a metodologia de Vesanto e Alhoniemi (2000), a metodologia de Boscarioli (2008), entre outras.

1.2. Objetivos

Estudar e implementar uma metodologia de agrupamento de dados a partir de Mapas Auto-Organizáveis no módulo de Análise de Agrupamento da ferramenta YADMT– *Yet Another Data Mining Tool*, uma ferramenta de *KDD* em desenvolvimento na Unioeste no campus de Cascavel. Este objetivo divide-se nos seguintes objetivos específicos:

- Implementar o *SOM* permitindo a utilização de diferentes grades (1D-*SOM* e 2D-*SOM*), arranjos de vizinhança de grade retangular e hexagonal além da utilização de diferentes parametrizações;
- Implementar métodos para a visualização dos resultados obtidos com a metodologia desenvolvida (Matriz-U e Matriz de Densidade);
- Implementar metodologias para realizar o agrupamento de um mapa *SOM* treinado;
- Realizar experimentos para validação da implementação realizada, utilizando como métricas os índices externos: medida F e índice aleatório R, além da porcentagem de classificação correta;
- Aplicação a uma base de dados real e inédita.

1.3. Organização do Documento

Além do Capítulo 1 que é introdutório ao contexto do trabalho, o trabalho está organizado da seguinte forma:

No Capítulo 2 é apresentada uma visão geral da Análise de Agrupamentos, bem como as principais categorias e algoritmos de agrupamento existentes.

O Capítulo 3 apresenta-se uma breve descrição sobre as Redes Neurais Artificiais, seguido dos principais conceitos e características dos Mapas Auto-Organizáveis, com questões sobre métricas utilizadas e as parametrizações, além dos métodos de visualização e metodologias de agrupamento mais utilizadas.

O Capítulo 4 apresenta a metodologia adotada e detalhes de implementações realizadas na ferramenta YADMT.

No Capítulo 5 são apresentados os resultados dos experimentos realizados para a validação da metodologia desenvolvida em comparação a outros métodos de agrupamento.

O Capítulo 6 apresenta as conclusões finais, com a discussão geral dos resultados e da pesquisa, além de relato das dificuldades encontradas e trabalhos futuros.

Capítulo 2

Análise de Agrupamentos

A tarefa de agrupar objetos semelhantes é um processo usualmente adotado pelo ser humano ao longo da história da humanidade, podendo ser associado inclusive à própria criação da linguagem. As palavras podem ser interpretadas como rótulos associados a conjuntos de objetos semelhantes. Tomando os adjetivos como exemplo, eles são rótulos que permitem classificar e discriminar agentes e objetos do meio (ZUCHINI, 2003).

A tarefa de Agrupamento procura grupos de padrões tal que padrões pertencentes ao mesmo grupo são mais similares, e são dissimilares a padrões pertencentes a outros grupos. Segundo Hair *et al.* (2005), a análise de agrupamentos é a denominação para um grupo de técnicas analíticas para desenvolver subgrupos significativos de objetos.

O agrupamento é feito com base numa medida de similaridade ou dissimilaridade. A medida de similaridade avalia se os objetos são similares, ou seja, quanto maior o valor da medida mais parecidos são os objetos. A medida de dissimilaridade avalia se os objetos são dissimilares, ou seja, quanto maior o valor da medida menos parecidos serão os objetos (HAIR et al, 2005).

Segundo Zuchini (2003), pode-se entender que os algoritmos de agrupamento são métodos que buscam dividir um conjunto de objetos não rotulados em grupos, de forma que os objetos de cada grupo tenham mais semelhanças entre si do que em relação aos objetos de qualquer outro grupo.

2.1. Algoritmos de Agrupamento

Os algoritmos de agrupamento podem ser divididos em categorias de diversas formas de acordo com as suas características. Entre as classes de algoritmos de agrupamento as duas

principais são os métodos hierárquicos e os métodos de particionamento. A Figura 2.1 mostra uma hierarquia simples presente na literatura sobre as categorias de algoritmos de agrupamento (ZUCHINI, 2003).

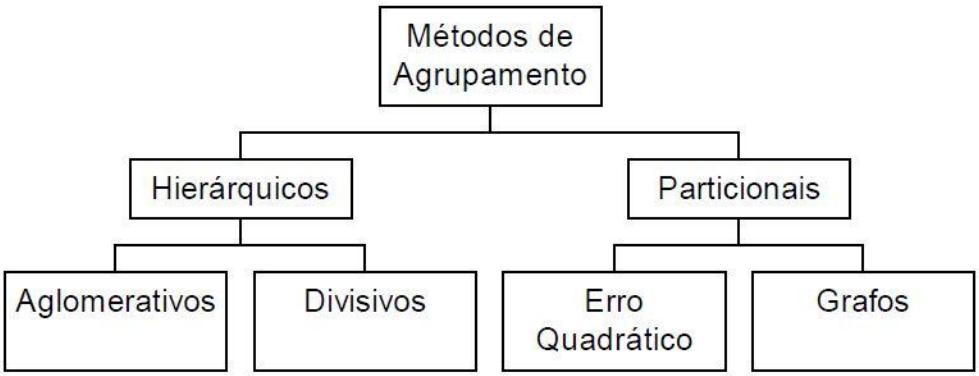


Figura 2.1: Classificação simplificada dos métodos de agrupamento

Fonte: (ZUCHINI, 2003)

2.1.1. Métodos de Particionamento

Métodos de Particionamento ou Não-Hierárquicos procuram uma partição sem a necessidade de associações hierárquicas. Seleciona-se uma partição dos elementos em *K* grupos, otimizando algum critério (DINIZ; NETO, 2000).

Segundo Zuchini (2003), os métodos de particionamento dividem o conjunto contendo *N* objetos em *K* grupos, sem relacioná-los entre si como é feito nos métodos hierárquicos. Os métodos de particionamento são vantajosos em aplicações que envolvem grandes bases de dados. Por outro lado os métodos de particionamento, em geral, assumem que o número de grupos do conjunto de dados é conhecido, dessa forma a escolha correta do número *K* é extremamente importante para o seu bom desempenho, essa escolha, na prática, é empírica.

O método mais conhecido entre os métodos de particionamento é o do *K*-Médias (MACQUEEN, 1967). Normalmente, os *K* grupos encontrados por ele são de melhor qualidade do que os *K* grupos produzidos pelos métodos hierárquicos (JOHNSON; WICHERN, 1998).

O algoritmo *K*-Médias recebe como entrada o número *K* de grupos que se deseja formar. Logo após são definidos os valores ou os objetos para serem centroides iniciais de cada grupo. Esta escolha pode ser feita de forma aleatória, por diversas heurísticas ou ainda de forma manual, sendo que os resultados diferem de acordo com a forma escolhida. Sucessivamente,

cada objeto é associado ao grupo mais próximo e o centroide de cada grupo é recalculado levando-se em conta o novo grupo formado. O critério de parada do algoritmo pode ser um erro estipulado ou quando houver poucas trocas de objetos entre grupos (JAIN; MURTY; FLYNN, 1999).

2.1.2. Métodos Hierárquicos

Os métodos hierárquicos englobam técnicas que buscam de forma hierárquica os grupos e, por isso, admitem obter vários níveis de agrupamento. Os métodos hierárquicos podem ser subdivididos em divisivos ou aglomerativos. O método hierárquico aglomerativo considera cada padrão como um grupo distinto, e ao decorrer do algoritmo agrupa o par de grupos com maior similaridade em um novo grupo até que um critério de parada seja satisfeito. O método hierárquico divisivo faz exatamente o contrário, inicia todos os padrões como um único grupo e executa um processo de sucessivas subdivisões, até que o critério de parada seja satisfeito (DINIZ; NETO, 2000).Os métodos de agrupamento hierárquicos mais conhecidos na literatura são: Ligação Simples, Ligação Completa, Ligação Média e Método Ward.

Na Ligação Simples (*Single Linkage* ou vizinho mais próximo), a distância entre dois grupos é a mínima das distâncias entre todos os pares de padrões *i* e *j*, *i* pertencente ao primeiro grupo e *j* ao segundo (JAIN; MURTY; FLYNN, 1999). Por exemplo, se um Grupo 1 é formado pelos padrões *U* e *V* e um Grupo 2 é formado pelo padrão *W*, a distância entre esses grupos é calculada pela Equação 2.1 (JOHNSON; WICHERN, 2007).

$$d_{(1,2)} = \min\{d_{UW}, d_{VW}\} \tag{2.1}$$

Na Ligação Completa (*Complete Linkage* ou vizinho mais distante), a distância entre dois grupos é a máxima das distâncias entre todos os pares de padrões *i* e *j*, *i* pertencente ao primeiro grupo e *j* ao segundo (JAIN; MURTY; FLYNN, 1999). Por exemplo, se um Grupo 1 é formado pelos padrões *U* e *V* e um Grupo 2 é formado pelo padrão *W*, a distância entre esses grupos é calculada pela Equação 2.2 (JOHNSON; WICHERN, 2007).

$$d_{(1,2)} = \max\{d_{UW}, d_{VW}\} \tag{2.2}$$

Já na Ligação Média (*Average Linkage*), a distância entre dois grupos é a média das distâncias entre todos os pares de padrões, sendo que cada padrão do par é de um grupo. Se

um Grupo 1 é formado pelos elementos *U* e *V* e um Grupo 2 é formado pelo elemento *W*, a distância entre os grupos 1 e 2 é calculada pela Equação 2.3 onde d_{ik} é a distância entre o padrão *i* no Grupo 1 e o padrão *k* no Grupo 2, N_1 é o número de padrões no Grupo 1 e N_2 é o número de padrões no Grupo 2 (JOHNSON; WICHERN, 2007).

$$d_{(1,2)} = \frac{\sum\sum d_{ik}}{N_1 * N_2} \tag{2.3}$$

Ainda segundo Johnson e Wichern (2007), o Método de Ward faz a junção de dois grupos baseando-se na "perda de informação". Considera-se como critério de "perda de informação" a soma do Quadrado do Erro (*SQE*). Para cada Grupo *i*, calcula-se a média (ou centróide) do grupo e a soma do Quadrado do Erro do Grupo *i* (SQE_i) que é a soma do quadrado do erro de cada padrão do grupo em relação à média. Segundo Hair Jr *et al.* (2005), este método tende a obter grupos de mesmo tamanho devido à minimização de sua variação interna, por apresentar esta característica, bases de dados com grupos de mesmo tamanho obtém os melhores resultados.

Segundo Jain, Murty e Flynn (1999), a forma mais comum de representar um agrupamento hierárquico é utilizando dendrogramas. Os dendrogramas representam o agrupamento dos padrões e os níveis de similaridade em que os grupos se formam. O dendrograma pode ser dividido em diferentes níveis, onde cada nível pode mostrar diferentes grupos. A Figura 2.2 ilustra o dendrograma obtido utilizando o método hierárquico ligação simples, pode-se observar a divisão feita pela linha pontilhada no dendrograma, formando assim, três grupos. O primeiro grupo é composto pelos padrões *A*, *B* e *C*, o segundo é composto pelos padrões *D* e *E*, e por fim o terceiro grupo formado pelos padrões *F* e *G*.

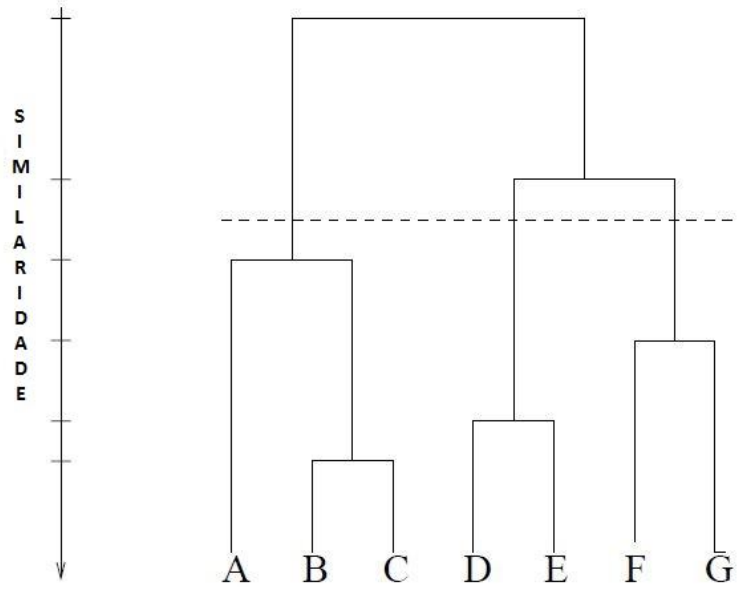


Figura 2.2: Dendrograma obtido utilizando a ligação simples

Fonte: Adaptado de (JAIN; MURTY; FLYNN, 1999)

2.1.3. Avaliação do Agrupamento

Os algoritmos de agrupamento sempre vão fazer a divisão do conjunto de dados sem se preocupar se este agrupamento existe realmente no conjunto. Sendo assim é muito importante saber se esses agrupamentos formados pelo algoritmo estavam realmente ocultos aos dados, ou se não possuem significado. Para aumentar o grau de confiabilidade nos resultados gerados pelo algoritmo, faz-se necessária uma avaliação por parte dos especialistas de domínio (BOSCARIOLI, 2008).

Para a avaliação dos agrupamentos, alguns aspectos podem ser observados: comparação dos resultados de uma análise de grupos com resultados previamente conhecidos; avaliações de quanto os resultados de uma análise de grupos se ajustam aos dados sem resultados previamente conhecidos; comparação dos resultados de dois conjuntos de grupos para determinar qual deles é melhor ou, ainda, determinação do número “correto” de grupos (TAN; STEINBACH; KUMAR, 2005).

Segundo Tan, Steinbach e Kumar (2005), as medidas numéricas aplicadas para julgar vários aspectos de avaliação de grupos são classificadas em três tipos:

- Índices Externos (supervisionados): São usados para medir até que ponto rótulos de grupos corresponde a rótulos de classes fornecidos a partir de algum conhecimento *a priori*.
- Índices Internos (não supervisionados): São usados para medir quão boa é a estrutura de agrupamento sem depender de conhecimento prévio.
- Índices Relativos: São usados para comparar dois grupos ou agrupamentos diferentes, e dar a referência de qual deles melhor apresenta as características reais dos objetos.

Os índices de avaliação do agrupamento apesar de serem bastante variáveis, conseguem fornecer mais um instrumento de apoio à confiabilidade dos agrupamentos gerados.

2.2. Considerações Finais

Este capítulo apresentou a tarefa de agrupamento de dados de forma rápida e introdutória, com vistas à melhor compreensão do leitor no restante do trabalho.

A Análise de Agrupamentos têm grande importância dentro das tarefas disponíveis no processo *KDD*. Devido à complexidade e à escalabilidade das bases de dados, essas técnicas são eficientes, trazendo redução da complexidade, para uma melhor interpretação em processos decisórios.

As técnicas de agrupamento têm a capacidade de extrair informações sobre estruturas de dados relativamente complexas e volumosas, as quais eram previamente desconhecidas e difíceis de serem observadas sem uma redução de sua complexidade. Embora técnicas de classificação e de agrupamento tenham um resultado final similar, com a divisão de diferentes elementos em classes, ou agrupamentos, as técnicas de agrupamento são mais poderosas e complexas, uma vez que as categorias, ou agrupamentos, não são previamente conhecidos.

Capítulo 3

Mapas Auto-Organizáveis

As Redes Neurais Artificiais (RNAs) são sistemas computacionais que visam adaptar certas atividades do cérebro humano para os computadores. Elas foram inspiradas nas redes neurais biológicas onde o conhecimento é obtido por algum processo de aprendizagem.

O cérebro humano é altamente complexo, não linear e paralelo, e tem a capacidade de organizar sua estrutura (neurônios) de forma que consiga realizar certos processamentos com muito mais rapidez que qualquer computador existente (HAYKIN, 2001).

As RNAs são algoritmos que podem ser programados em diversas linguagens. São modelos matemáticos com diversos parâmetros ajustáveis, os quais são modificados de acordo com um conjunto de dados que contém informações sobre o comportamento que esta rede deve apresentar (FAUSET, 1994).

Segundo Haykin (2001) a principal propriedade de uma rede neural é a sua habilidade de aprender e de melhorar o seu desempenho através da aprendizagem à medida que os pesos sinápticos (ou forças de conexão entre neurônios) são ajustados, de forma ordenada, para alcançar um objetivo desejado.

O processo de aprendizagem de uma RNA pode ocorrer de forma supervisionada ou não supervisionada (auto organizada). No processo de aprendizagem supervisionada a rede neural possui o auxílio de um "professor" que tem um conhecimento prévio do vetor de entrada exposto a ela. Dessa forma o professor é capaz de fornecer uma resposta desejada sobre aquele vetor de entrada (HAYKIN, 2001).

Já no processo de aprendizagem auto organizada, não existe um conhecimento prévio sobre o vetor de entrada, ou seja, não há um professor externo para supervisionar o processo de aprendizado da rede (HAYKIN, 2001). Segundo Costa (1999), a auto-organização é um

processo coletivo, onde unidades que fazem parte deste processo competem, com chances de sucesso semelhantes, por recursos limitados, onde o processo é parcialmente autônomo em relação às suas condições iniciais. Ainda, segundo Costa (1999), a definição do que seja um processo auto organizado pode ser muito complexa e ainda é uma questão em aberto.

Mapa auto-organizável ("*Self Organizing Map*" - *SOM*), também conhecido por Rede Neural Artificial de Kohonen, é um modelo de rede neural desenvolvido por Teuvo Kohonen (KOHONEN, 1989). Pode ser considerada uma rede neural com aprendizado não supervisionado e competitivo, com a habilidade de realizar mapeamentos que preservam a topologia entre os espaços de entrada e de saída. Esta propriedade é observada no cérebro, mas não é encontrado em outras redes neurais artificiais. O objetivo do *SOM* é transformar padrões de entrada de dimensão arbitrária eu um mapa discreto.

Segundo Fauset (1994), os mapas auto-organizáveis fazem parte de um grupo de redes neurais artificiais baseadas em *modelos de competição*, ou simplesmente *redes competitivas*, pois estas redes combinam competição com uma forma de aprendizagem para fazer os ajustes de seus pesos.

Segundo Francisco (2004), o desenvolvimento de *SOM* como modelo neural foi inspirado por uma característica presente no cérebro humano, que é a organização das informações em muitas regiões, de modo que as entradas sensoriais distintas são representadas por mapas computacionais topologicamente ordenados.

O processo de aprendizagem é baseado no aprendizado competitivo, o algoritmo tem natureza local e as modificações dos pesos sinápticos são confinadas à vizinhança do neurônio ativado. Os neurônios de saída competem entre si para serem ativados, de forma que apenas um neurônio de saída seja considerado "vencedor" (FRANCISCO, 2004).

A rede *SOM* é baseada em uma grade de neurônios, que normalmente é de uma ou duas dimensões. Mapas de dimensões maiores são também possíveis, porém de mais difícil aplicação e compreensão. Uma grade de neurônios bidimensional pode apresentar topologia retangular ou hexagonal (Figura 3.1). Para cada topologia existe uma forma de ligação entre os neurônios, que define o tipo de vizinhança, que para a topologia retangular será de oito vizinhos e para a hexagonal será seis vizinhos. Ainda, como pode ser visto na Figura 3.1, são definidos níveis ou raios de vizinhança (FAUSETT, 1994).

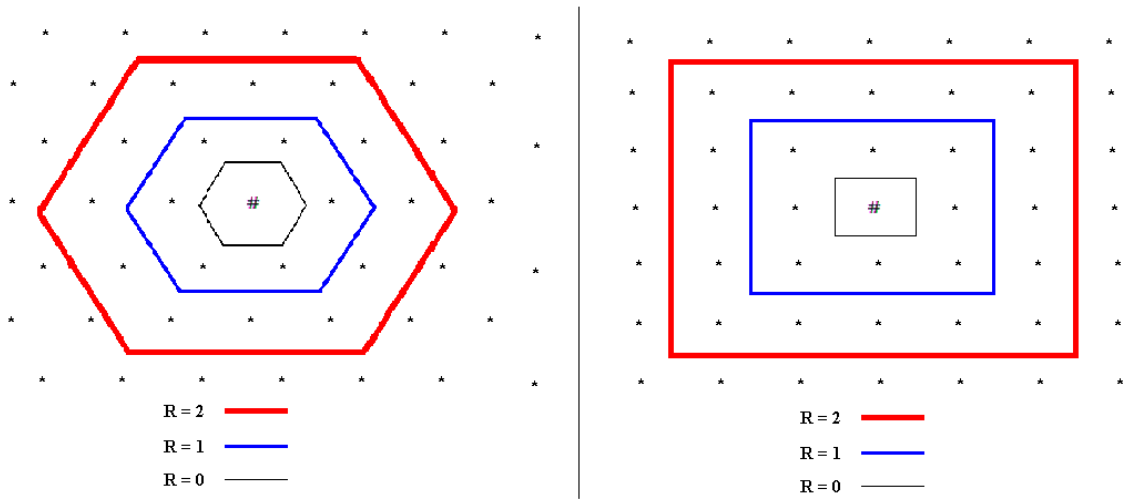


Figura 3.1: Grade bidimensional hexagonal e retangular com raios de vizinhança (R) iguais a zero, um e dois

Fonte: (FAUSETT, 1994)

A arquitetura de um *SOM* é formada por duas camadas: camada de entrada *i* e a camada de saída *j*, conforme ilustra a Figura 3.2. A camada de entrada do *SOM* é descrita por (KOHONEN, 2001) como um vetor.

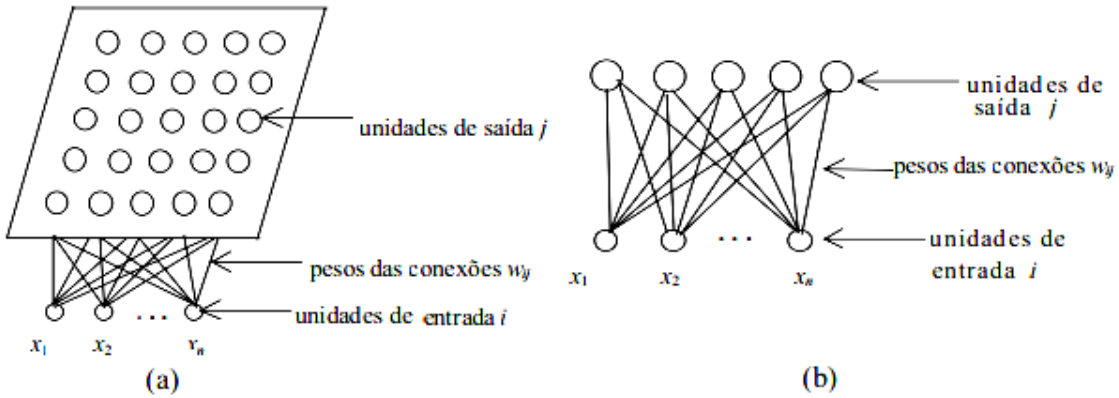


Figura 3.2: Arquiteturas típicas de um *SOM*: (a) arquitetura bidimensional, (b) arquitetura unidimensional

Fonte: (CASTRO, 1999)

Cada neurônio da camada de saída é completamente conectado com todos os padrões do vetor de entrada. São apresentados os vetores de entrada à camada de saída e a cada padrão do vetor de entrada apresentado tem-se uma região de atividade na grade. A localização e natureza de uma determinada região variam de um padrão de entrada para outro. Assim sendo, todos os neurônios da rede devem ser expostos a um número suficiente de diferentes padrões de entrada, garantindo assim que o processo de auto-organização ocorra de forma

apropriada (FRANCISCO, 2004).Como pode ser observado na Figura 3.2, para cada conexão entre uma unidade da camada de entrada *i* com uma unidade da camada de saída *j*, existe um peso para a conexão ou peso sináptico w_{ij} (CASTRO, 1999).

Uma vez que concluído o algoritmo *SOM*, o mapa de saída gerado, representado pelos pesos sinápticos w_{ij}, mostrará características importantes dos vetores de entrada. De acordo com Haykin (2001) e Kohonen (2001), o mapa de características gerado pelo *SOM* conta com as seguintes propriedades:

- Ordenação topológica: O mapa de características é ordenado topologicamente, no sentido de que a localização espacial de um neurônio na grade corresponde a um domínio particular ou características dos vetores de entrada.
- Casamento de densidade: O mapa de características reflete variações na estatística da distribuição da entrada, embora a distribuição das unidades do *SOM* não seja exatamente a mesma da distribuição dos dados amostrais. Regiões no espaço de entrada onde os vetores de dados são presentes com uma alta probabilidade de ocorrência são mapeadas para domínios maiores do mapa de saída e, portanto, com melhor resolução que regiões com baixa probabilidade de ocorrência.
- Seleção de características: Pode-se afirmar que os Mapas Auto-Organizáveis fornecem uma aproximação discreta das assim chamadas curvas principais,e podem, portanto, ser vistos como uma generalização não-linear da análise de componentes principais.

3.1. Processos de um *SOM*

Conforme Haykin (2001) o algoritmo responsável pela formação do mapa auto-organizável pode começar iniciando os pesos sinápticos da grade com valores pequenos e aleatórios, dessa forma nenhuma organização prévia é adicionada ao mapa de características. Segundo Kohonen (2001), a inicialização dos pesos sinápticos pode ser feita de forma linear, que impõe uma ordem na criação do mapa, iniciando-o com uma forma mais organizada, e que se estabiliza em muito menos iterações que o método aleatório. Após a iniciação do mapa há três processos envolvidos na formação de um mapa auto-organizável: processo competitivo, cooperativo e adaptativo.

No processo competitivo, cada padrão do vetor de entrada é apresentado a todos os neurônios da camada de saída, que competem entre si e, de acordo com algum critério, é definido um *BMU* (*Best Matching Unit*), ou neurônio vencedor. Geralmente o critério adotado para a escolha do neurônio vencedor é a distância mínima entre o padrão e o neurônio, a distância Euclidiana é a mais utilizada. Outra forma de avaliar a semelhança entre o padrão de entrada e o neurônio vencedor, que segundo Fauset (1994), é o produto interno dos vetores de entrada e peso normalizados, que pode ser interpretado como a correlação entre o vetor de entrada e o vetor peso.

No processo Cooperativo o neurônio vencedor determina a localização espacial de uma vizinhança de neurônios excitados. A definição de vizinhança é baseada na evidência neurobiológica de que há interação lateral entre um conjunto de neurônios biológicos excitados (FRANCISCO, 2004).

Seja uma vizinhança h_{ij} centrada por um neurônio vencedor i cercado por um conjunto de neurônios excitados cooperativos, dos quais um neurônio é denotado por j. A distância lateral entre o neurônio vencedor i e o neurônio j é denotada por d_{ij}. Assume-se então que a vizinhança topológica h_{ij} é uma função unimodal, ou seja, que apresenta um único mínimo ou máximo, da distância lateral d_{ij}, tal que satisfaça duas exigências distintas (CASTRO, CASTRO, 2001):

1. A vizinhança h_{ij} é simétrica ao redor do seu ponto máximo definido por $d_{ij} = 0$, ou seja, atinge o valor máximo somente no neurônio vencedor.
2. A amplitude da vizinhança h_{ij} decresce monotonicamente com o aumento da distância lateral d_{ij}.

Abaixo, as Equações 3.1, 3.2, 3.3 e 3.4, apresentam algumas funções de vizinhança utilizadas na literatura que satisfazem as exigências para a vizinhança topológica h_{ij}, nas quais o raio de vizinhança topológica na iteração t é denominado por σ_t. A função mais utilizada para esse processo é a função gaussiana (Equação 3.2), a qual foi utilizada neste trabalho. Na Figura 3.3 as funções são apresentadas graficamente (VESANTO, 2000).

$$h_{ij}(t) = (\sigma_t - d_{ij}) \tag{3.1}$$

$$\mathrm{h_{ij}}\,(t) = e^{\frac{-d_{ji}^2}{2\sigma_t^2}} \tag{3.2}$$

$$h_{ij}\,(t) = e^{\frac{-d_{ji}^2}{2\sigma_t^2}}\,(\,\sigma_t - d_{ij}\,) \tag{3.3}$$

$$h_{ij}\,(t) = \max\{0, 1 - (\,\sigma_t - d_{ij}\,)^2\,\} \tag{3.4}$$

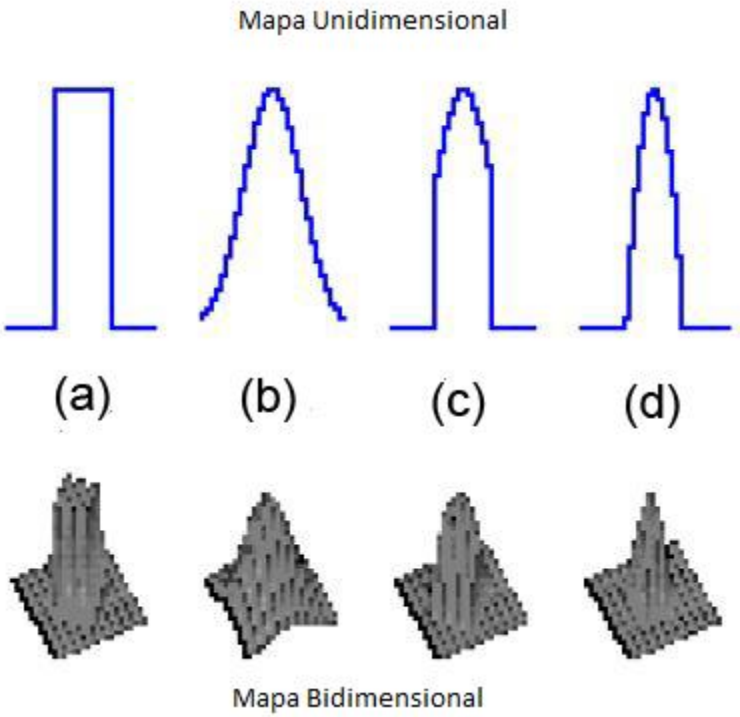


Figura 3.3**:** Funções de vizinhança – (a) Equação 3.1 (b) Equação 3.2 (c) Equação 3.3 (d) Equação 3.4
Fonte: (VESANTO, 2000).

Na fase adaptativa, os neurônios excitados aumentam seus valores individuais em relação ao padrão de entrada por meio de uma atualização em seus pesos sinápticos, conforme a Equação 3.5. Nesta atualização leva-se em consideração a distância do vizinho até o neurônio vencedor e a atualização é mais intensa nos vizinhos mais próximos (KOHONEN, 2001).

$$w_j\,(t+1) = w_j\,(t) + \eta\,(\mathrm{t})\mathrm{h_{j,i(x)}}(\mathrm{t})\left(\mathrm{x} - w_j\,(t)\right) \tag{3.5}$$

Com as atualizações realizadas nos neurônios excitados, observa-se na Figura 3.4, que o neurônio que venceu a competição (*BMU*) "arrasta" seus vizinhos na direção do padrão de entrada v_k numa proporção que depende da função de vizinhança h_{ij} . Dessa forma, os neurônios tendem a se aproximar ao padrão de entrada, promovendo a ordenação topológica da rede em relação aos dados de entrada (ZUCHINI, 2003).

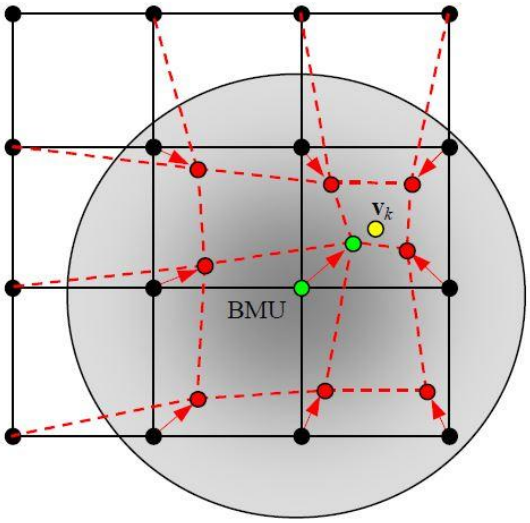


Figura 3.4: Atualização do neurônio vencedor e de seus vizinhos

Fonte: (ZUCHINI, 2003)

Ao final da fase adaptativa a taxa de aprendizagem e o raio de vizinhança devem ser atualizados. A taxa de aprendizagem $\eta(t)$ deve variar com o tempo, como indicado na Equação 3.5. Deve-se iniciar em um valor η_0 e decrescer gradualmente conforme o tempo *t*. Para satisfazer essa condição pode ser utilizada a Equação 3.6, 3.7 ou 3.8 a seguir extraída de (VESANTO, 2000), que são apresentadas graficamente na Figura 3.5.

$$\text{Linear:}\eta(t) = \eta_0\left(1 - \frac{t}{T_2}\right) \tag{3.6}$$

$$\text{Recíproca} : \eta(t) = \frac{\eta_0}{\left(1+\frac{100\,t}{T_2}\right)} \tag{3.7}$$

$$\text{Exponencial} : \eta(t) = \eta_0 e^{\frac{n}{T_2}} \tag{3.8}$$

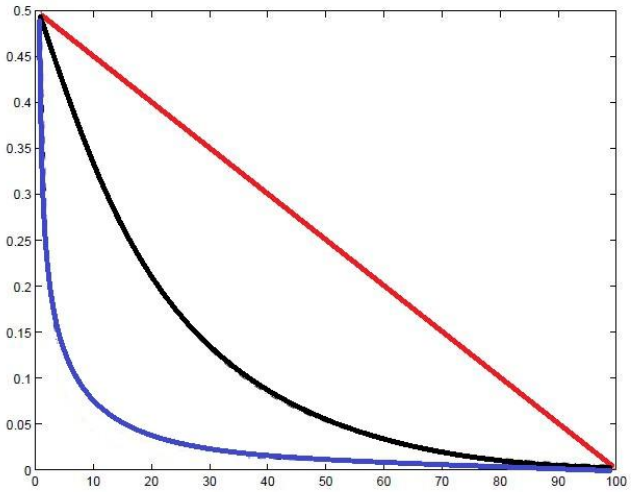


Figura 3.5: Funções de taxa de aprendizagem, onde a cor vermelha representa a função linear, a preta a exponencial e a azul a recíproca

Fonte: (VESANTO, 2000)

A função mais utilizada é a exponencial (Equação 3.8), onde T_2 é uma constante de tempo e t é o número da iteração. Segundo Haykin (2001), a função exponencial juntamente com a função para a redução do raio de vizinhança, é adequada para a formação do mapa de características de forma auto-organizável.

Já para a atualização do raio de vizinhança é utilizada a Equação 3.9. Os valores de *N* e T_2 utilizados por Haykin (2001) são $N = T_2 = 1000$.

$$\sigma(t) = \sigma_0 e^{\frac{n}{T_1}} \tag{3.9}$$

onde $T_1 = \frac{N}{\log(\sigma_0)}$ e σ_0 é o raio de vizinhança inicial.

3.1.1. O Algoritmo *SOM*

Segundo Kohonen (2001), o algoritmo original incremental do *SOM* é basicamente composto por: inicialização, treinamento e atualização.

O primeiro passo do algoritmo é fazer a inicialização dos valores. Neste passo são inicializados os pesos sinápticos dos neurônios presentes na camada de saída, essa inicialização pode ser feita com valores aleatórios ou de forma linear. Também é definido o valor da taxa de aprendizagem, raio topológico inicial e número máximo de iterações.

No treinamento do *SOM*, para cada padrão do vetor de entrada são aplicados os processos de competição, cooperação e adaptação, onde os neurônios competem entre si pelo padrão do vetor de entrada e o neurônio vencedor determina a localização espacial de uma vizinhança de neurônios excitados, tendo esses neurônios seus pesos atualizados.

Após o treinamento é atualizada a taxa de aprendizagem usando uma função decrescente, podendo ser linear, exponencial ou recíproca, em função das iterações. Também é atualizado o raio topológico com uma função monotonicamente decrescente em função das iterações. Depois de feita a atualização o algoritmo retorna ao treinamento. Este processo é realizado até que o critério de parada seja satisfeito, seguido da apresentação da saída dos dados.

3.1.2. Avaliação do aprendizado de um *SOM*

Segundo Silva (2004), existe um conjunto razoável de mecanismos para a avaliação da qualidade de aprendizagem de um *SOM* treinado. As principais métricas utilizadas são o Erro Médio de Quantização e o Erro Topológico (KOHONEN, 2001).

Erro de Quantização (E_q) corresponde à média das distâncias entre cada vetor de dados v_n e o correspondente vetor de pesos w_{bmu}, vetor vencedor no processo competitivo para o padrão v_n. A medida E_q corresponde a acuidade, ou resolução, do mapa e é inversamente proporcional ao número de neurônios, ou seja, o erro de representação diminui com o aumento do número de neurônios no mapa (ZUCHINI, 2003). O índice E_q é dado pela Equação 3.10.

$$E_q = \frac{1}{N}\sum_{n=1}^{N} |v_n - w_{bmu}| \tag{3.10}$$

O Erro topológico (E_t), avalia a capacidade do mapa em representar a topologia dos dados de entrada. Considerando que para cada padrão v_n tem-se o *BMU* como o primeiro neurônio na ordem de competição na grade, e o *BMU2*, que corresponderá ao segundo neurônio nesta ordem. Assim, o erro topológico corresponderá ao percentual de padrões cujo *BMU* e *BMU2* não são adjacentes na grade (SILVA, 2004). O erro topológico é calculado utilizando a Equação 3.11.

$$E_t = \frac{1}{N}\sum_{n=1}^{N} u(v_n) \tag{3.11}$$

onde $u(v_n)$ = 1 se o *BMU* e o *BMU2* não forem adjacentes, e 0 caso contrário.

A medida E_t indica a dificuldade do mapa em representar relações de vizinhança entre os dados, e por esse motivo, indica a necessidade de uma análise ou melhoria no mapeamento, podendo-se realizar um novo treinamento com novos ajustes de parâmetros (BOSCARIOLI, 2008).

3.2. Análise de dados a partir do *SOM*

Segundo Boscarioli (2008), a análise de dados a partir do *SOM* pode ser realizada por uma variedade de técnicas de visualização e análise de agrupamentos, para a busca de padrões em uma base de dados. A Figura 3.6 ilustra um exemplo de análise de dados a partir do *SOM* para uma base de dados que apresenta três grupos, onde na primeira etapa o conjunto de dados é reduzido em vetores protótipos (neurônios), que posteriormente são utilizados por técnicas de visualização ou de agrupamento de dados.

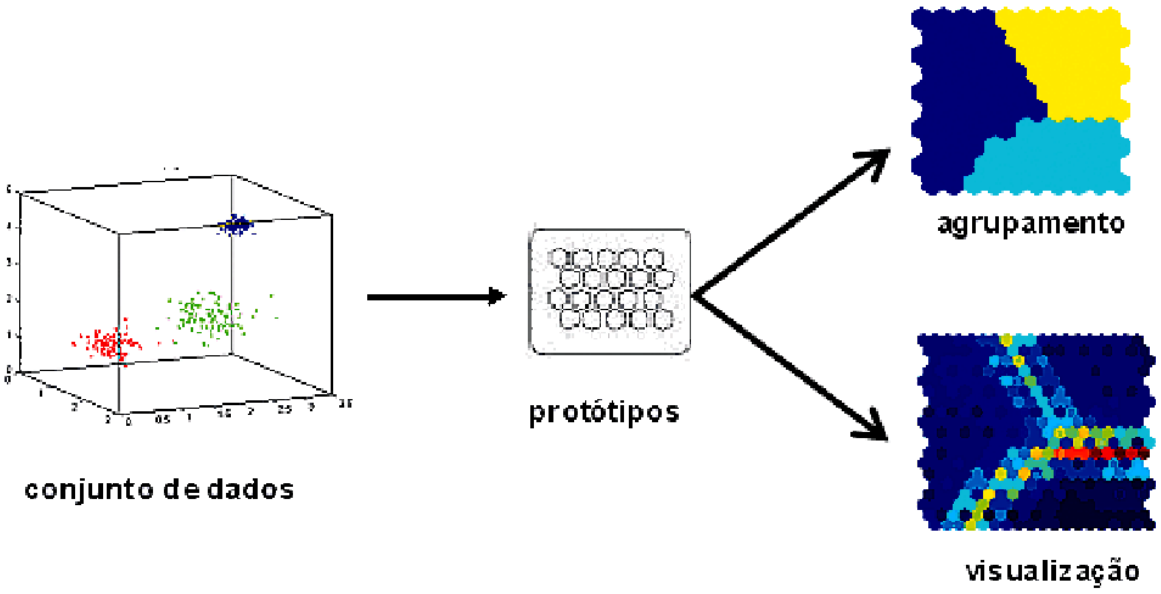


Figura 3.6: Análise de dados a partir do *SOM*

Fonte: (BOSCARIOLI, 2008)

3.2.1. Visualização de um *SOM*

Para efetuar a análise de dados de um mapa auto organizável apenas o mapeamento topologicamente ordenado do *SOM* não é suficiente, pois a informação de distâncias entre os

neurônios é perdida. Por esse motivo se torna necessário a utilização de técnicas que possibilitem a análise visual do possível resultado da ordenação topológica (SILVA, 2004).

Segundo Silva (2004), a técnica mais utilizada para a visualização de *SOM* é a matriz de distâncias unificadas ou Matriz-U. Além do mapa de distâncias unificadas utiliza-se a Matriz de Densidade, que representa os números de padrões de entrada associados a cada neurônio do mapa.

3.2.1.1. Matriz de Distâncias Unificadas: Matriz-U

A Matriz de distâncias unificadas é um método de visualização de um *SOM* treinado que foi desenvolvido por Ultsch (1992), com o objetivo de permitir a análise visual da ordenação topológica dos neurônios.

Na Matriz-U é utilizada a mesma métrica de distância usada no treinamento do *SOM* para calcular a distância entre neurônios adjacentes. O resultado da aplicação da Matriz-U sobre um *SOM* é uma imagem $f(x, y)$, onde a intensidade de cada pixel da imagem corresponde a uma distância entre os neurônios adjacentes. Considerando um *SOM* de dimensões $X \times Y$, aplicando-se uma Matriz-U sobre esse mapa, o resultado será de uma imagem de dimensões $(2X - 1) \times (2Y - 1)$ (SILVA, 2004).

Segundo Costa (1999), pode-se pensar em uma imagem tridimensional em que o valor da coordenada (x, y) é representado por um ponto na coordenada z. No caso de uma Matriz-U representada em três dimensões, superfície topológica, como na Figura 3.7, os vales formados pelo relevo topográfico, correspondem a regiões em que os neurônios são similares e essas regiões são candidatas a representar agrupamentos de neurônios. Já os locais onde os valores são mais elevados, onde se formam montanhas, correspondem a regiões em que os neurônios são mais dissimilares, essas regiões ou neurônios são associados a fronteiras de agrupamento. Os valores dos pixels variam de acordo com a legenda à direita da matriz-U.

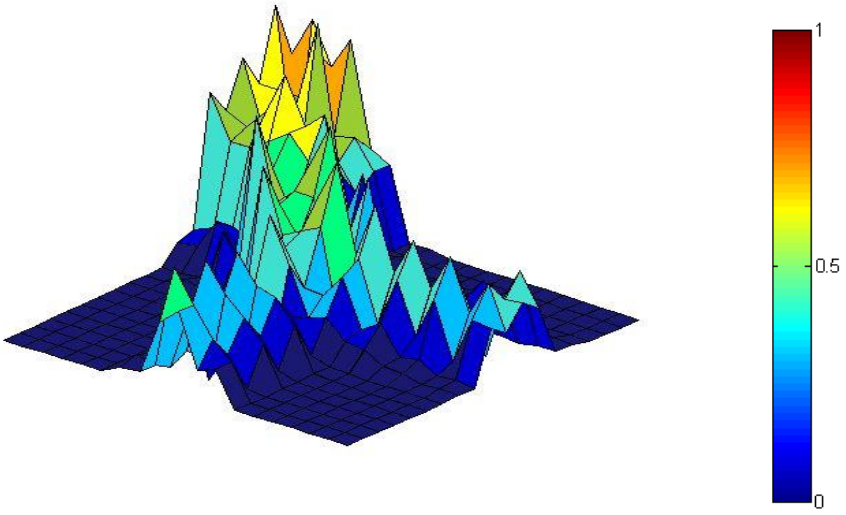


Figura 3.7: Exemplo da matriz-U representada por superfície topológica de um *SOM* 10x10

Uma Matriz-U de duas dimensões (Figura 3.8) também pode ser interpretada da mesma forma que a matriz representada por superfície topológica. Os pixels com a coloração mais elevada representam neurônios vizinhos mais dissimilares e correspondem a fronteiras de agrupamento. Valores mais baixos representam neurônios vizinhos mais similares que correspondem aos vales que agrupam os neurônios similares.

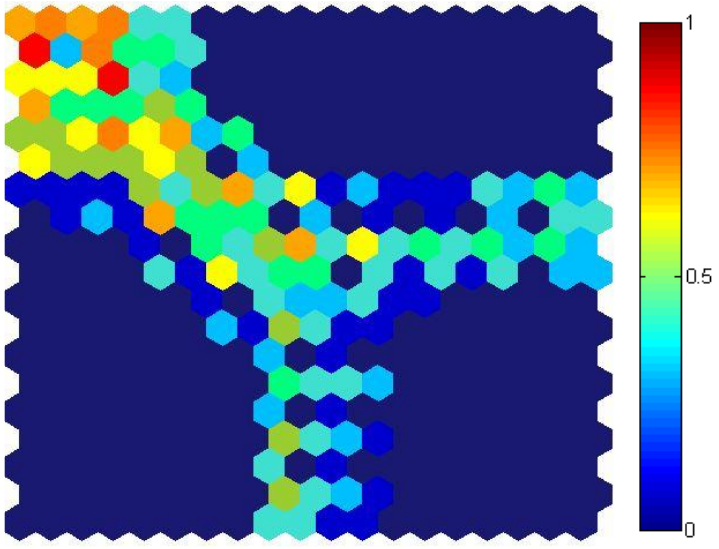


Figura 3.8: Exemplo da matriz-U na forma 2D de um *SOM* 10x10

Em um *SOM* bidimensional retangular é encontrada a Matriz-U calculando-se as distâncias $dx, dy\ e\ dxy$ (Figura 3.9) para cada neurônio. Esta lógica de aplicação da Matriz-U para

vizinhança retangular também pode ser estendida e aplicada em uma vizinhança hexagonal. Considere um mapa retangular de dimensões $X \times Y$. Seja $[b_{x,y}]$ a matriz de neurônios e $wi_{x,y}$ a matriz de pesos. Para cada neurônio em b existem três distâncias a seus vizinhos: $dx, dy\ e\ dxy$, que devem ser calculadas e armazenadas na Matriz-U (COSTA, 1999).

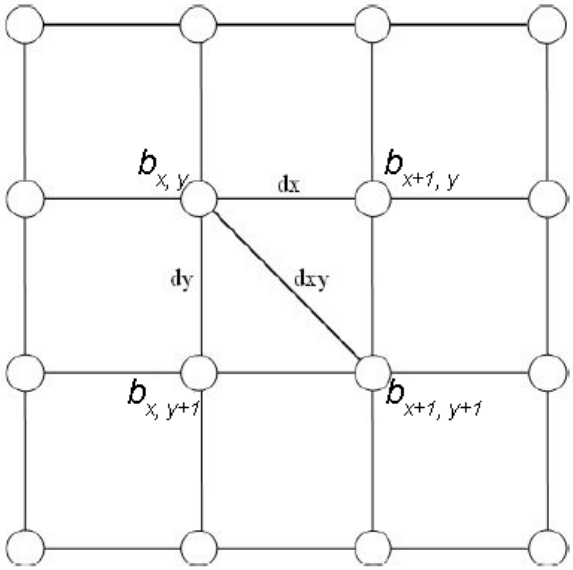


Figura 3.9:Distâncias para a construção da matriz-U

Fonte: (COSTA, 1999).

As distâncias $dx, dy\ e\ dxy$ devem ser calculadas pela mesma métrica utilizada no treinamento do *SOM*. As Equações 3.12, 3.13 e 3.14 apresentam as equações das distâncias considerando como métrica a distância Euclidiana:

$$dx(x,y) = \sqrt{\sum_i (wi_{x,y} - wi_{x+1,y})^2} \quad (3.12)$$

$$dy(x,y) = \sqrt{\sum_i (wi_{x,y} - wi_{x,y+1})^2} \quad (3.13)$$

$$dxy(x,y) = \frac{1}{2\sqrt{2}}\left[\sqrt{\sum_i (wi_{x,y} - wi_{x+1,y+1})^2} + \sqrt{\sum_i (wi_{x,y+1} - wi_{x+1,y})^2}\right] \quad (3.14)$$

Para cada neurônio de b, as distâncias para os vizinhos tornam-se $dx, dy\ e\ dxy$, e são plotadas em uma matriz-U de tamanho $(2X-1) \times (2Y-1)$. A matriz-U é preenchida de acordo com o Quadro 3.1:

Os valores *I* e *P* apresentados na tabela representam a posição do neurônio impar e par, respectivamente. O valor du apresentado pode ser calculado em função dos valores dos elementos vizinhos do neurônio relativo ao du. O valor de du pode ser a média ou a mediana destes elementos (COSTA, 1999). Segundo Silva (2004), du pode ser também o valor máximo ou mínimo dos elementos circunvizinhos.

Quadro 3.1:Esquema para o preenchimento dos elementos da matriz-U

$i \quad j$	(i,j)	$U_{i,j}$
$I \quad P$	$(2X+1,2Y)$	$dx(x,y)$
$P \quad I$	$(2X,2Y+1)$	$dy(x,y)$
$I \quad I$	$(2X+1,2Y+1)$	$dxy(x,y)$
$P \quad P$	$(2X,2Y)$	$du(x,y)$

Fonte: (COSTA, 1999).

A Figura 3.10 traz os elementos da Matriz-U, preenchidos de acordo com o Quadro 3.1.

$$\begin{bmatrix} du(0,0) & dx(0,0) & du(1,0) & \dots & du(X\text{-}1,0) \\ dy(0,0) & dxy(0,0) & dy(1,0) & \dots & dy(X\text{-}1,0) \\ du(0,1) & dx(0,1) & du(1,1) & \dots & du(X\text{-}1,1) \\ dy(0,1) & dxy(0,1) & dy(1,1) & \dots & dy(X\text{-}1,1) \\ \dots & \dots & \dots & \dots & \dots \\ du(0,Y\text{-}1) & du(0,Y\text{-}1) & du(1,Y\text{-}1) & \dots & du(X\text{-}1,Y\text{-}1) \end{bmatrix}$$

Figura 3.10: Elementos da matriz-U

Fonte: (COSTA, 1999).

3.2.1.2. Matriz de Densidade

A Matriz de Densidade foi proposta inicialmente por Zhang (1993) e é baseada na densidade do mapa, ou seja, para cada neurônio do mapa é calculado o número de padrões associados a ele.

Na Matriz de Densidade os vales ou fronteiras topológicas entre os agrupamentos são determinados pelos neurônios com um número reduzido de padrões associados, esses neurônios permitem a criação dos agrupamentos do *SOM*, utilizando-se linhas para unir esses neurônios e delimitar os grupos (SIQUEIRA, 2005).

Na Figura 3.11 encontra-se representada a Matriz de Densidade de um *SOM* 10 × 10, nela encontram-se representados os números de padrões de entrada associados a cada neurônio do mapa, onde os neurônios com nenhum padrão associado determinam as fronteiras entre os agrupamentos, formando assim três grupos de neurônios.

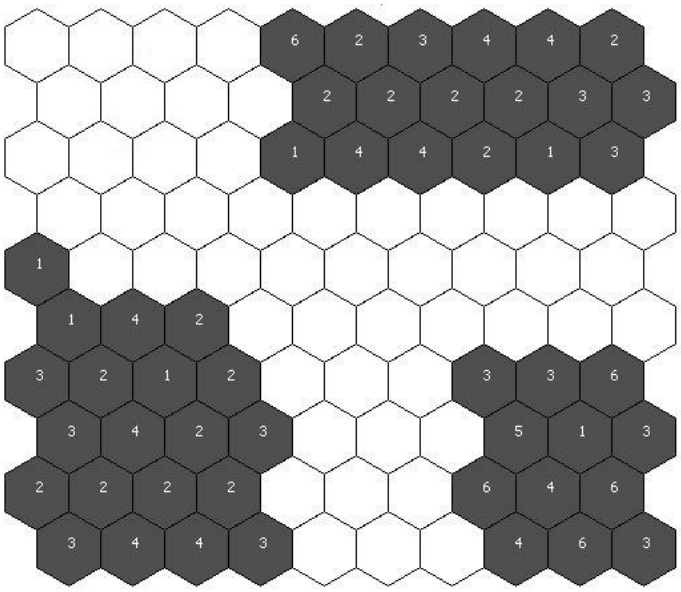


Figura 3.11: Exemplo de matriz de densidade de um *SOM* 10x10

3.2.2. Agrupamento de dados por Mapa Auto-Organizável

Segundo Boscarioli (2008), a saída de um mapa *SOM* fornece apenas uma representação dos dados organizados topologicamente pelos neurônios. Esta representação muitas vezes não fornece uma análise visual direta, dificultando a análise de grupos existentes. Sendo assim, somente o mapeamento topológico do *SOM* não é suficiente para realizar uma análise de agrupamentos. Para a realização da análise de agrupamentos via *SOM*, outras técnicas devem ser utilizadas sobre o mesmo, para que os resultados sejam observáveis e úteis na geração de novos conhecimentos. A seguir serão apresentadas as técnicas mais utilizadas para agrupar dados a partir de um *SOM* treinado.

3.2.2.1. Algoritmo SL-*SOM*

O algoritmo SL-*SOM* (*Self-Labeled SOM*) foi proposto por Costa (1999). Este método consiste em encontrar grupos de dados a partir de um *SOM* treinado, através do

particionamento da matriz-U. O algoritmo SL-*SOM* utiliza o método de segmentação de imagens *Watershed* para particionar a matriz-U em regiões conectadas.

Segundo Costa (1999), o método *Watershed* foi proposto inicialmente por Beucher e Lantuejoul (1979), e tem sido considerada uma das ferramentas mais eficientes utilizadas dentro da morfologia matemática para a segmentação de imagens. Esse método baseia-se no princípio de inundação de relevos topográficos. Considerando uma imagem em níveis de cinza ela pode ser vista como um relevo topográfico. Os *pixels* com valores baixos, ou seja, as regiões mais escuras da imagem representam os vales, e os *pixels* com valores elevados, porções mais claras da imagem, representam as montanhas do relevo (COSTA, 1999).

Conforme as bacias vão sendo inundadas, águas provenientes de diferentes bacias se encontram,formando, nos pontos de encontro, represas ou linhas divisoras de águas, que são chamados de marcadores ou *watersheds*. Como resultado, o relevo é particionado em regiões ou bacias separadas pelos marcadores.

Ainda segundo Costa (1999) o algoritmo SL-*SOM* é composto pelos seguintes passos:

1. Obtenção da matriz-U de um *SOM* treinado.
2. Encontrar os marcadores para a matriz-U.
3. Aplicar o algoritmo de *Watershed* sobre a matriz-U usando os marcadores obtidos no passo 2.
4. Rotular as regiões conectadas da imagem segmentada no passo 3.
5. Copiar os rótulos obtidos no passo 4 para os neurônios associados a cada pixel da matriz-U.
6. Caso ainda existam neurônios não rotulados, rotule-os usando o método do vizinho mais próximo, calculando as distâncias no espaço de pesos dos neurônios, e atribuindo o código do neurônio rotulado mais próximo.

Observa-se na Figura 3.12, um exemplo de aplicação do algoritmo SL-*SOM*. O algoritmo apresenta bons resultados, mas a sua aplicação não é recomendada para mapas com número reduzido de neurônios ou para problemas com estrutura complexa, pois a matriz-U tende a ser de difícil interpretação para estes casos. Apesar das limitações do algoritmo com relação à complexidade da Matriz-U pode-se afirmar que o SL-*SOM* oferece um bom mecanismo de investigação de dados multivariados (SILVA, 2004).

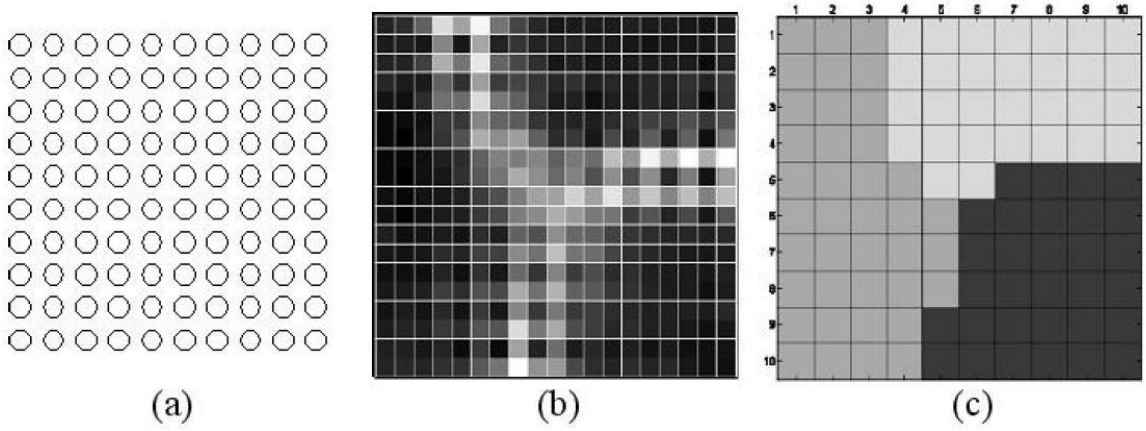


Figura 3.12: Exemplo da aplicação do método SL-*SOM*. (a) *SOM* bidimensional 10x10; (b) matriz-U gerada a partir deste *SOM* treinado; (c) rotulação dos neurônios do *SOM* com o auxilio da imagem (b)
Fonte: Adaptado de Costa (1999).

3.2.2.2. Agrupamento por Matriz de Densidade

Segundo Siqueira (2005), o agrupamento de um *SOM* pode ser realizado através da matriz de densidade, utilizando o método proposto em Xu e Li (2002). Este método leva em consideração a densidade de cada neurônio, ou seja, o número de padrões de entrada associados ao neurônio.

O primeiro passo do método é encontrar a densidade para cada neurônio do *SOM* treinado. Os neurônios com a densidade não nula formam um conjunto *P*, esse conjunto de neurônios deve ser classificado em ordem crescente, através da densidade de cada um.

Depois de encontrado o conjunto *P*, deve-se determinar a distância entre cada par de neurônios i e j presentes no conjunto, conforme Equação 3.15:

$$distância(i,j) = ||w(i) - w(j)|| \tag{3.15}$$

Se a distância entre *i* e *j* for menor que *E* (onde *E* representa um erro usado para determinar a tolerância de distância entre as características de neurônios do mesmo agrupamento), então *i* e *j* são considerados membros do mesmo grupo. Este passo é repetido até concluir a comparação entre os elementos do conjunto *P*.

Para cada grupo encontrado, o neurônio com a maior densidade é considerado o centróide do agrupamento.

3.2.2.3. Metodologia de Vesanto & Alhoniemi

A proposta de Vesanto e Alhoniemi (2000) é usar o *SOM* como um redutor de tamanho do conjunto de dados a ser analisado. O método consiste em duas fases, na primeira os dados são usados para treinar um *SOM* e então na segunda fase são aplicados métodos tradicionais de descoberta de agrupamento nos vetores de pesos do *SOM* treinado. Nesta segunda fase, os autores utilizaram métodos hierárquicos aglomerativos (Ligação Simples, Ligação Completa e Ligação Média) e o algoritmo K-médias para realizar o agrupamento.

Segundo Vesanto e Alhoniemi (2000), a principal função do *SOM* é reduzir o volume de dados e o custo computacional do processamento dos algoritmos tradicionais de agrupamento. Segundo Boscarioli (2008), a aplicação do *SOM* nesse método funciona como uma espécie de pré-processamento para o agrupamento.

Um grande problema encontrado nessa abordagem é que ela ignora grupos formados com apenas um elemento, que podem ser vistos como grupos raros. Outro problema é a utilização do algoritmo *K*-médias para o agrupamento, pois ele calcula o grau de separação entre os grupos por meio de centróides, o que impõe uma geometria esférica aos grupos, o que nem sempre é encontrado em dados reais. Apesar dessas limitações, essa é uma das metodologias mais utilizadas para agrupamento de dados por meio do *SOM* (BOSCARIOLI, 2008).

3.2.2.4. Metodologia de Boscarioli

A metodologia proposta por Boscarioli (2008) tem com idéia principal utilizar a propriedade de preservação de vizinhança topológica dos dados no treinamento do *SOM* como critério de agrupamento e validação do mesmo.

Segundo Boscarioli (2008), a metodologia divide-se em quatro passos:

1. **Treinar o mapa *SOM*:** nesta etapa é gerado um mapa *SOM* e também são calculadas as medidas de avaliação de aprendizado, para averiguar a qualidade de mapeamento aos vetores de dados de entrada.
2. **Construir um dendrograma sobre os protótipos do *SOM* treinado:** esta etapa tem o objetivo de encontrar os neurônios vizinhos que se agruparam pela máxima

similaridade. Isto é feito utilizando o cálculo da proximidade mínima entre os neurônios (Equação 3.16), onde D_i *e* D_j representam dois grupos de vetores de dados. A proximidade mínima (*Single Link*) é definida como a mínima distância (máximo de similaridade) entre dois grupos diferentes.

$$d_{g\_min} = (D_i, D_j) = min_{x \in Di,\ y \in Dj}\ d(x, y) \tag{3.16}$$

Cada neurônio *BMU* forma uma folha do dendrograma, e os demais níveis são definidos pelas ligações entre os neurônios, formando um agrupamento inicial.

3. **Avaliar potenciais grupos:** em cada ramo do dendrograma avaliam-se os potenciais grupos, no intuito de encontrar a estrutura de agrupamento subjacente aos dados, caso exista uma, determinando o número de grupos existentes na base de dados em análise.
4. **Visualizar a estrutura de grupos no mapa:** a partir da estrutura de agrupamento encontrada, colore-se o *SOM* de saída para visualização, onde cada cor representa um grupo.

Na avaliação dos potenciais grupos, para cada nível do dendrograma é avaliado o número de grupos existentes. Uma matriz de ligações (ML), simétrica de ordem r é construída, onde r corresponde ao número de neurônios *BMU*s presentes no potencial grupo em avaliação. A Figura 3.13 ilustra como é feita a construção da matriz de ligações ML.

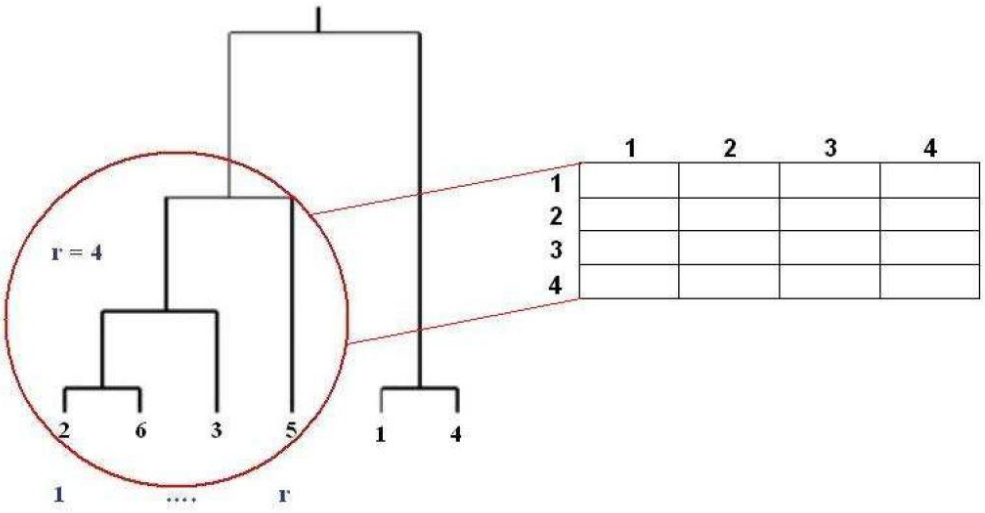


Figura 3.13: Construção da matriz de ligações ML

Fonte: (BOSCARIOLI, 2008).

A diagonal principal da matriz de ligações é preenchida com zeros e a relação *linha x coluna* estabelece uma junção entre os neurônios, a qual é obtida pela Equação 3.17.

Para cada BMU, encontra-se a sua máxima distância em relação aos vetores de dados por ele representados, e a mínima distância em relação aos vetores de dados que ele não representa tal que:

$$min_{x_j \notin BMU_u}\left(d\left(BMU_u, x_j\right)\right) < max_{x_i \in BMU_u}\left(d(BMU_u, x_i)\right) \qquad (3.17)$$

onde $x_j \notin BMU_u$ significa que o dado x_j não é representado pelo neurônio u e $x_i \in BMU_u$ significa que o dado x_i está representado pelo neurônio u. Se a condição da Equação 3.17 for verdadeira, encontram-se o neurônio *l* tal que $x_j \notin BMU_l$, e estabelecem-se a junção *u* e *l*, índices de *BMU*s, em ML. Neste caso, a relação *linha x coluna* é considerada consistente, indicando a junção, e a matriz de ligações assume o valor 1 na posição referente a relação em questão.

Segundo Boscarioli (2008), o agrupamento ideal é considerado, quando, dentro de cada nível do dendrograma para cada *BMU*, na construção da ML, todas as suas posições estão preenchidas com o valor 1, exceto a diagonal principal.

3.3. Considerações Finais

Segundo Castro (1999), o *SOM* tem a capacidade de resolver problemas de alta dimensionalidade, tais como: extração de características e classificação de imagens, controle adaptativo de robôs, equalização, transmissão de sinais, agrupamento e visualização de dados. Além disso, pode-se considerar que o *SOM* é um dos modelos representativos mais realísticos baseados no aprendizado do cérebro humano.

Somente o mapeamento topologicamente ordenado do *SOM* não é suficiente para a análise de dados, se fazendo necessária a utilização de outras técnicas a partir de um *SOM* treinado para a realização da mesma. Estas técnicas podem ser de visualização ou de agrupamento, como apresentadas neste capítulo.

Também foi apresentada neste capítulo uma breve descrição sobre as Redes Neurais Artificiais, seguido dos principais conceitos e características dos Mapas Auto-Organizáveis,

com questões sobre métricas utilizadas e as parametrizações, além dos métodos utilizados para a análise do *SOM*.

Neste trabalho, além do algoritmo *SOM*, foram implementadas as técnicas de visualização apresentadas, a Matriz-U e Matriz de Densidade, e as técnicas de agrupamento a partir do *SOM*: Agrupamento por Matriz de Densidade, Algoritmo SL-*SOM* e a metodologia em Vesanto e Alhoniemi (2000).

Capítulo 4

Implementação na Ferramenta YADMT

A implementação da metodologia de agrupamento de dados a partir do *SOM* foi desenvolvida na Ferramenta YADMT utilizando linguagem de programação Java, conforme o modelo proposto por Benfatti *et al.*(2010).

A Ferramenta YADMT é uma proposta de ferramenta de *KDD* em desenvolvimento na UNIOESTE, em alternativa a outras já existentes na área. Ela é uma ferramenta livre e modular, o que permite o desenvolvimento em um ambiente acadêmico colaborativo e facilita evoluções. Atualmente ela conta com os módulos de Pré-Processamento, Classificação, Extração de Características e o módulo de Agrupamento de dados, que está em fase de desenvolvimento.

A seguir, serão apresentadas as implementações realizadas na ferramenta YADMT para a construção da metodologia de agrupamento de dados a partir do *SOM* no módulo de agrupamento de dados.

4.1. Implementação do Algoritmo *SOM*

A Figura 4.1 ilustra a tela inicial do *SOM* no módulo de Agrupamento de Dados da ferramenta YADMT. O agrupamento de dados a partir do *SOM* divide-se em duas etapas, a primeira etapa é o treinamento do *SOM* e a segunda etapa é a análise do mapa treinado, podendo ser feita por métodos de visualização ou de agrupamento.

A implementação foi feita de forma que o usuário pode realizar o treinamento do mapa *SOM*, e após o treinamento pode-se escolher algum método para a análise. Ao término do treinamento é apresentado em tela o Erro de Quantização e Erro Topológico para a avaliação de aprendizagem do mapa treinado. Pode-se realizar o treinamento quantas vezes achar necessário antes da aplicação de algum método de análise, podendo entre as execuções alterar

parâmetros ou número de iterações do algoritmo. Da mesma forma, os métodos de agrupamento podem ser executados inúmeras vezes sob um *SOM* treinado. Após a execução de cada método de agrupamento são mostrados em tela os resultados obtidos. Os métodos de visualização são executados automaticamente após cada treinamento realizado.

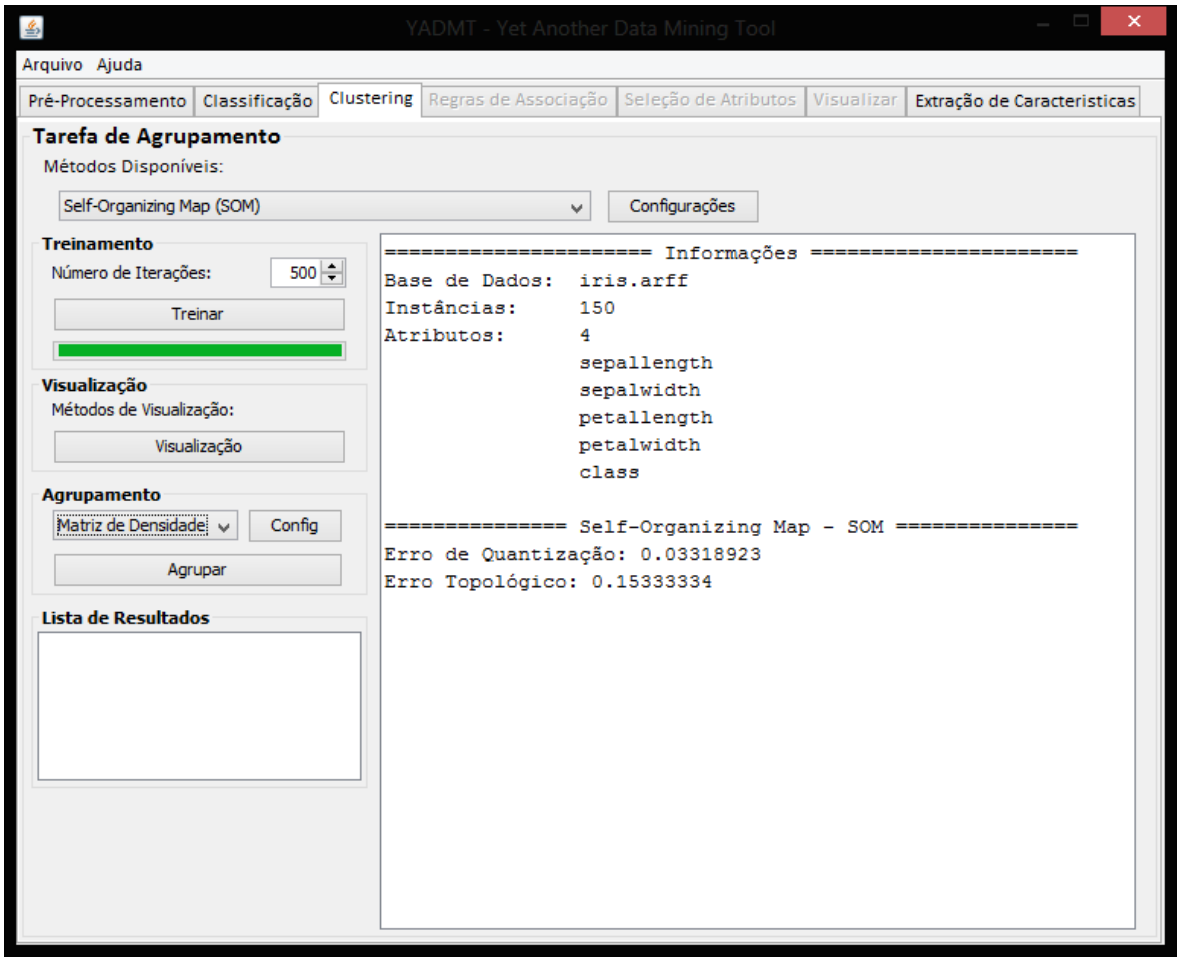


Figura 4.1: Tela da ferramenta YADMT– Tela inicial do *SOM,* no módulo de Agrupamento de Dados

Na tela de configurações do *SOM* é possível ajustar alguns parâmetros do algoritmo (Figura 4.2). No painel de Configurações do Mapa é possível definir as dimensões da grade de neurônios, podendo ser bidimensional (2-D *SOM*) ou unidimensional (1-D *SOM*), e abaixo é mostrada a quantidade de neurônios que vão estar presentes naquela configuração.

É possível ajustar o tipo de vizinhança topológica, retangular ou hexagonal, e o raio de vizinhança inicial. No painel de Distância é possível escolher a distância a ser utilizado durante todo o algoritmo, treinamento do SOM e cálculo da matriz-U. As distâncias

disponíveis são: Distância Euclidiana, Mahalanobis, City Block, Chebyshev, Similaridade de Cosseno e Coeficiente de Correlação.

No painel de Aprendizagem pode-se escolher a função de atualização a ser utilizada, podendo ser Exponencial, Recíproca ou Linear, além de ajustar o valor da aprendizagem inicial do algoritmo.

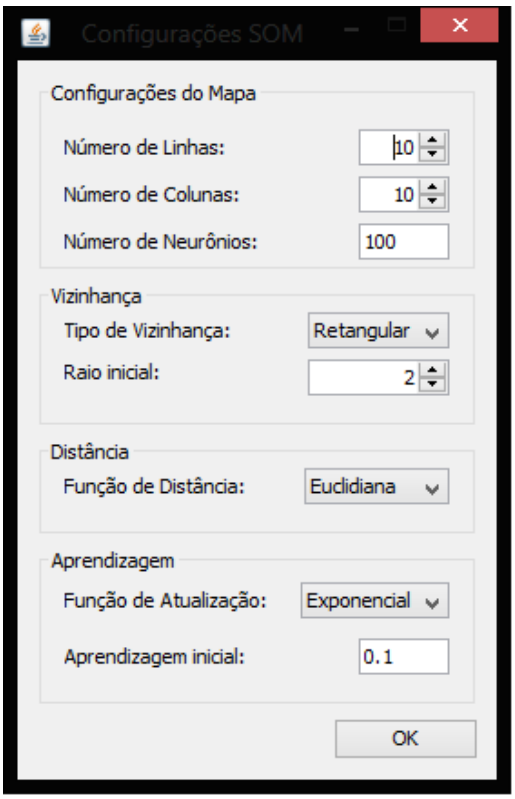


Figura 4.2:Tela de Configuração dos parâmetros do *SOM* na YADMT.

4.2. Implementação dos Métodos de Visualização

Como métodos de visualização foram implementadas as duas técnicas apresentadas no Capítulo 3, a Matriz-U e a Matriz Densidade. Para a Matriz-U, foram implementadas as duas formas da mesma, a sua forma tridimensional, com representação da superfície topológica e sua forma bidimensional, conforme exemplos que seguem nas Figuras 4.3 e 4.4, respectivamente.

Para a Matriz-U com a representação por superfície topológica a implementação foi feita baseada no modelo de câmera proposto por Smith (1983), que faz a transformação dos pontos tridimensionais para a dimensão da tela.

A ocultação das faces não visíveis ao observador deu-se pela implementação do algoritmo de ocultação de faces do pintor, que é uma das soluções mais simples para resolver o problema de visibilidade em gráficos 3D projetados em um plano 2D. A aplicação do Algoritmo do Pintor foi de suma importância para a visualização correta da superfície topológica, que consiste em pintar primeiramente as partes distantes do observador em uma cena, e então as cobrir com as partes mais próximas.

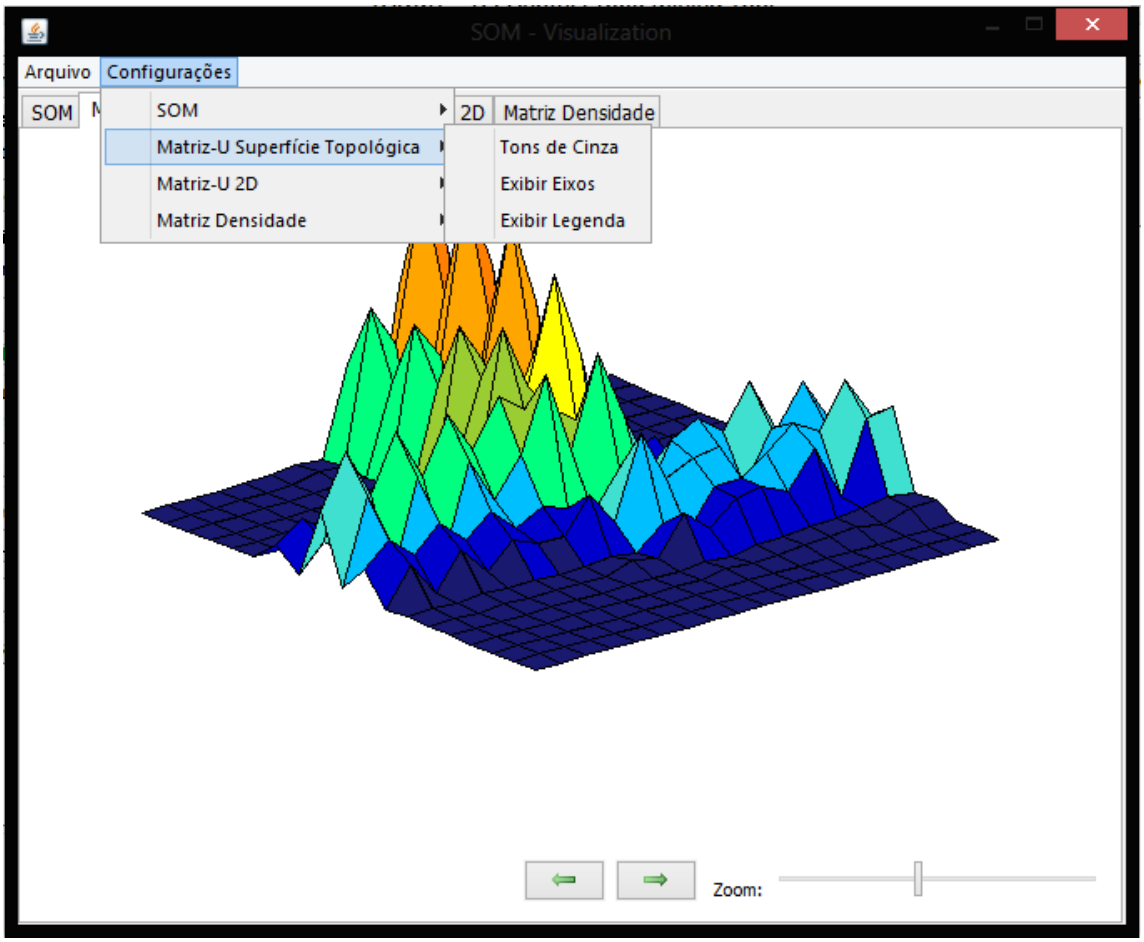


Figura 4.3: Tela da ferramenta YADMT – Matriz-U representada por superfície topológica

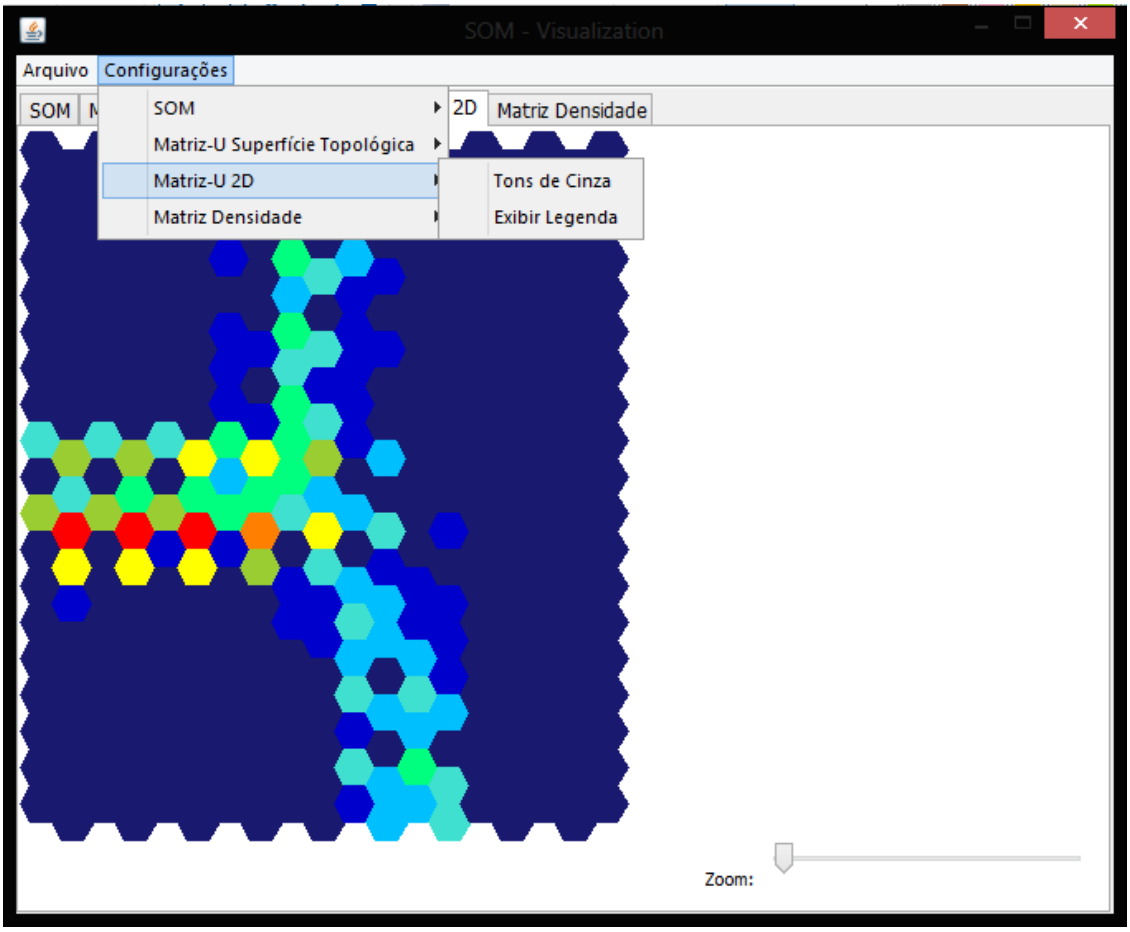


Figura 4.4: Tela da ferramenta YADMT – Matriz-U bidimensional

Na Matriz de Densidade, ilustrada na Figura 4.5, os neurônios com padrões associados são representados na cor cinza, já os neurônios sem nenhum padrão associado são representados pela cor branca. Nos neurônios com padrões associados também é ilustrado o número de padrões associados a ele. Ainda, é possível interagir com a Matriz de Densidade; ao clicar no neurônio é possível visualizar quais padrões estão associados e os pesos do neurônio após o treinamento, como mostrado para o neurônio em vermelho na Figura 4.5.

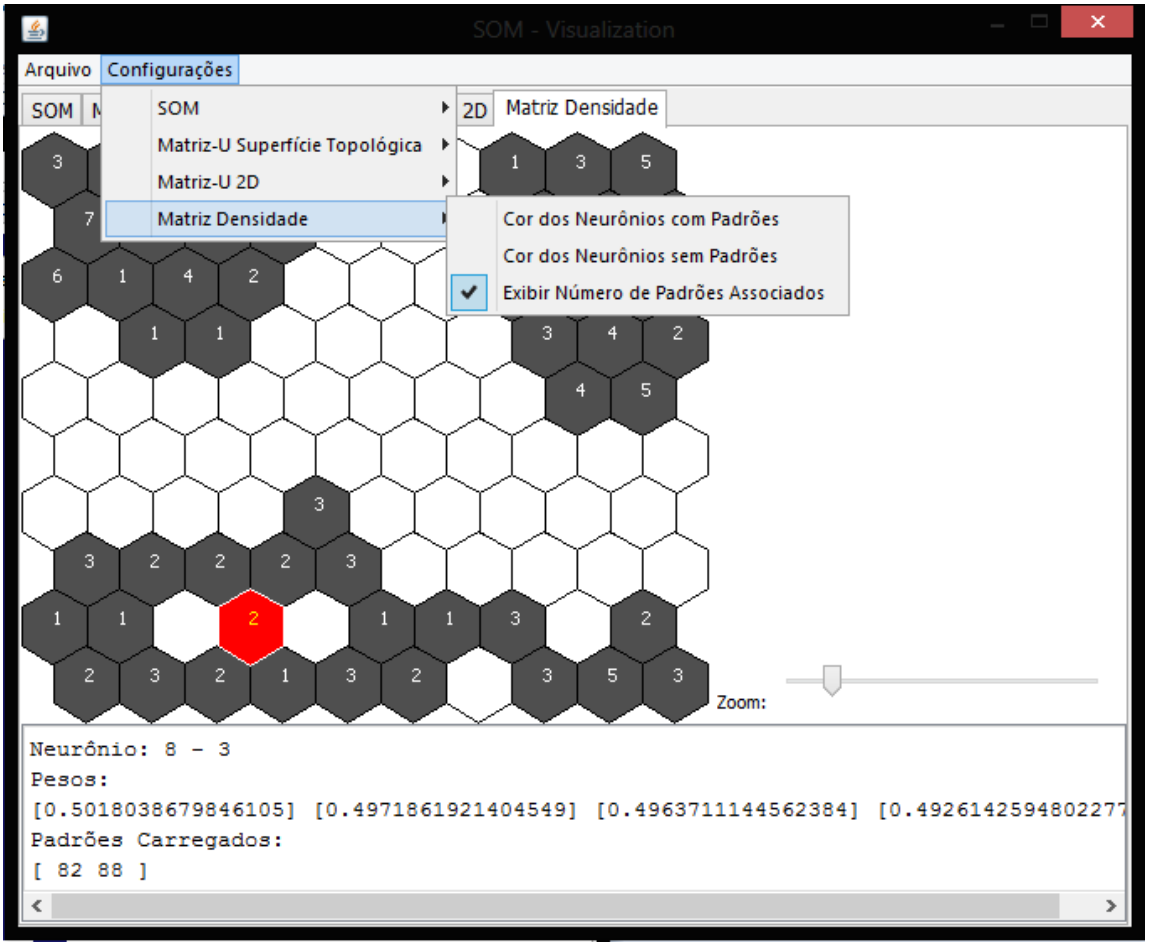


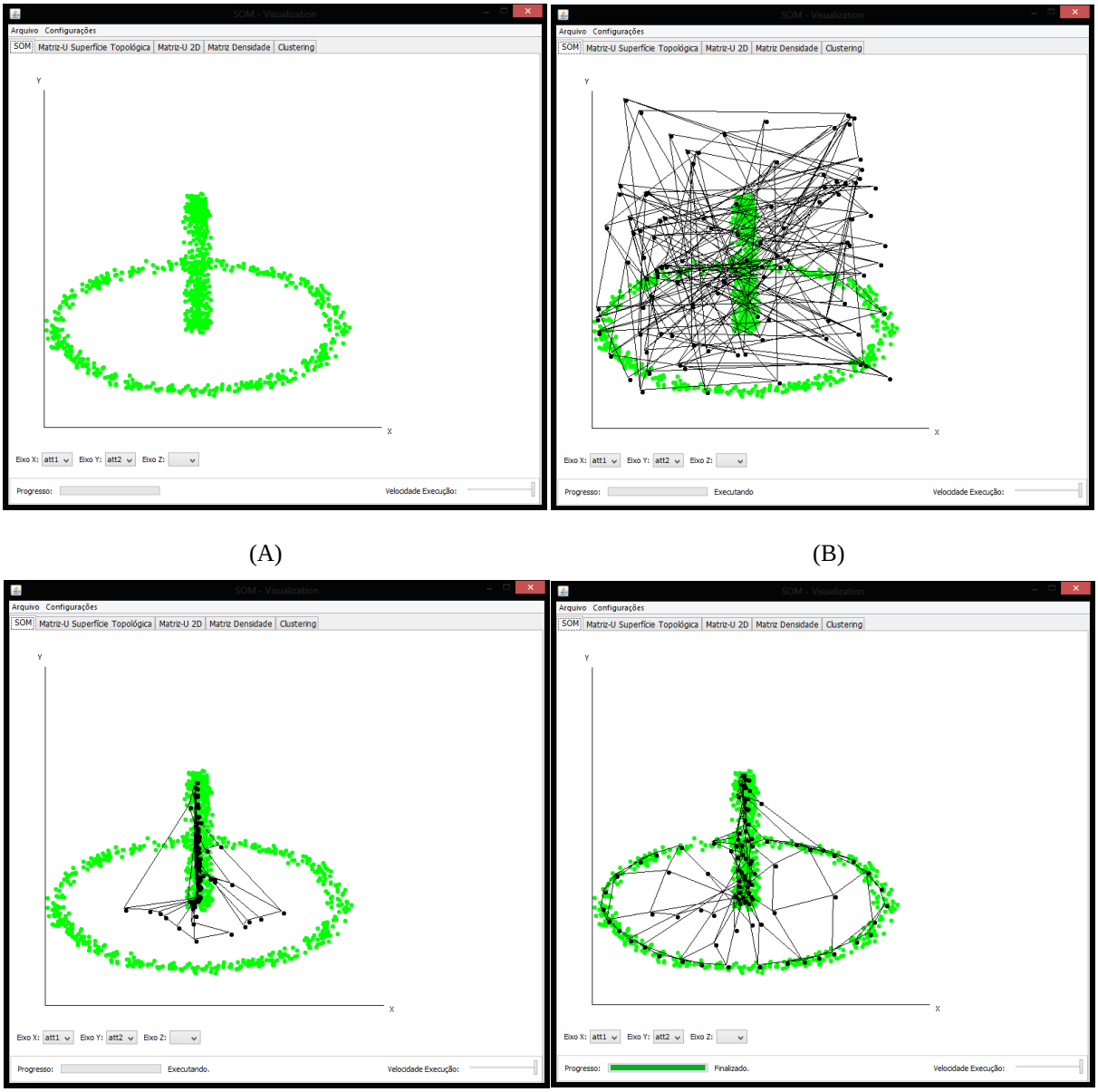
Figura 4.5: Tela da ferramenta YADMT – Matriz de Densidade

Como se pode observar nas Figuras 4.3, 4.4 e 4.5, cada método de visualização implementado possui configurações próprias, como trocar as cores de elementos e exibir ou não elementos na tela. Além disso, todos os métodos de visualização podem ser exportados como imagem no formato de imagem *png*.

Ainda no módulo de visualização do *SOM* foi implementado um método para visualização do treinamento do mapa. Nele é possível visualizar treinamento do mapa em tempo real de execução. Em alguns casos ele auxilia na escolha dos parâmetros para o algoritmo *SOM*, pois oferece a visualização do comportamento do mapa na execução do algoritmo.

A Figura 4.6 ilustra o método de visualização implementado em diferentes iterações do algoritmo *SOM* para o treinamento de um mapa utilizando a base de dados *Chainlink Dataset* (proposto por Ultsch e Vetter (1994)). Na ilustração a base de dados é representada pelos pontos verdes do gráfico, já o mapa é representado na cor preta, onde os neurônios são ilustrados pelos pontos e as linhas são as ligações entre eles.

Na Figura 4.6 a ilustração (A) mostra a base de dados plotada pelo algoritmo. Em (B) é ilustrado a iniciação aleatória do *SOM*. Já em (C) é possível visualizar o inicio da ordenação do mapa, e em (D) é mostrado mapa treinado ao fim do algoritmo.



(A) (B)

(C) (D)

Figura 4.6: Tela da ferramenta YADMT – Método de visualização do treinamento do *SOM*

Este método implementado ainda permite a seleção dos atributos que devem ser plotados em cada eixo cartesiano, podendo ser um gráfico 2D para a seleção de dois atributos ou 3D para a seleção de três atributos da base de dados.

4.3. Implementação dos Métodos de Agrupamento

A utilização de *SOM* para o Agrupamento de Dados é o foco deste trabalho. Os métodos de agrupamento implementados foram: Algoritmo SL-*SOM*, Agrupamento por Matriz de Densidade, 1D-*SOM* e a metodologia de Vesanto e Alhoniemi (2000) utilizando como métodos tradicionais de descoberta de agrupamento as Ligações Simples, Média e Completa e o método de Ward. A Figura 4.7 mostra a tela de seleção dos métodos na ferramenta *YADMT*.

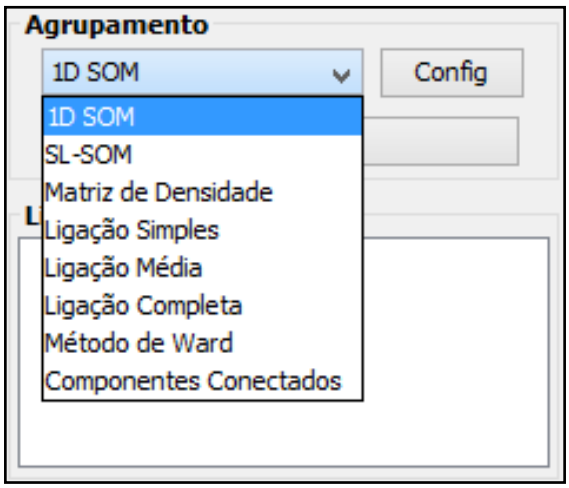


Figura 4.7: Tela da ferramenta YADMT – Seleção dos métodos de agrupamento

Além dos métodos citados, ainda foi proposto um método de agrupamento que faz a extração dos componentes conectados da matriz de Densidade, formando assim os agrupamentos.

4.3.1. Métodos de Agrupamento Unidimensionais

4.3.1.1. 1D-*SOM*

Este método consiste em pré-definir o número de grupos que se deseja ter para a base de dados. Este número de entrada é o número de neurônios no mapa *SOM*, que terá a forma unidimensional ($1 \times n$, onde n é o número de neurônios).

Ao final do treinamento do *SOM* os padrões são atribuídos aos neurônios mais similares e cada neurônio representa um grupo. A Figura 4.15 ilustra a matriz de densidade para uma base de dados artificial onde foram pré-definidos três grupos. Pode-se observar os três neurônios, cada um representando um grupo com 50 padrões associados.

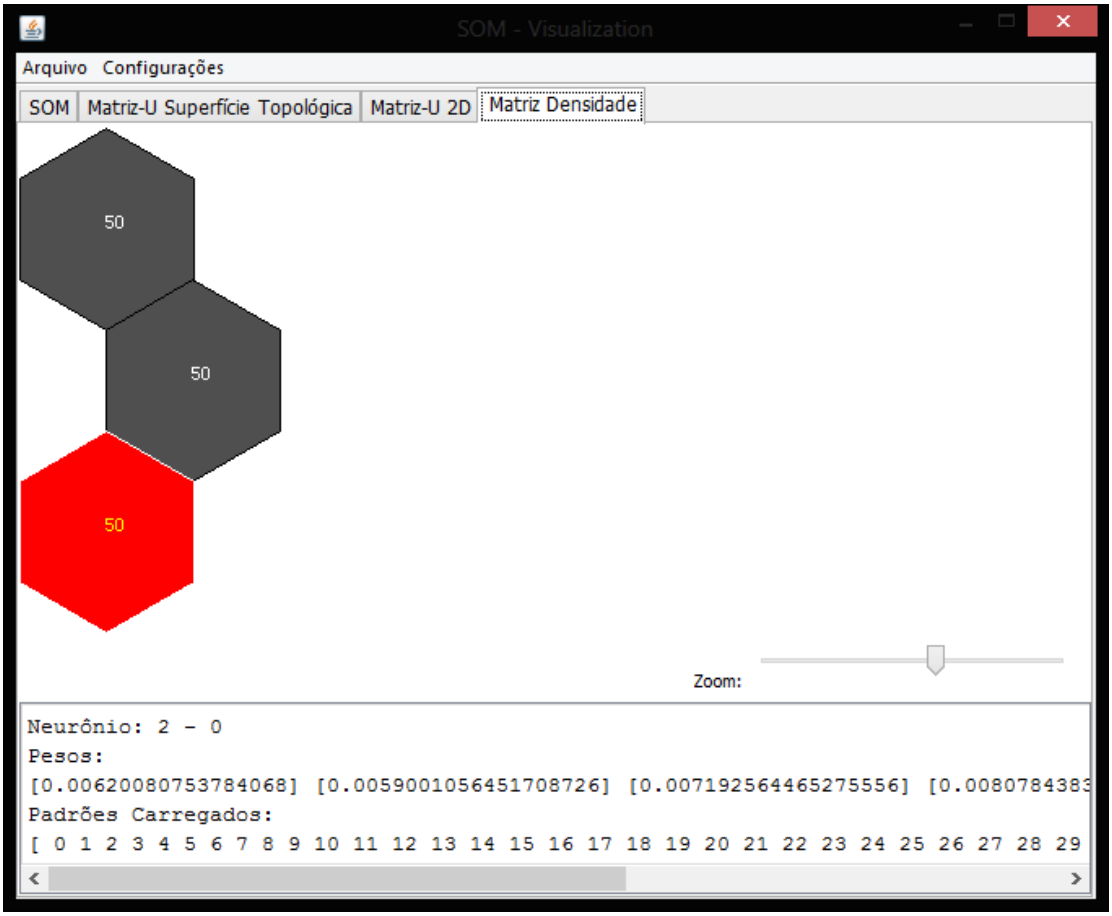


Figura 4.8: Tela da ferramenta YADMT – Matriz de densidade de um *SOM* de uma dimensão (1 × 3)

4.3.2. Métodos de Agrupamento Bidimensionais

4.3.2.1. Algoritmo SL-*SOM*

O algoritmo SL-*SOM* (COSTA, 1999) consiste em encontrar grupos de dados através do particionamento da matriz-U de um *SOM* treinado. Basicamente este método utiliza o algoritmo de *watershed* (BEUCHER; LANTUÉJOUL, 1979) para segmentar a matriz-U e após a segmentação faz a rotulação dos neurônios associados a cada grupo encontrado com a segmentação.

O maior problema encontrado para a implementação desse algoritmo de agrupamento é encontrar os marcadores iniciais corretos para o algoritmo de *watershed*, pois o resultado do agrupamento depende da escolha desses marcadores. Para melhor segmentação esses marcadores devem ser os pixels correspondentes aos vales da matriz-U.

Seja a Matriz-U de um *SOM* treinado dada pela imagem *f*, considerando que [f_{min}, f_{max}] = [0, 255], ou seja, há 256 níveis de cinza na imagem *f*, Costa (1999) propõe que a escolha dos marcadores pode ser feita da seguinte forma:

1- Filtragem: a imagem f_1 é gerada removendo-se pequenos buracos na imagem f Pequenas depressões com área inferior a *T* pixels são eliminados.
2- Para *k=1,...,* $f_{\max}$, onde $f_{\max}$ é o nível de cinza máxima na imagem f_1 , crie as imagens binárias f_2^k correspondendo a conversões de f_1 usando *k* como valor de limiar.
3- Calcule o número de regiões conectadas de f_2^k, para cada valor de *k*, N_{rc}^k.
4- Procure no gráfico $k \times N_{\mathrm{rc}}^k$ a maior sequência contígua e constante de número de regiões conectadas N_{rc}^k, denotado por S_{max}.
5- A imagem de marcadores será a imagem f_2^j, onde *j* é o valor inicial da sequência S_{max}.

O passo 1 faz sumir pequenos ruídos da imagem para melhorar o processamento da mesma, visto que a matriz-U possui, em geral, muitas rugosidades. A operação de filtragem utilizada neste algoritmo foi a operação morfológica de erosão, usando um elemento estruturante de raio 1.

O passo 2 cria 256 imagens, onde cada imagem é binarizada utilizando os níveis de cinza, do mínimo ao máximo [0,255]. O processo de binarização faz com que os pixels menores que o limiar definido tome o valor 0 e os pixels com valores maiores que o limiar toma o valor 255, desta forma deixando a imagem somente com duas cores, 0 (preto) e 255 (branco). Após as imagens serem binarizadas é feita a extração dos componentes conectados das imagens, onde é retornado o número de objetos presentes nas imagens.

Depois do cálculo dos componentes conectados de cada imagem, é feito um gráfico do valor do limiar k *versus* o número de objetos encontrados na imagem, como na Figura 4.8.

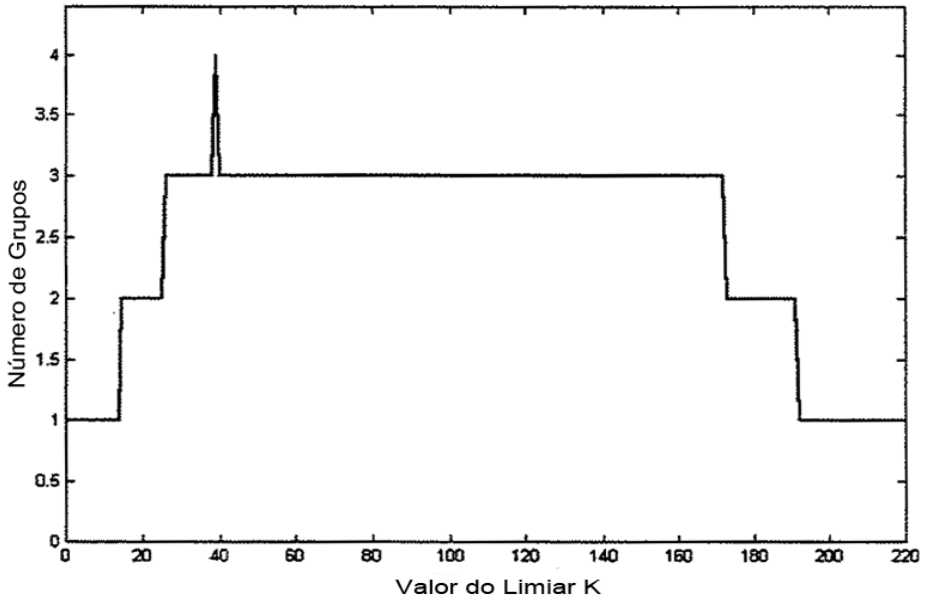


Figura 4.9: Gráfico do valor do limiar k *versus* o número de objetos encontrados

Fonte: Costa (1999).

Analisando a Figura 4.8, pode-se observar que a partir do limiar $k = 43$ até $k = 170$, ocorre maior sequência contígua e constante do mesmo valor de objetos encontrados nas imagens (igual a três). Dessa forma, três é o número de marcadores escolhidos para utilizar na segmentação da Matriz-U em questão. Segundo Costa (1999) a valor de *k* pode ser o valor inicial da maior sequência contigua do gráfico; neste exemplo, o valor de *k* escolhido seria $k = 43$.

Após a escolha dos marcadores de *watershed* dá-se o inicio do algoritmo, que consiste em inundar a imagem a partir dos marcadores escolhidos. No momento que duas inundações distintas se encontram é criada uma barreira entre elas, essas barreiras criadas são chamadas de linhas de *watershed*. A saída do algoritmo de *watershed* é uma imagem binária contendo as linhas de *watershed*, essas linhas vão delimitar os agrupamentos encontrados na imagem. A Figura 4.9 ilustra à esquerda os marcadores encontrados e à direita as linhas de *watershed* encontradas para esses marcadores.

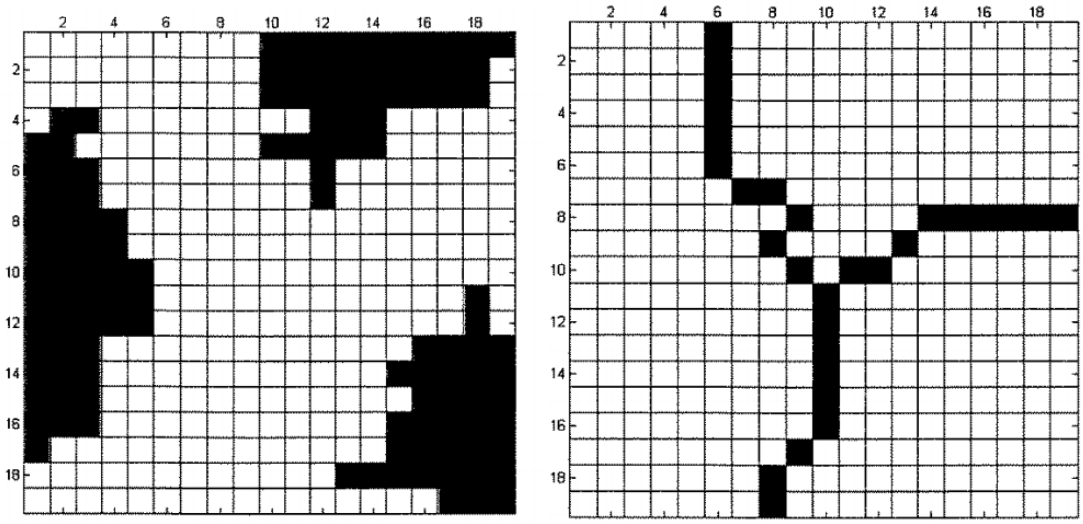


Figura 4.10: Algoritmo de *watershed* (a) Marcadores de *watershed*, (b) Linhas de *watershed*
Fonte: Costa (1999).

Através da imagem obtida do algoritmo de *watershed* contendo as linhas delimitadoras dos agrupamentos, os neurônios devem ser rotulados conforme os agrupamentos encontrados com a segmentação.

Dentre os vários algoritmos de *watershed* apresentados na literatura, o algoritmo escolhido para a implementação foi o algoritmo baseado na fila de prioridade de Beucher e Meyer (1990), que é um algoritmo relativamente simples de simulação de inundação, mas que consegue obter os mesmos resultados de outros algoritmos de *watershed* mais complexos. O procedimento deste algoritmo funciona em dois passos, sendo um a inicialização da fila e outro de trabalho. A inicialização da fila é feita por meio dos marcadores escolhidos, onde os mesmos são enfileirados conforme o valor de seu pixel em ordem crescente. O processamento trata de remover o pixel com maior prioridade da fila, aquele pixel que está no primeiro lugar da fila, que tem o menor custo, e então os seus vizinhos não rotulados são analisados, inserindo-os na fila com custo correspondente ao seu valor. Itera-se desta forma até que a fila esteja vazia, indicando que todos os pixels da imagem foram processados.

A seguir, o Algoritmo 1 apresenta o *watershed* baseado em fila de prioridade de Beucher e Meyer (1990 *apud* Klava, 2006).

Algoritmo 1: Watershed por Marcadores

```
Entrada:
f: imagem de entrada
L: imagem rotulada (entrada e saída)
Ws: imagem preenchida com 0, é marcada com 1 nas linhas de watershed

fila: fila de prioridade

  1. Inicialização
     Para cada pixel p com L(p) != 0, fila.insere(p, 0)
  2. Propagação
     enquanto não fila.vazia()
          p <- fila.remove()
     marcar p como permanente
     para cada vizinho q não-permanente de p
     se q não está rotulado
                    L(q) <- L(p)
     Fila.insere(q, f(q))
               Senão
                    Se L(q) != L(p) e ws(q) = 0 e ws(p) = 0
                         Ws(q) <- 1
```

Visto a dificuldade de se encontrar os marcadores de forma automática, também foi implementado um método para que o usuário, através da interface, escolha adequadamente as posições dos marcadores, deixando que o algoritmo *watershed* se encarregue de detectar, de forma ótima, as bordas entre os objetos.

A Figura 4.10 ilustra o método implementado ao selecionar a opção de escolha manual dos marcadores, sendo que uma nova janela é aberta com a matriz-U a ser segmentada. A escolha dos marcadores é feita pelo clique nos *pixels* desejados, onde os pixels escolhidos como marcadores ficam representados na cor preta. Como visto no Capítulo 3, os grupos na matriz-U são representados por *pixels* com valores mais baixos, ou seja, nos vales do mapa. Dessa forma, estes são os locais indicados para a escolha dos marcadores. Em uma imagem podem ser escolhidos vários marcadores, mas vale salientar que cada marcador escolhido irá gerar um grupo na matriz-U.

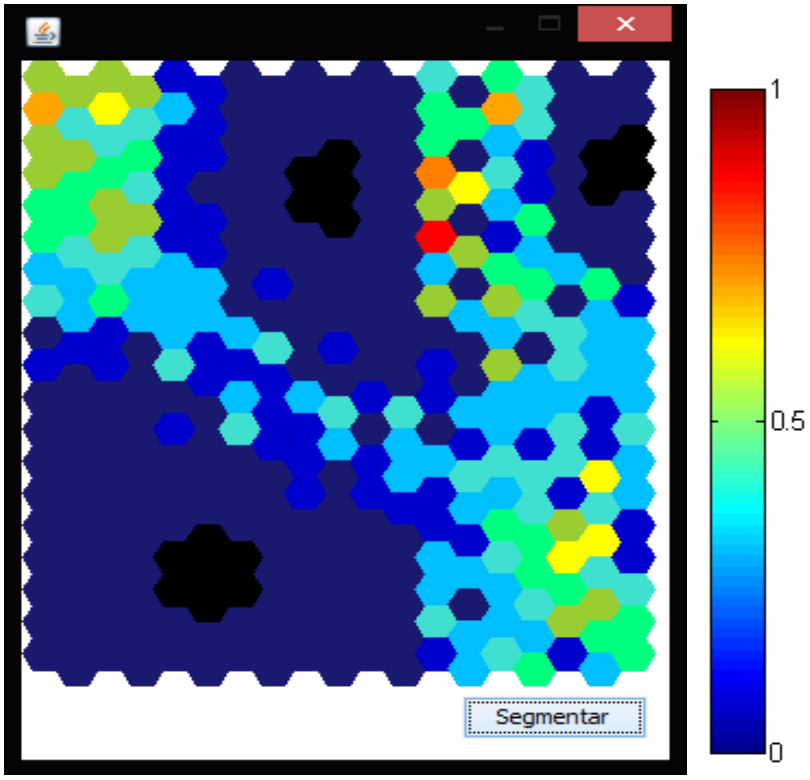


Figura 4.11: Selecionar os marcadores

A Figura 4.11 ilustra a tela de configuração dos marcadores de *watershed* para o algoritmo SL-*SOM*. Além de poder utilizar marcadores automáticos ou selecioná-los manualmente, ainda existe a possibilidade de definir o número de marcadores que se deseja, e de forma automática esses marcadores são encontrados.

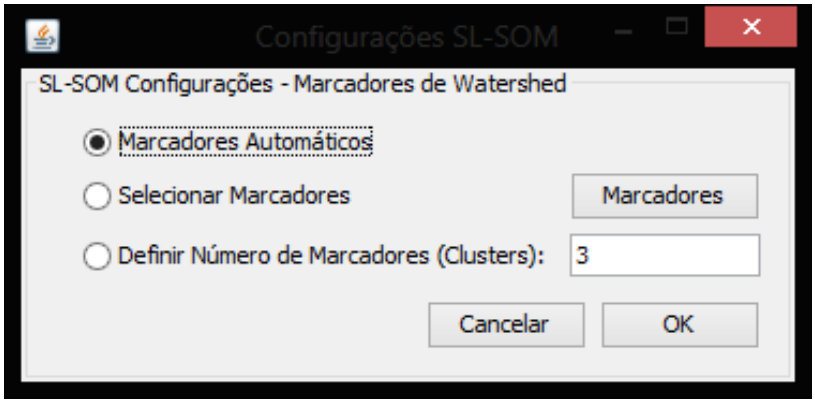


Figura 4.12: Tela da Ferramenta YADMT: Configuração do algoritmo SL-*SOM*

4.3.2.2. Agrupamento por Matriz de Densidade

O método de agrupamento baseado na Matriz de Densidade, como já visto, faz o agrupamento de um mapa treinado através da densidade dos neurônios, com o auxílio da Matriz de Densidade. A Figura 4.12 ilustra o resultado da execução do método implementado para um mapa *SOM* treinado, utilizando uma base de dados artificial de 150 instâncias onde existem três grupos que são linearmente separáveis. Ao final da execução do algoritmo de agrupamento são apresentados em tela os resultados obtidos, como ilustrado na Figura 4.12.

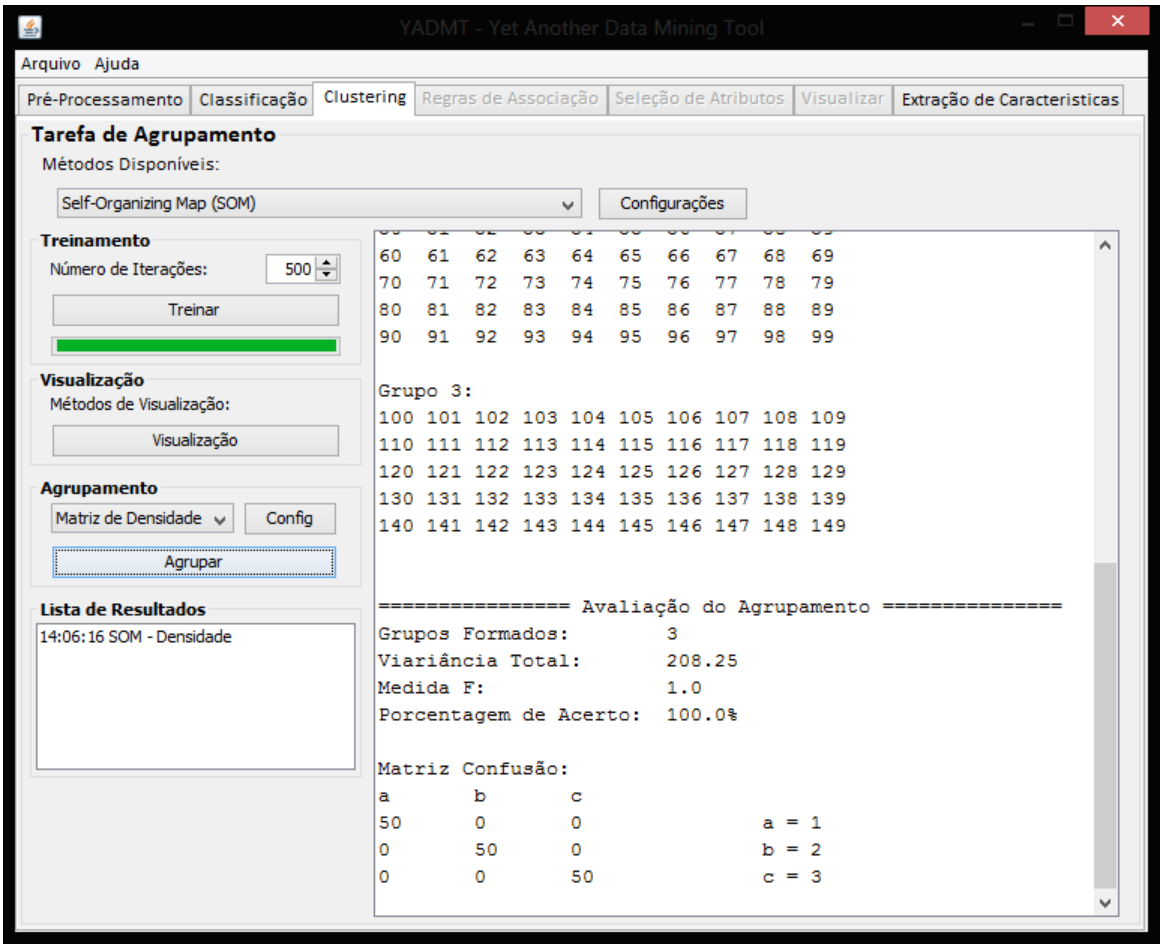


Figura 4.13: Tela da ferramenta YADMT – Execução do método de agrupamento

Neste algoritmo pode-se configurar o erro E, que corresponde à tolerância de distância entre as características dos neurônios, como ilustra a Figura 4.13. O desempenho deste algoritmo é totalmente dependente da configuração desse parâmetro.

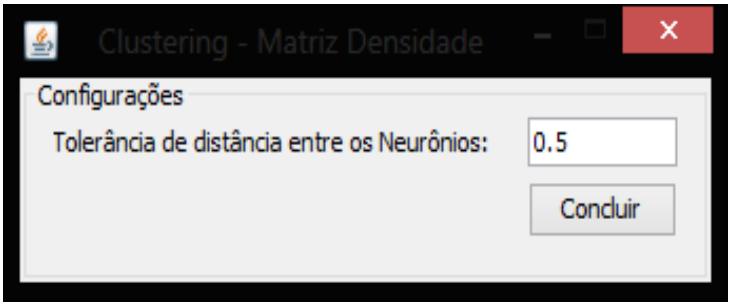


Figura 4.14:Tela da ferramenta YADMT – Configuração do erro E no método baseado na matriz de densidade

4.3.2.3. Metodologia de Vesanto & Alhoniemi

Como proposto por Vesanto e Alhoniemi (2000), o agrupamento deve ocorrer em duas fases, na primeira é realizado o treinamento do *SOM*, e após o treinamento, são aplicados métodos tradicionais de agrupamento nos vetores de pesos do *SOM* treinado.

Em Vesanto e Alhoniemi (2000), os autores utilizaram métodos hierárquicos e o algoritmo de K-médias para a realização do agrupamento a partir do SOM. Neste trabalho somente os métodos hierárquicos foram implementados (adicionando o Método de Ward aos utilizados pelos autores). Além disso, os autores utilizaram um procedimento automatizado para encontrar o particionamento e, neste trabalho, optou-se por informar o número de grupos que se deseja formar.

Os métodos de agrupamento disponibilizados para a segunda fase desta metodologia são os métodos hierárquicos apresentados no Capítulo 2 (Ligação Simples, Ligação Média, Ligação Completa e Método de Ward). O parâmetro de entrada dos algoritmos é o número de grupos que se deseja formar, como ilustra a Figura 4.14.

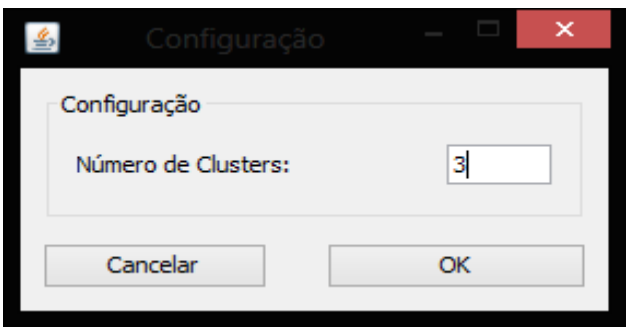


Figura 4.15: Tela da ferramenta YADMT – Parâmetro de entrada para os algoritmos de agrupamento hierárquicos

4.3.2.4. Metodologia Proposta

A extração e rotulação de componentes conectados são tarefas centrais em muitos dos sistemas de análise automática de imagens. O processo é realizado em uma imagem binária que possui valores 255 para o fundo da imagem e 0 para os objetos conectados, onde se deseja rotular ou encontrar os objetos conectados da imagem através de operações da morfologia matemática, que tem como base a teoria dos conjuntos.

O princípio básico da morfologia matemática consiste em extrair as informações relativas à geometria e à topologia de um conjunto desconhecido (uma imagem), pela transformação através de outro conjunto definido, chamado de elemento estruturante (MARQUES, 1999).

Para exemplificar, considere *Y* um componente conectado em um conjunto A e suponha-se que um ponto *p* de *Y* é conhecido. Então, a Equação 4.1 resulta em todos os elementos conectados de *Y*:

$$X_k = (X_{k-1} \oplus B) \cap A \qquad k = 1, 2, 3, \ldots \quad (4.1)$$

onde $X_0 = p$ e B é o elemento estruturante adequado, e $\oplus$ representa a operação de dilatação na morfologia matemática. O algoritmo converge quando $X_k - X_{k-1}$, e o valor final de X_k é atribuído a *Y*.

A Figura 4.16 ilustra um exemplo de extração de componentes conectados. Em (a) são mostrados o conjunto original A e o pixel de partida, indicado por 0. O elemento estruturante utilizado está ilustrado em (b). Já em (c) e (d) mostram respectivamente, os resultados após a primeira e segunda iterações. O resultado final (após 6 iterações) é mostrado em (e).

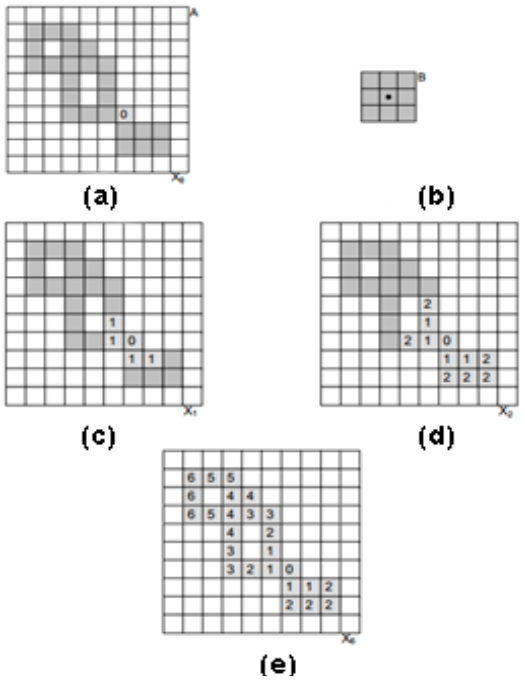


Figura 4.16: Extração de componentes conectados

Fonte: (MARQUES, 1999).

Por meio deste princípio, a idéia desta metodologia é utilizar o algoritmo de extração de componentes conectados para a rotulação dos grupos na matriz de densidade de um *SOM* treinado. A Seguir, a Figura 4.16 ilustra o diagrama para a metodologia proposta.

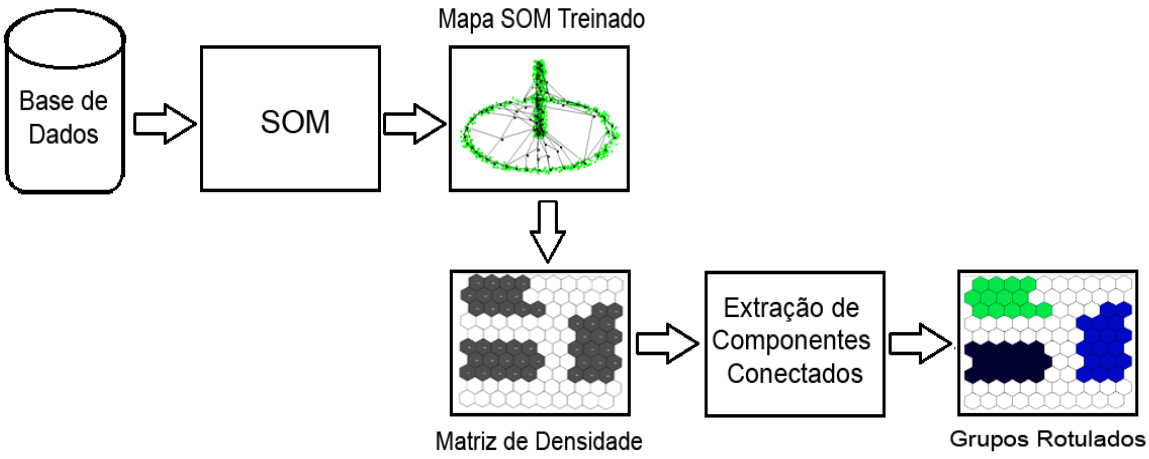


Figura 4.17: Diagrama da metodologia proposta

A metodologia proposta divide-se em quatro etapas:

1. Treinamento do mapa *SOM;*
2. Geração da matriz de densidade;
3. Aplicação do algoritmo de extração de componentes conectados a matriz de densidade;
4. Rotular os neurônios de acordo com os grupos encontrados na etapa 3.

Na etapa 3, considerando-se que a matriz de densidade é uma imagem binária (valor 0 para neurônios sem padrões atribuídos e 255 para neurônios com padrões atribuídos), o algoritmo de extração de componentes conectados percorre a imagem binária do canto superior esquerdo ao canto inferior direito, atribuindo um código a cada região (ou sequência) de *pixels* adjacentes. A busca se inicia pelo *pixel* da imagem que se localiza na primeira linha e primeira coluna. Ao localizar um *pixel* com valor 255, este é atribuído a um grupo. São identificados (e atribuídos ao mesmo grupo) todos os *pixels* vizinhos a este com valor 255. A cada novo *pixel* atribuído ao grupo, seus vizinhos são identificados e também atribuídos ao mesmo grupo, o que recursivamente faz preencher toda uma região conectada da imagem. O grupo fica completamente formado quando todos os vizinhos, de todos os *pixels* atribuídos ao grupo, forem também adicionados ao mesmo. A busca se reinicia, para obter novos grupos, até que todos os *pixels* com valor 255 forem atribuídos a um grupo.

Após esta etapa os neurônios devem ser rotulados conforme os grupos encontrados a partir do algoritmo de extração de componentes conectados.

A Figura 4.17 (a) ilustra a matriz de densidade para uma base artificial, linearmente separável, contendo 3 grupos. Após o treinamento do *SOM,* observando-se a matriz de densidade gerada, verifica-se que o *SOM* conseguiu separar espacialmente os três grupos. A Figura 4.17 (b) ilustra a matriz de densidade após a aplicação da metodologia proposta, com os neurônios já rotulados, onde foram coloridos conforme o grupo ao qual pertencem.

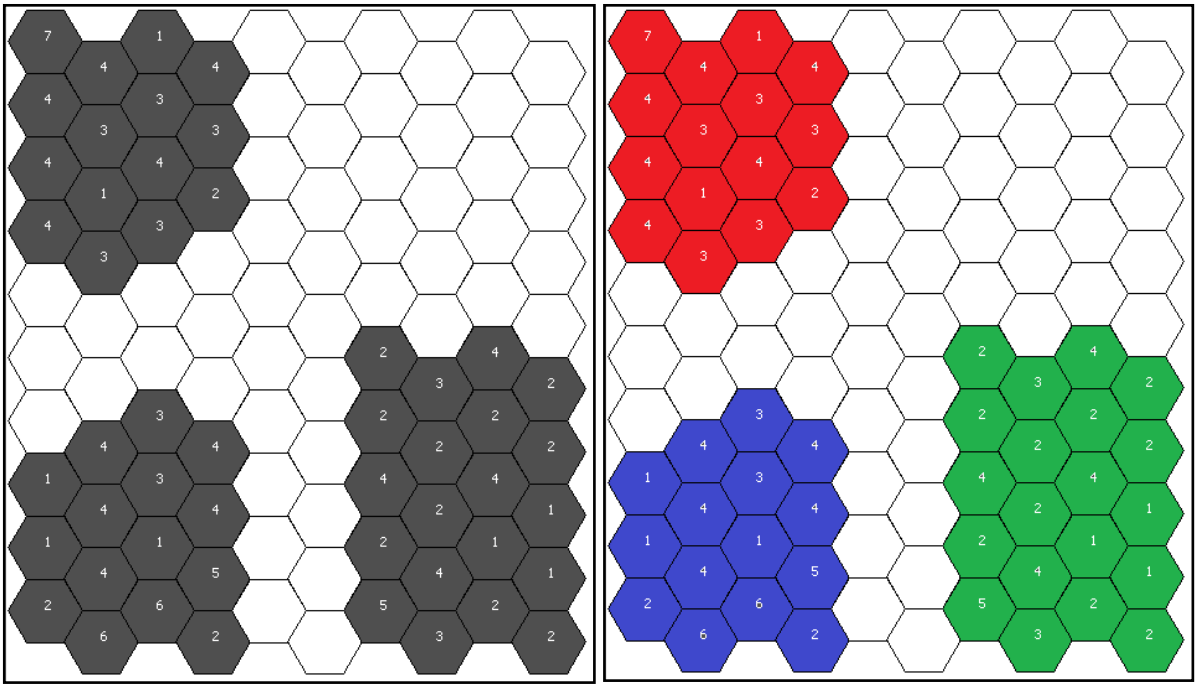


Figura 4.18: Metodologia de agrupamento proposto por matriz de densidade, (a) Matriz de Densidade de um *SOM* treinado, (b) Resultado da metodologia proposta com os grupos rotulados

Para obter melhores resultados, esta metodologia necessita que o número de neurônios presentes no mapa seja igual ou maior que o número de elementos da base, para que os grupos consigam ficar espacialmente separados na matriz de densidade, como ocorreu na ilustração da Figura 4.17.

4.4. Validação das metodologias implementadas

Para a validação das metodologias de agrupamentos implementadas, as mesmas foram comparadas a outros métodos de agrupamento presentes na ferramenta YADMT, algoritmo de K-médias e algoritmo baseado em Colônia de Formigas. Nesta avaliação foram utilizadas como métrica a medida F, o índice aleatório R e a porcentagem de agrupamento correto.

Alem disso, a metodologia implementada foi aplicada a uma base de dados real e inédita criada a partir dos dados fornecidos pelo IBGE sobre as cidades do Paraná. Os detalhes desses experimentos serão descritos no Capítulo 5.

4.5. Considerações Finais

Neste Capítulo foram apresentados detalhes de implementação do *SOM*, bem como métodos de visualização e metodologias de agrupamento incorporadas a ferramenta YADMT.

Como métodos de visualização foram implementados a matriz-U representada por superfície topológica, matriz-U bidimensional, Matriz de Densidade e também um método para visualização em tempo real do aprendizado do mapa *SOM*.

Para agrupamento a partir do *SOM* foram implementados ao todo cinco métodos de agrupamento: agrupamento por Matriz de Densidade, SL-*SOM*, 1D-*SOM*, Metodologia baseada em Vesanto e Alhoniemi (2000) e também foi proposto um método de agrupamento a partir da Matriz de Densidade de um *SOM* treinado, utilizando um algoritmo de extração de componentes conectados.

Capítulo 5

Avaliação Experimental

Para a realização da avaliação experimental foram utilizadas as bases de dados reais e públicas disponíveis no repositório da Universidade da Califórnia em Irvine, "*UCI Machine Learning Repository*" [1] : *Iris Plants Database*, *Dermatology Database, Pima Indians Diabetes, Libras Movemente Vehicle Silhouettes.*

Os métodos de agrupamento utilizados para estes experimentos foram: o agrupamento por Matriz de Densidade, 1D-*SOM*, metodologia de Vesanto e Alhoniemi (2000) (utilizando a Ligação Simples, Ligação Média e Ligação Completa e o Método de Ward), o algoritmo SL-*SOM* (1 – os marcadores são encontrados de forma automática e 2 – os marcadores são informados via interface), e a metodologia de agrupamento ora proposta.

5.1. Experimentos sobre Bases de Dados Rotuladas

Para a realização de todos os experimentos a seguinte parametrização básica foi utilizada: mapa bidimensional, inicialização linear, função de vizinhança gaussiana, função de aprendizagem exponencial, distância Euclidiana, aprendizagem inicial 0,01 e raio inicial igual a cinco. O número de neurônios do mapa utilizado para os experimentos foi igual ao número de elementos presentes na base de dados, em função de que a metodologia proposta obtém melhores resultados quando aplicada em mapas de tamanho igual ou maior o número de elementos da base. Esta mesma configuração também é utilizada em Kaski (1997). Para o método de agrupamento por matriz de densidade o erro *E*, que corresponde à tolerância de distância entre as características dos neurônios, para todos os experimentos foi fixado em 0,5.

[1] Disponível em: http://archive.ics.uci.edu/ml/datasets.html

Para todas as bases de dados os experimentos deu-se da seguinte forma: primeiramente foi realizado o treinamento do mapa *SOM* e, após o treinamento, sob este mesmo mapa foram aplicados os métodos de agrupamento. Os resultados dos métodos de agrupamento apresentados são as médias das medidas para agrupamentos realizados sob 10 mapas *SOM* treinados.

Os resultados dos agrupamentos utilizando a metodologia implementada foram comparados ao algoritmo de Colônias de Formigas e *K*-Médias presentes na ferramenta YADMT. As medidas de avaliação utilizadas foram: porcentagem de agrupamento correto e os índices externos: Aleatório (*R*) e Medida *F*. Para o cálculo da porcentagem de agrupamento correto foram utilizadas as informações dos rótulos de cada padrão, apresentadas previamente nas bases de dados, comparadas com os resultados dos agrupamentos formados ao término do algoritmo. . Os melhores resultados são apresentados em negrito.

5.1.1. Iris Plants

A base de dados *Iris* contém dados de três tipos de uma flor conhecida por Iris: Setosa, Versicolour e Virginica, onde a classe Setosa é linearmente separável das demais. Cada instância possui quatro atributos que trazem as informações de tamanho e espessura de sépala e tamanho e espessura de pétala, com as dimensões dadas em centímetros. Cada instância possui um atributo de classificação.

A base de dados é composta por 150 amostras, sem nenhum valor ausente para seus atributos. O Quadro 5.1 mostra a distribuição dos dados por classes.

Quadro 5.1: Distribuição da base dados Iris

Classe	**Frequência**
Setosa	50
Versicolour	50
Virgínica	50

A seguir o Quadro 5.2 apresenta os resultados obtidos com o algoritmo de Colônia de Formiga e *K*-médias.

Quadro 5.2: Resultados dos métodos Colônia de Formigas e K-médias para a base de dados Iris

Iris	**Formigas**	**K-médias**
F (quanto maior, melhor)	0,525	**0,886**
R (quanto maior, melhor)	0,533	**0,892**
Agrupamento correto (%)	41	**88**

A seguir o Quadro 5.3 apresenta os resultados obtidos com os algoritmos de agrupamento a partir do *SOM* para a base de dados Iris.

Quadro 5.3: Resultados dos métodos de agrupamento a partir do *SOM* para a base de dados Iris

Iris	**Metodologia de Vesanto e Alhoniemi**				**SL-*SOM***		**Densidade**	**1D-*SOM***	**Metodologia Proposta**
	Ligação Simples	**Ligação Média**	**Ligação Completa**	**Método de Ward**	**SL-*SOM* (1)**	**SL-*SOM* (2)**			
F	0,774	0,733	0,762	0,742	0,760	0767	0,812	**0,832**	0,49
R	0,776	0,747	0,798	0,787	0,775	0,764	0,817	**0,832**	0,33
Agrupamento correto (%)	66	65,33	78	76,66	70,66	78,2	82,66	**83,33**	34

Analisando os resultados obtidos podemos observar que dentre todos os métodos aplicados sob a base de dados Iris, o método *K*-médias obteve o melhor resultado, com 88% de acerto. Para os métodos de agrupamento a partir do *SOM*, o método 1D-*SOM* obteve o melhor resultado, com 83,33% de acerto e Medida *F* e Índice *R* igual a 0,832 .

A Figura 5.1 apresenta a matriz-U representada por superfície topológica, matriz-U bidimensional e a matriz de densidade para o mapa *SOM* com o menor erro topológico dentre as 10 execuções realizadas para a base de dados Iris. Nela pode-se observar que um grupo foi linearmente separado dos demais. Mas não é possível observar a divisão dos outros dois grupos.

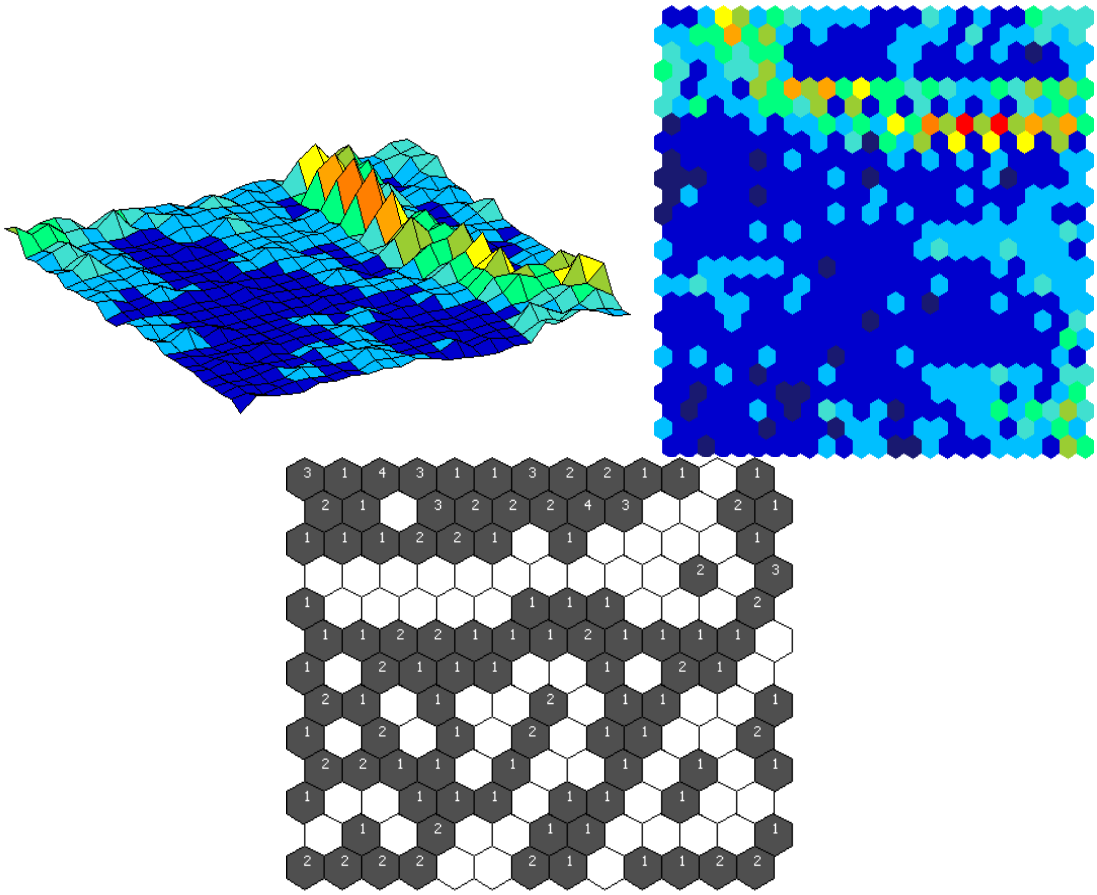


Figura 5.1: matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados Iris

5.1.2. Dermatology Database

A base de dados Dermatology é formada por 366 amostras, onde estão informações clínicas e histopatológicas de pacientes a respeito de seis tipos de doenças dermatológicas: Psoríase, Seborréia, *Lichen Planus*, Pitiríase Rosa, Dermatite Crônica e Pitiríase Vermelha. O Quadro 5.4 mostra a distribuição dos dados por classes.

Quadro 5.4: Distribuição de classes da base de dados Dermatology

Classe	Frequência
Psoríase	111
Seborréia	60
Lichen Planus	71
Pitiríase Rosa	48
Dermatite Crônica	48
Pitiríase Vermelha	20

A seguir o Quadro 5.5 apresenta os resultados obtidos com o algoritmo de Colônia de Formiga e *K*-médias.

Quadro 5.5: Resultados dos métodos Colônia de Formigas e K-médias para a base de dados Dermatology

Dermatology	**Formigas**	**K-médias**
F (quanto maior, melhor)	0,226	**0,714**
R (quanto maior, melhor)	0,227	**0,756**
Agrupamento correto (%)	20	**67**

A seguir o Quadro 5.6 apresenta os resultados obtidos com os algoritmos de agrupamento a partir do *SOM* para a base de dados Dermatology. Os melhores resultados são apresentados em negrito.

Quadro 5.6: Resultados dos métodos de agrupamento a partir do *SOM* para a base de dados Dermatology

Dermatology	**Metodologia de Vesanto e Alhoniemi**				**SL-*SOM***		**Densidade**	**1D-*SOM***	**Metodologia Proposta**
	Ligação Simples	**Ligação Média**	**Ligação Completa**	**Método de Ward**	**SL-*SOM* (1)**	**SL-*SOM* (2)**			
F	0,506	0,692	0,777	0,567	0,489	0,684	0,117	**0,806**	0,328
R	0,515	0,653	0,769	0,613	0,649	0,745	0,278	**0,758**	0,219
Agrupamento correto (%)	49,32	55,19	66,39	52,18	50,71	**73,74**	15,28	72,13	30,60

Analisando os resultados obtidos podemos observar que dentre todos os métodos aplicados sob a base de dados Dermatology, o método SL-*SOM* (2) obteve a maior classificação correta, com 73,74% de acerto. Enquanto o método 1D-*SOM* obteve o melhor resultado para a medida *F* e índice *R*, com 0,806 e 0,758, respectivamente.

A Figura 5.2 apresenta a matriz-U representada por superfície topológica, matriz-U bidimensional e a matriz de densidade para o mapa *SOM* com o menor erro topológico dentre as 10 execuções realizadas para a base de dados Dermatology. Analisando a Figura 5.2 podemos observar que os neurônios apresentaram uma grande dissimilaridade, é possível observar na matriz-U um número elevado de picos no mapa. E com isso não foi possível identificar os seis grupos presentes na base Dermatology.

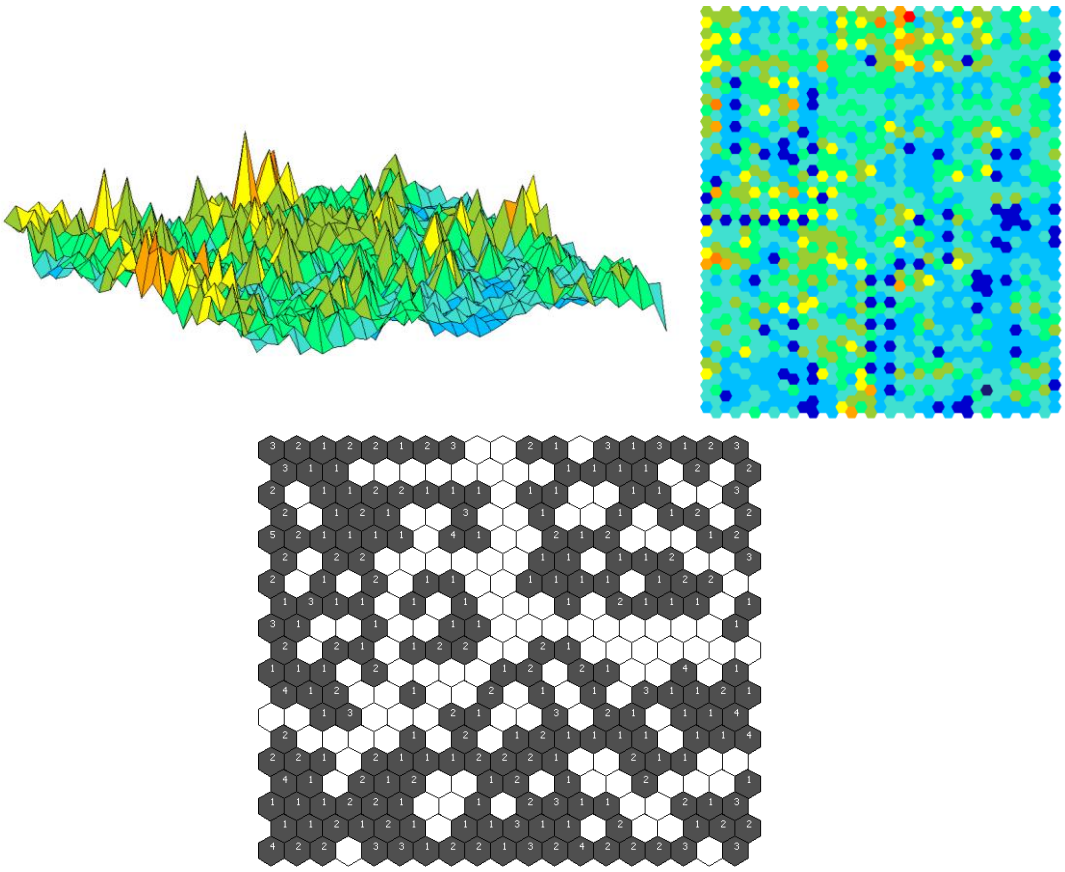


Figura 5.2: matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados Dermatology

5.1.3. Pima Indians Diabetes

A base *Pima Indians Diabetes* armazena dados de mulheres com diabetes com mais de 21 anos da tribo Pima do Arizona, EUA. Os registros contêm um atributo-classe que indica se a paciente é ou não diabética. A base apresenta 768 registros. O Quadro 5.4 mostra a distribuição dos dados por classes.

Quadro 5.7: Distribuição de classes da base de dados Pima

Classe	**Frequência**
Diabética	268
Não Diabética	500

A seguir o Quadro 5.8 apresenta os resultados obtidos com o algoritmo de Colônia de Formiga e *K*-médias.

Quadro 5.8:Resultados dos métodos Colônia de Formigas e K-médias para a base de dados Pima

Pima	**Formigas**	**K-médias**
F (quanto maior, melhor)	**0,693**	0,665
R (quanto maior, melhor)	**0,701**	0,621
Agrupamento correto (%)	65	**66**

A seguir o Quadro 5.9 apresenta os resultados obtidos com os algoritmos de agrupamento a partir do *SOM* para a base de dados Pima.

Quadro 5.9: Resultados dos métodos de agrupamento a partir do *SOM* para a base de dados Pima

Pima	**Metodologia de Vesanto e Alhoniemi**				**SL-*SOM***		**Densidade**	**1D-*SOM***	**Metodologia Proposta**
	Ligação Simples	**Ligação Média**	**Ligação Completa**	**Método de Ward**	**SL-*SOM* (1)**	**SL-*SOM* (2)**			
F	**0,694**	0,556	0,650	0,632	0,683	0,654	0,598	0,644	0,653
R	0,555	0,518	0,598	0,528	**0,643**	0,642	0,585	0,549	0,625
Agrupamento correto (%)	64,95	59,5	64,4	62,10	62,84	64,3	61,7	64,58	**66,10**

Analisando os resultados obtidos podemos observar que dentre todos os métodos aplicados sob a base de dados Pima, o agrupamento pela metodologia proposta obteve a maior classificação correta, com 66,10% de acerto. Para a Medida *F* o método de agrupamento baseado em Vesanto e Alhoniemi utilizando a Ligação Simples obteve o melhor resultado, com 0,694, e para o Índice *R*, o algoritmo baseado em Colônia de Formigas obteve o melhor resultado, com 0,701.

A Figura 5.3 apresenta a matriz-U representada por superfície topológica, matriz-U bidimensional e a matriz de densidade para o mapa *SOM* com o menor erro topológico dentre as 10 execuções realizadas para a base de dados Pima. Analisando a Figura 5.3 podemos observar que os neurônios do mapa são muito similares, formando assim um mapa plano, onde não foi possível identificar os dois grupos presentes na base Pima.

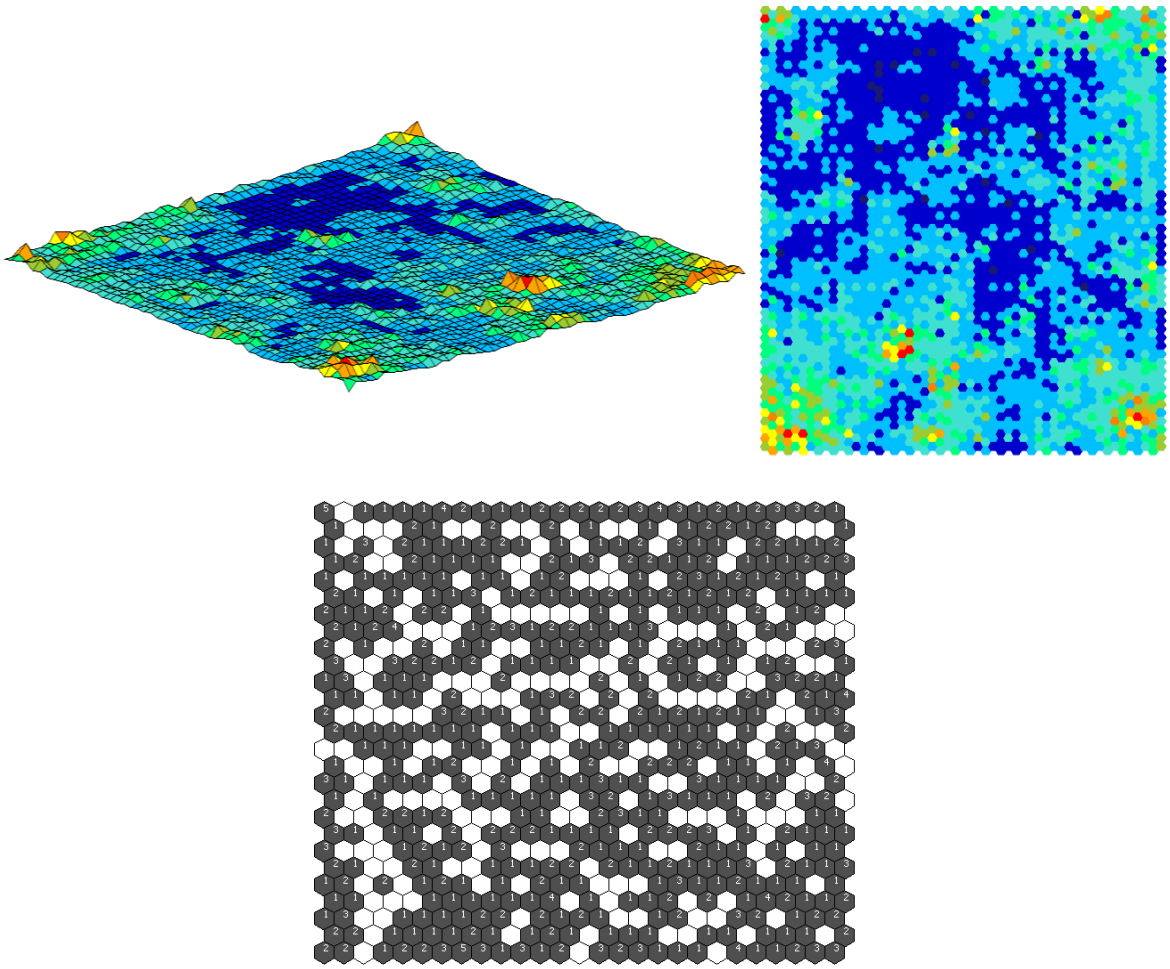


Figura 5.3: matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados Pima

5.1.4. Libras Movement

O conjunto de dados Libras Movement contém 15 classes com 24 padrões cada, onde cada classe faz referência a um tipo de movimento da mão em LIBRAS. O movimento de mão está representado como uma curva bidimensional realizada pela mão durante um período de tempo. As curvas foram obtidas a partir de vídeos dos movimentos da mão, de quatro pessoas diferentes. Cada vídeo corresponde a apenas um movimento da mão e tem cerca de 7 segundos. O Quadro 5.10 mostra a distribuição dos dados por classes.

Quadro 5.10: Distribuição de classes da base de dados Libras

Classe	Frequência
Curved swing	24
Horizontal swing	24
Vertical swing	24
Anti-clockwisearc	24
Clockwisearc	24
Circle	24
Horizontal straight-line	24
Vertical straight-line	24
Horizontal zigzag	24
Vertical zigzag	24
Horizontal wavy	24
Vertical wavy	24
Face-up curve	24
Face-down curve	24
Tremble	24

A seguir o Quadro 5.11 apresenta os resultados obtidos com o algoritmo de Colônia de Formiga e *K*-médias. Os melhores resultados são apresentados em negrito.

Quadro 5.11:Resultados dos métodos Colônia de Formigas e K-médias para a base de dados Libras

Libras	Formigas	K-médias
F (quanto maior, melhor)	0,183	**0,440**
R (quanto maior, melhor)	0,185	**0,446**
Agrupamento correto (%)	14	**33**

A seguir o Quadro 5.12 apresenta os resultados obtidos com os algoritmos de agrupamento a partir do *SOM* para a base de dados Libras.

Quadro 5.12: Resultados dos métodos de agrupamento a partir do *SOM* para a base de dados Libras

Libras	**Metodologia de Vesanto e Alhoniemi**				**SL-*SOM***		**Densidade**	**1D-*SOM***	**Metodologia Proposta**
	Ligação Simples	**Ligação Média**	**Ligação Completa**	**Método de Ward**	**SL-*SOM* (1)**	**SL-*SOM* (2)**			
F	0,225	0,410	0,428	0,462	0,219	0,449	0,296	**0,480**	0,247
R	0,219	0,398	0,412	0,459	0,226	**0,469**	0,327	0,460	0,219
Agrupamento correto (%)	18,83	37,2	38,77	**39,5**	21,38	38,27	27,7	36,77	19

Analisando os resultados obtidos podemos observar que dentre todos os métodos aplicados sob a base de dados Libras, o método de agrupamento baseado em Vesanto e Alhoniemi utilizando o método de Ward obteve a melhor classificação correta, com 39,5% de acerto. Para a Medida *F*, o algoritmo 1D-*SOM* obteve o melhor resultado, com o valor de 0,469. O algoritmo SL-*SOM* (2) obteve o melhor resultado para o índice aleatório *R*, com 0,469.

A Figura 5.4 apresenta a matriz-U representada por superfície topológica, matriz-U bidimensional e a matriz de densidade para o mapa *SOM* com o menor erro topológico dentre as 10 execuções realizadas para a base de dados Libras. Analisando a Figura 5.4 podemos observar que os neurônios apresentaram uma grande dissimilaridade, é possível observar na matriz-U um número elevado de picos no mapa. E com isso não foi possível identificar os quinze grupos presentes na base Libras.

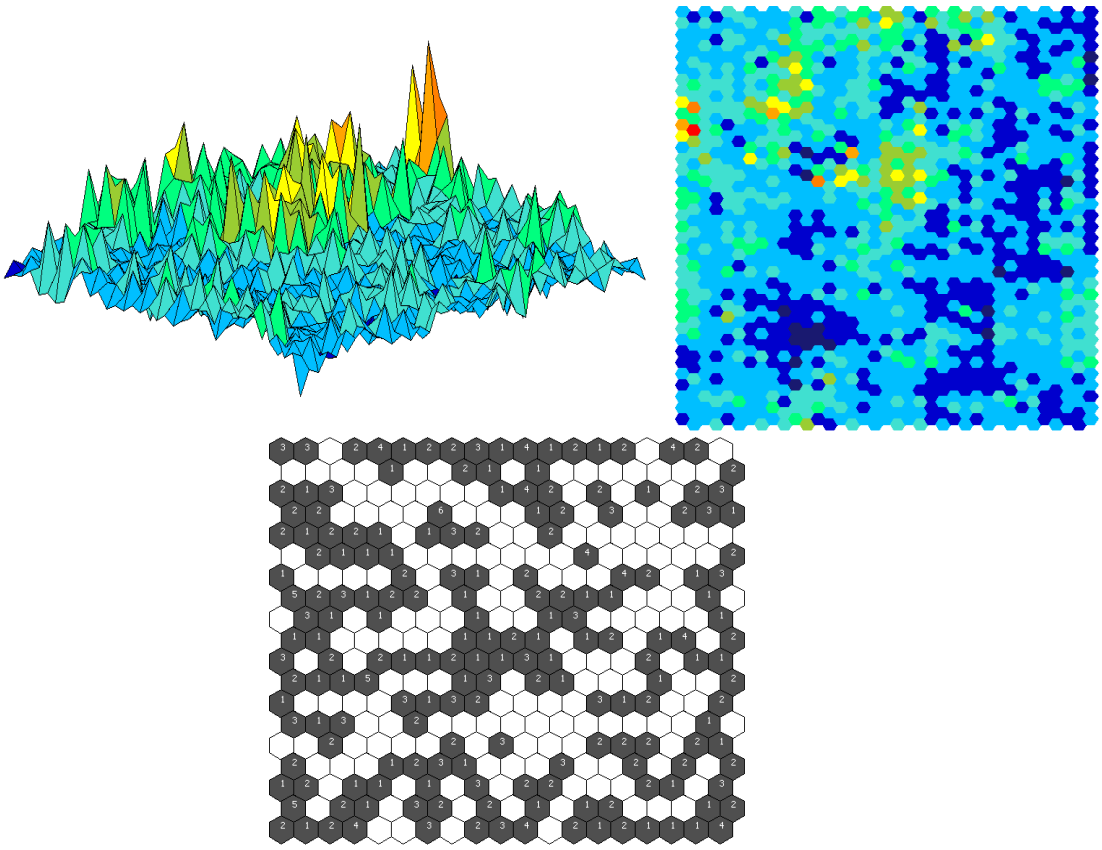


Figura 5.4:matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados Libras

5.1.5. Vehicle Silhouettes

Formada por 946 amostras divididas em 4 classes, com 18 atributos. Cada elemento da base representa um veículo, onde seus atributos são valores de suas silhuetas, retirados de ângulos diferentes. O Quadro 5.13 mostra a distribuição dos dados por classes.

Quadro 5.13: Distribuição de classes da base de dados Vehicle

Classe	Frequência
Opel	240
Saab	240
Bus	240
Van	226

A seguir o Quadro 5.14 apresenta os resultados obtidos com o algoritmo de Colônia de Formiga e *K*-médias. Os melhores resultados são apresentados em negrito.

Quadro 5.14: Resultados dos métodos Colônia de Formigas e K-médias para a base de dados Vehicle

Vehicle	Formigas	K-means
F (quanto maior, melhor)	0,316	**0,428**
R (quanto maior, melhor)	0,319	**0,420**
Agrupamento correto (%)	29	**33**

A seguir o Quadro 5.15 apresenta os resultados obtidos com os algoritmos de agrupamento a partir do *SOM* para a base de dados Vehicle.

Quadro 5.15: Resultados dos métodos de agrupamento a partir do *SOM* para a base de dados Vehicle

Vehicle	Metodologia de Vesanto e Alhoniemi				SL-*SOM*		Densidade	1D-*SOM*	Metodologia Proposta
	Ligação Simples	Ligação Média	Ligação Completa	Método de Ward	SL-*SOM* (1)	SL-*SOM* (2)			
F	0,339	0,383	0,386	0,318	0,320	0,409	0,423	**0,452**	0,323
R	0,279	0,400	0,376	0,324	0,269	0,394	**0,451**	0,427	0,281
Agrupamento correto (%)	27,12	28,54	37,82	28,05	26,76	38,22	36,73	**42,48**	27,12

Analisando os resultados obtidos podemos observar que dentre todos os métodos aplicados sob a base de dados Libras, o método 1D-*SOM* obteve o melhor resultado, com 42,48% de acerto, e também obteve a melhor Medida *F*, com 0,452. Para o Índice *R* o método de agrupamento por matriz de densidade obteve o melhor resultado, com 0,451.

A Figura 5.5 apresenta a matriz-U representada por superfície topológica, matriz-U bidimensional e a matriz de densidade para o mapa *SOM* com o menor erro topológico dentre as 10 execuções realizadas para a base de dados Vehicle. Analisando a Figura 5.5 podemos observar que os neurônios do mapa são muito similares, formando assim um mapa plano, onde não foi possível identificar os quatro grupos presentes na base Vehicle.

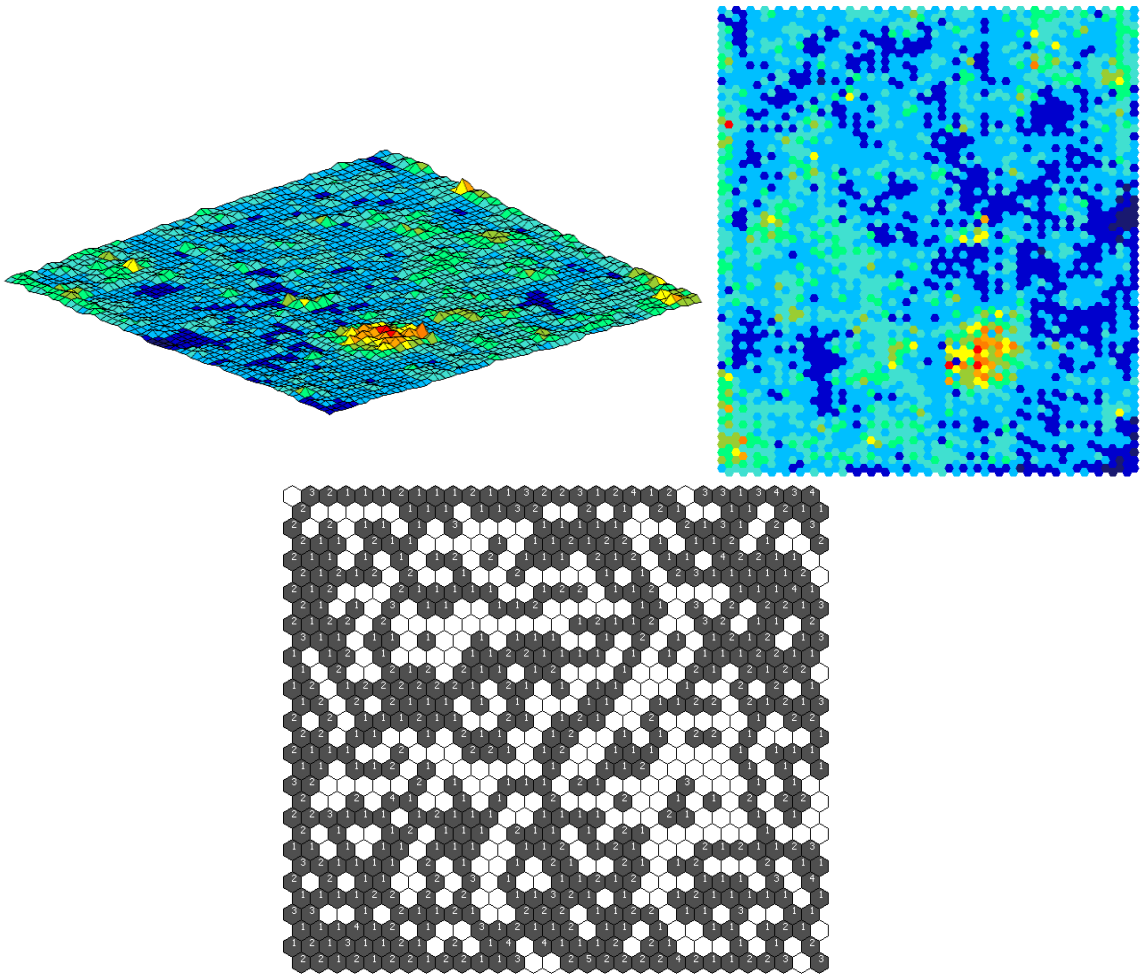


Figura 5.5: matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados Vehicle

5.2. Aplicação da metodologia a base de dados real

A base de dados utilizada neste trabalho foi extraída do site do Instituto Brasileiro de Geográfica e Estatística (IBGE) [2]. Trata-se da base de dados "Síntese" que contém informações sobre estado do Paraná do ano de 2010. A base de dados é formada por 399 objetos, cada um representando uma cidade do estado do Paraná, contendo 18 atributos cada. Os atributos da base de dados Síntese são referentes a educação, trabalho, renda rural e renda urbana, entre outros. A base de dados Síntese original contém um atributo referente a área territorial de cada município, este atributo foi removido para a realização deste experimento, para não haver algum tipo de influência do tamanho do município no resultado.

Para a extração dos dados do site foi implementado um pequeno aplicativo (Figura 5.6) para viabilizar a aquisição e a seleção dos dados no site do IBGE e também para a geração de um arquivo contendo a base de dados, do tipo ARFF, compatível com a ferramenta YADMT.

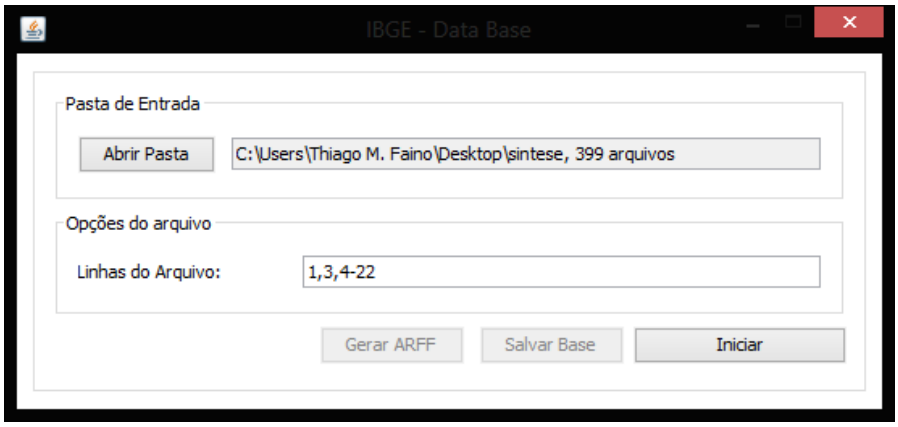


Figura 5.6: Aplicativo desenvolvido para a extração da base de dados do IBGE

Após a extração dos dados, os mesmos necessitavam de pré-processamento, pois a desproporcionalidade referente ao número total de habitantes entre as cidades era grande, então para os atributos referentes ao número de população da base foi aplicada a proporção pela população total de cada município.

A parametrização do *SOM* deu-se de forma empírica, foram feitos diversos ajustes nos parâmetros até que se obtivesse o menor erro topológico e erro de quantização possível. Através da realização de vários testes, a parametrização que obteve o menor erro na formação do mapa *SOM* foi a seguinte:

[2]Disponível em: http://www.ibge.gov.br/cidadesat/topwindow.htm?1

- Mapa 20 × 20;
- Aprendizagem inicial igual a 0,3;
- Raio inicial igual a 4;
- Distância euclidiana;
- Função de atualização exponencial.

Utilizando essa configuração para o treinamento do *SOM* o erro de quantização foi igual a 0,0062 e o erro topológico foi igual a 0,087.

A Figura 5.7 mostra os métodos de visualização de um dos resultados para o mapa auto-organizável a qual a base de dados foi aplicada utilizando os parâmetros citados anteriormente.

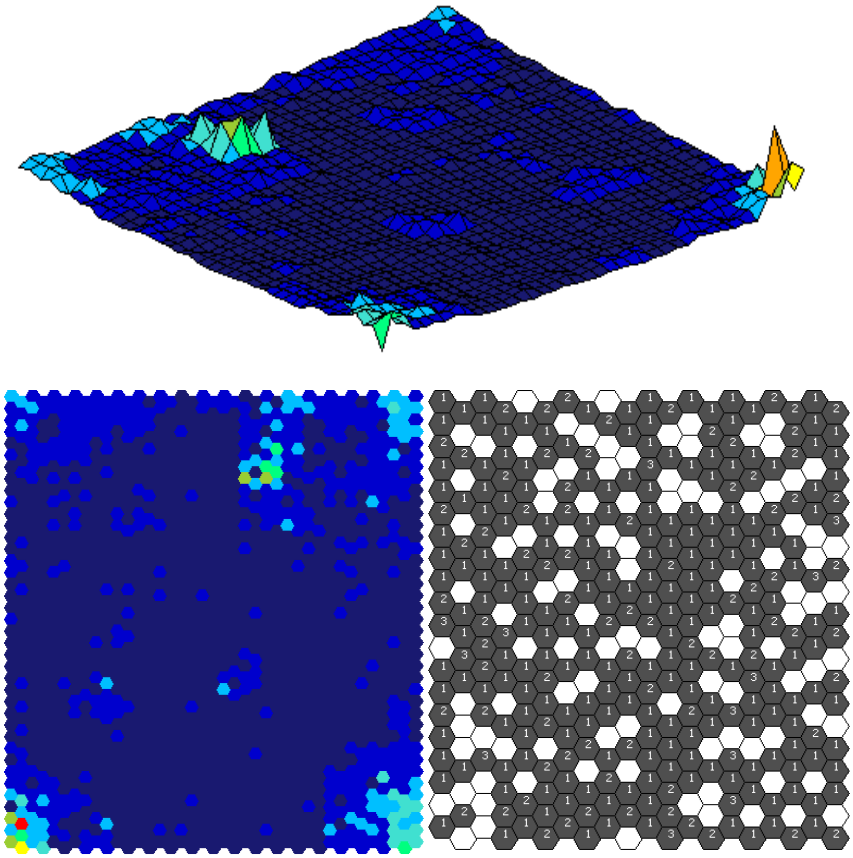


Figura 5.7: matriz-U representada por superfície topológica, matriz-U bidimensional e matriz de densidade de um *SOM* treinado pra a base de dados do IBGE

Fazendo a análise visual da Matriz-U e da Matriz de Densidade para a base de dados do IBGE pode-se observar a dificuldade em encontrar os agrupamentos, pelo fato de não

apresentar montanhas e vales bem definidos na Matriz-U e nem a separação dos grupos na Matriz de Densidade. Deste modo os algoritmos de agrupamento que utilizam a matriz-U (SL-*SOM*) e a matriz de densidade (método baseado na Matriz de Densidade e método Proposto) não conseguiram obter um resultado satisfatório, formando, na maioria das vezes, somente um agrupamento.

Sendo assim, a melhor metodologia para a realização do agrupamento foi a de Vesanto e Alhoniemi (2000), pois não utiliza a Matriz-U ou a Matriz de Densidade, e sim os vetores de pesos dos neurônios. Ao realizar os experimentos com os métodos implementados para a metodologia de Vesanto e Alhoniemi (2000), Ligação Simples, Ligação Média e Ligação Completa, a tendência foi de agrupar a grande maioria das cidades em um único grupo, e o restante dos grupos foram formados por poucos elementos. Quando realizado experimentos com o método de Ward, os grupos obtidos tiveram padrões melhor distribuídos, formando alguns grupos com número de elementos parecidos, o que é uma característica do método de Ward, deste modo o método de Ward foi escolhido para a aplicação na base de dados do IBGE.

Como o método de Ward não encontra o número de grupos de forma automática, é necessário informar o número de agrupamentos desejados. Este número de grupos foi escolhido de forma empírica, após experimentos realizados, o número de grupos igual a 10 apresentou a melhor distribuição das cidades nos grupos.

A seguir o Quadro 5.16 apresenta o resultado dos agrupamentos obtidos aplicando a metodologia de Vesanto e Alhoniemi (2000) com o método de Ward na base de dados do IBGE.

Quadro 5.16: Distribuição das cidades em grupos para a base de dados do IBGE

Grupo	Número de Cidades
1	144
2	80
3	41
4	34
5	30
6	27
7	17
8	13
9	12
10	1

O Grupo 1 apresentou menor número de matrículas no ensino médio e segundo menor número de pessoas que frequentavam creche ou escola dentre todos os grupos. O grupo é formado por cidades como: Antonina, Cafelândia, Campo Bonito, Catanduvas, Céu Azul, São Miguel do Iguaçu, Santa Izabel do Oeste, Vera Cruz do Oeste, Paranavaí, entre outras

O Grupo 2 não apresentou uma característica específica. O grupo é formado por cidades como: Apucarana, Assis Chateaubriand, Boa Vista da Aparecida, Campo Mourão, Corbélia, Guaíra, Ibema, Laranjeiras do Sul, Matelândia, entre outras.

O Grupo 3 apresentou o segundo menor número de estabelecimentos do SUS e apresentou o segundo maior número em residências, pessoas ocupadas, PIB, pessoas alfabetizadas, rendimento médio e per capta rural e rendimento médio e per capta urbano dentre todos os grupos. O grupo é formado por cidades como: Campo Largo, Cascavel, Foz do Iguaçu, Londrina, Maringá, Paranaguá, Pato Branco, Ponta Grossa, Toledo, entre outras.

O Grupo 4 apresentou o segundo maior número de matrículas no ensino médio e o menor número de pessoas da religião espírita dentre todos os grupos. O grupo é formado por cidades como Castro, Chopinzinho, Diamante do Oeste, Entre Rios do Oeste, General Carneiro, Maripá, Mercedes, entre outras.

O Grupo 5 apresentou o segundo menor número de matrículas no ensino fundamental e médio, o menor número de pessoas que frequentaram creche ou escola, o segundo maior número de católicos e o maior rendimento per capta rural dentre todos os grupos. O grupo é formado por cidades como Bom Sucesso, Califórnia, Cianorte, Jesuítas, Nova Aurora, Quatro Pontes, Terra Roxa, Tupãssi, Umuarama, entre outras.

O Grupo 6 apresentou o segundo maior número de matrículas no ensino fundamental, o menor PIB, o segundo menor número de pessoas alfabetizadas e o segundo menor rendimento per capta urbano dentre todos os grupos. O grupo é formado por cidades como: Altamira do Paraná, Cantagalo, Matinhos, Morretes, Palmas, Ramilândia, Santa Terezinha de Itaipu, entre outras.

O Grupo 7 apresentou o maior número de matrículas no ensino fundamental, o segundo menor número de pessoas ocupadas, o menor número de pessoas alfabetizadas, o segundo menor rendimento per capta rural e menor rendimento per capta urbano dentre todos os grupos. O grupo é formado por cidades como: Diamante do Sul, Espigão Alto do Iguaçu, Guaraniaçu, Nova Laranjeiras, Salgado Filho, entre outras.

O Grupo 8 apresentou o maior número de estabelecimentos do SUS, o menor número de residências, o maior rendimento médio rural e o menor rendimento médio urbano dentre todos os grupos. O grupo é formado por cidades como: Cafezal do Sul, Cruzeiro do Iguaçu, Itaúna do Sul, Mirador, São Jorge do Oeste, Sulina, Vere, entre outras.

O Grupo 9 apresentou o segundo maior número de estabelecimentos do SUS, o maior número de matrículas no ensino médio, o menor número de pessoas ocupadas, o segundo menor PIB e o segundo maior rendimento médio rural dentre todos os grupos. O grupo é formado por cidades como: Adrianópolis, Enéas Marques, Fernandes Pinheiro, Guaraquecaba, Inácio Martins, Manfrinópolis, Rio Bom, entre outras.

O Grupo 10 apresentou o menor número de estabelecimentos do SUS, o menor número de matrículas no ensino fundamental, o maior número de residências, pessoas ocupadas, maior PIB, maior número de pessoas alfabetizadas, menor rendimento médio e per capta rural e maior rendimento médio e per capta urbano dentre todos os grupos. O grupo é formado apenas por Curitiba.

A Figura 5.8 ilustra o mapa do Paraná dividido por municípios, onde cada município foi colorido com a respectiva cor de seu grupo, como representado na legenda.

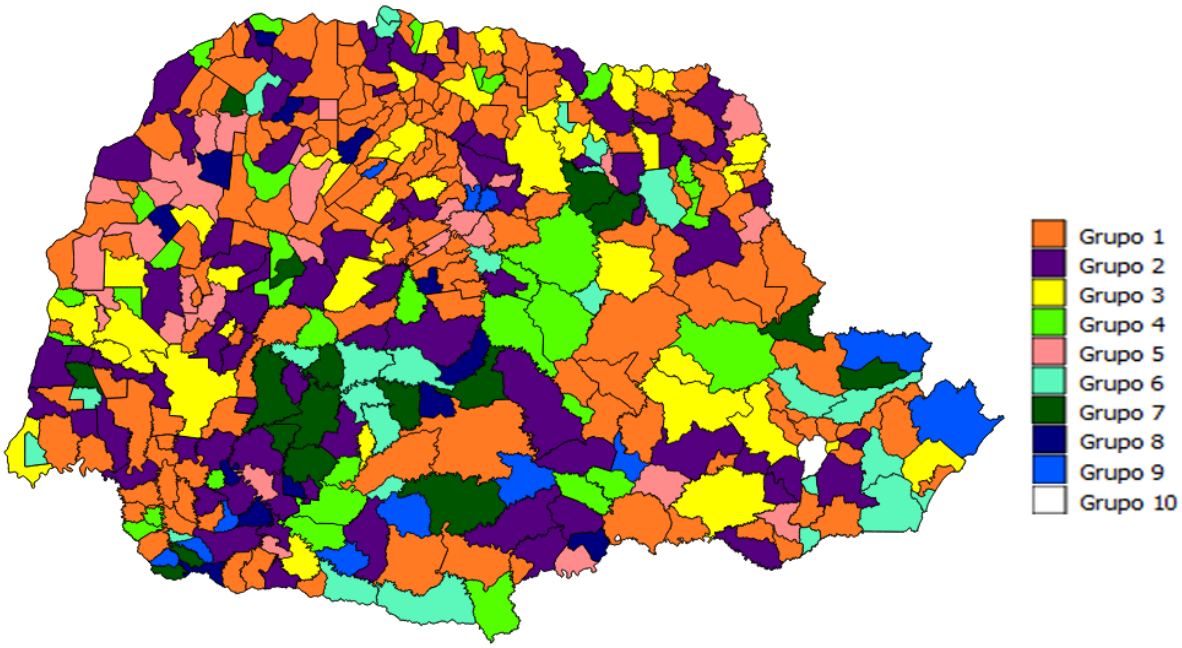


Figura 5.8: Mapa do Paraná com os grupos formados

5.3. Considerações Finais

Neste capítulo, foi apresentada uma avaliação experimental da metodologia de agrupamento a partir do *SOM* implementada, com o intuito de mostrar sua efetividade comparando-a ao método de agrupamento baseado em Colônia de Formigas e K-médias presentes na ferramenta YADMT.

Analisando os resultados obtidos com os experimentos podemos observar que a metodologia de agrupamento de dados a partir do *SOM* apresentou um resultado satisfatório, pois em quatro das cinco bases de dados aplicadas o *SOM* obteve os melhores resultados.

Dentre as cinco bases de dados experimentadas, a metodologia proposta obteve o melhor resultado para uma das bases. Cabe salientar que a metodologia proposta apresenta melhores resultados para mapas de tamanhos maiores, onde na matriz de densidade os grupos tendem a se separar de melhor forma.

Também foi observado com os experimentos que o algoritmo SL-*SOM* com os marcadores de *watershed* informados pelo usuário obteve melhores resultados do que quando os marcadores de *watershed* são encontrados de forma automática, o que mostra a dificuldade encontrada para o algoritmo escolher de forma automática os marcadores corretos.

Ao aplicar a metodologia à base de dados do IBGE, houve certa dificuldade na análise dos dados por se tratar de uma base inédita que ainda não foi rotulada. Mesmo os experimentos sendo realizados de forma empírica, com os resultados obtidos através da metodologia a partir do *SOM*, foram possíveis encontrar justificativas para a formação dos grupos. Porém os resultados mostram que a base de dados criada pode ser melhor analisada e melhores resultados podem ser obtidos.

Capítulo 6

Considerações Finais

No processo *KDD* técnicas baseadas em Redes Neurais Artificiais (RNA) vem se destacando, quando utilizadas para a tarefa de agrupamento de dados. É neste contexto que este trabalho propôs o estudo e implementação de uma metodologia de agrupamento de dados a partir do *SOM*, que pertence a uma classe de Redes Neurais Artificiais onde a aprendizagem é não supervisionada.

Após a breve descrição do processo KDD, apresentou-se uma introdução ao contexto do trabalho e uma visão geral da tarefa de agrupamento de dados, bem como as principais categorias e algoritmos existentes para esta tarefa.

Como foco deste trabalho, também foi apresentado uma introdução aos Mapas Auto Organizáveis, com questões sobre métricas utilizadas, parametrizações e como ocorre a análise de agrupamentos via *SOM*¸ por métodos de visualização e por métodos de agrupamento de dados. Após a descrição do *SOM*, foram apresentadas as metodologias e os detalhes das implementações realizadas no módulo de Agrupamento da ferramenta YADMT.

Para a avaliação das metodologias implementadas foram realizados experimentos utilizando cinco bases de dados reais e públicas comparando os resultados com o método de agrupamento Colônia de Formigas e K-médias, que também estão presentes na ferramenta YADMT. Os resultados obtidos foram satisfatórios, onde nenhuma metodologia de agrupamento a partir do *SOM* foi superior para todas as bases de dados experimentadas. Das cinco bases de dados experimentadas, o *SOM* obteve melhor resultado em quatro.

Como contribuições para a ferramenta YADMT foram implementados os métodos de visualização: Matriz-U por representação topológica, Matriz-U bidimensional e também a Matriz de Densidade, além de um método de visualização proposto do treinamento do *SOM*

em tempo real. Para métodos de agrupamento foram implementados o algoritmo SL-*SOM*, 1D-*SOM*, metodologia de Vesanto e Alhoniemi (2000), agrupamento por matriz de densidade, e ainda foi proposto um método de agrupamento utilizando o algoritmo de extração de componentes conectados sobre a matriz de densidade.

6.1. Trabalhos Futuros

A partir dos estudos realizados durante a elaboração deste trabalho, como trabalhos futuros propõem-se:

- Implementação de um método para escolher de forma automática as melhores parametrizações do *SOM* para cada base de dados informada;
- Escolha automática da melhor metodologia para agrupamento de dados a partir do *SOM;*
- Implementação de outras metodologias de agrupamento a partir do *SOM* a exemplo da proposta por Boscarioli (2008);
- Investigar outras bases de dados reais.

Referências

BENFATTI, *E.* W.; BONIFACIO, F. N.; GIRARDELLO, A. D.; BOSCARIOLI, C. **Descrição da Arquitetura e Projeto da Ferramenta YADMT - Yet Another Data Mining Tool**. Relatório Técnico nº 01 do Curso de Ciência da Computação, UNIOESTE, Campus de Cascavel, 2010.

BEUCHER, S.; LANTUÉJOUL, C. **Use of whatersheds in contour detection**. In: *Proceedings of the International Workshop Image Processing, Real-Time Edge and Motion Detection/Estimation*, CCETT/INSA/IRISA, Renees, França, Setembro 1979.Disponível em: <http://cmm.ensmp.fr/~beucher/publi/watershed.pdf>. Acesso em: 20 Jun. 2013

BEUCHER, S; MEYER, F. **Morphological segmentation.**Journal of Visual Communication and Image Represetation, 1(1): 21-46, Setembro 1990;

BOSCARIOLI, C. **Análise de agrupamentos baseada na topologia dos dados e em mapas auto-organizáveis**. 2008. Tese (Doutorado em Engenharia Elétrica) – Escola Politécnica, Universidade de São Paulo, São Paulo, 2008.

CASTRO, F.C.C.; CASTRO, M.C.F. *Redes Neurais Artificiais.* Porto Alegre: PUCRS, 2001. Paginação irregular. Apostila para fins didáticos. Disponível em: <http://diana.ee.pucrs.br/~decastro/RNA_hp/RNA.html>. Acesso em: 03 Fev. 2013.

CASTRO, L.N. **Análise e Síntese de Estratégias de Aprendizado para Redes Neurais Artificiais**. Campinas: FEEC, UNICAMP, 1998. Dissertação de Mestrado –Faculdade de Engenharia Elétrica e de Computação, Universidade Estadual de Campinas, São Paulo, 1999.

COSTA, J. A. F. **Classificação automática e análise de dados por redes neurais auto-organizáveis**. 1999. Tese (Doutorado em Engenharia Elétrica) –Faculdade de Engenharia Elétrica e Computação, Universidade Estadual de Campinas, São Paulo, 1999.

DINIZ, C. A. R.; NETO, F.L. **Data Mining: uma introdução.**São Paulo: ABE, 2000.

FAUSETT, L. **Fundamentals of Neural Networks – Architectures, Algorithms, and Applications**. New Jersey: Prentice Hall, 1994.

FAYYAD, U. M.; PIATETSKY-SHAPIRO, G.; SMYTH, P.; UTHRUSAMY, R. **Advances in knowledge Discovery & Data Mining**. Califórnia: AAAI/MIT, 1996.

FRANCISCO, C.A.C. **REDE DE KOHONEN: Uma ferramenta no estudo das relações tróficas entre espécies de peixes.**Dissertação (Mestrado) - Programa de Pós - Graduação em Métodos Numéricos em Engenharia, Universidade Federal do Paraná, Curitiba, 2004.

HAIR JR, J.F.; ANDERSON, R.E.; TATHAM, R.L.; BLACK, W.C. **Análise Multivariada de Dados**. Tradução de: SANTANNA, A. S.; CHAVES NETO, A. Porto Alegre: Bookman, 2005.

HAYKIN, S. **Redes neurais: princípios e prática**. Tradução: Paulo Martins Engel. Porto Alegre: Bookman, 2001.

JAIN A. K., MURTY M. N., FLYNN P. J. **Data clustering: a review**.ACM Computing Surveys. v. 31, n. 3, 1999.

JOHNSON, R.A.; WICHERN, D.W. **Applied Multivariate Statistical Analysis.** Fourth Edition. New Jersey: Prentice Hall, 1998.

KASKI, S. **Data Exploration using Self-Organizing Maps.** Acta Polytechnica Scandinavica, Mathematics, Computing and Management in Engineering Series nº 82. Dr. Tech Thesis, Helsinki University of Tecnology, Finland, 1997.7

KLAVA, B. **Ferramenta interativa para segmentação de imagens digitais.** 2006. Trabalho de Formatura Supervisionado. Instituto de Matemática e Estatística, Universidade de São Paulo, São Paulo.

KOHONEN, T. **Self-Organization and Associative Memory**. 3a. ed. Nova York, USA: Spring-Verlag, 1989.

KOHONEN, T. **Self-Organization**.3a. ed. Nova York, USA: Spring-Verlag, 2001.

MACQUEEN, J. B. ***SOM* e Methods for classification and Analysis of Multivariate Observations**.Proceedings of 5th Berkeley Symposium on Mathematical Statistics and Probability. University of California Press.pp.281–297.1967.

MARQUES, O. F.; VIEIRA, H. N. Processamento Digital de Imagens, Rio de Janeiro: Brasport, 1999.

SILVA, Marco Aurélio S. **Mapas auto-organizáveis na análise exploratória dedados Geoespaciais multivariados**. 2004. 117f. Dissertação (Mestrado em Computação Aplicada) – São José dos Campos: INPE (Instituto Nacional de Pesquisas Espaciais), São Paulo.

SIQUEIRA, Paulo Henrique. **Uma nova abordagem na resolução do problema do caixeiro viajante**. 2005. 102f. Tese (Doutorado em Ciências). Programa de Pós-graduação em Métodos Numéricos em Engenharia, Universidade Federal do Paraná, Curitiba, 2005.

SMITH, A. R., **The Viewing Transformation.** San Rafael, 1983. Technical Memo. nº 84.Disponível em: <http://alvyray.com/Memos/MemosCG.htm>. Acesso em: 01 Jul. 2013.

TAN, P. N.; STEINBACH, M.; KUMAR, V. **Introduction to Data Mining**. Inc. Boston, MA, USA: Addison-Wesley Longman Publishing Co. 2005.

ULTSCH, A. **Self-organizing neural networks for visualization and classification**. *Information and Classification*, Springer-Verlag, Dortmund, Alemanha, 1992.

ULTSCH, A.; VETTER, C. **Self-Organizing-Feature-Maps versus Statistical Clustering Methods: A Benchmark**. Research Report No. 9; Dep. Of Mathematics, University of Marburg, 1994.

VESANTO, J.; ALHONIEMI, E. **Clustering of the Self-Organizing Map**. IEEE Transactions on Neural Networks, v. 11, n. 3, p. 586–600, May 2000.

VESANTO, J.; HIMBERG, J.; ALHONIEMI, E.; PARHANKANGAS, J.***SOM Toolbox for Matlab 5***: report A57, April 2000. Libella Oy: Finland: *SOM* Toolbox Team, Helsinski University of Technology, 2000.

XU, B.; LI, S., **Automatic Color Identification in Printed Fabric Images by a Fuzzy Neural Network**. AATCC Review, v. 2, n. 9, p. 42-45, 2002.

ZHANG, X.; LI, Y., **Self-Organizing Map as a new method for clustering and data analysis**. In: International Joint Conference on Neural Networks, 1993, Nagoya: Proceedings of International Joint Conference on Neural Networks – IJCNN'93, p. 2448 -2451.

ZUCHINI, M.H. **Aplicações de mapas auto-organizáveis em mineração de dados e recuperação de informação**. 2003. Dissertação (Mestrado em Engenharia Elétrica). Faculdade de Engenharia Elétrica e Computação,Universidade Estadual de Campinas, São Paulo.