

Universidade Federal do Rio Grande do Sul  
Instituto de Informática  
Programa de Pós-Graduação em Computação

**Algoritmos e Ferramentas de  
Descoberta de Conhecimento em  
Bancos de Dados Geográficos**

por

VANIA BOGORNY

TI - III 1120 – PPGC - UFRGS

Prof. Dr. Luís Otávio Campos Álvares  
Orientador

Porto Alegre, março de 2003

CIP – Catalogação na Publicação

**Bogorny, Vania**

Algoritmos e Ferramentas de Descoberta de Conhecimento em Bancos de Dados Geográficos Vania Bogorny. – Porto Alegre: PPGC da UFRGS, 2003.

43.: il. – ( TI III - 1120).

Trabalho Orientado pelo Prof. Dr. Luís Otávio Campos Álvares

1. Descoberta de Conhecimento. 2. Sistemas de Informação Geográfica. 3. Algoritmos de DCBDG. 4. Ferramentas de DCBDG

Pe. II. Título. III. Série.

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL

Reitora: Profa. Wrana Panizzi

Pró-Reitor de Pós-Graduação: Prof. Franz Rainer Semmelmann

Diretor do Instituto de Informática: Prof. Philippe Oliver Alexander Navaux

Coordenadora do PPGC: Profa.

Bibliotecária-Chefe do Instituto de Informática: Beatriz Regina Bastos Hara

## Sumário

Sumário .....	3
Lista de Figuras .....	5
Resumo .....	6
Abstract .....	7
1 Introdução.....	8
2 Descoberta de Conhecimento em Bases de Dados .....	12
2.1 Etapas do Processo de Descoberta de Conhecimento.....	12
2.2 Tipos de Conhecimento.....	13
2.3 Formas de Aquisição de conhecimento .....	13
2.4 Dificuldades e Desafios da DCBD.....	14
2.5 Mineração de Dados.....	15
<b>2.5.1 Tipos de Aprendizado em Bases de Dados .....</b>	<b>15</b>
<b>2.5.2 Métodos de Mineração de Dados .....</b>	<b>16</b>
2.5.2.1 Árvores de Decisão.....	16
2.5.2.2 Redes Neurais .....	17
2.5.2.3 Algoritmos Genéticos .....	17
2.5.3.4 Regras de Decisão .....	18
2.6 Descoberta de Conhecimento em Bancos de Dados Geográficos .....	18
<b>2.6.1 Principais Tarefas e Abordagens .....</b>	<b>19</b>
3 Técnicas e Algoritmos de Mineração de Dados Geográficos.....	20
3.1 Generalização .....	20
3.2 Classificação.....	21
3.3 Regras de Associação.....	21
3.4 Agrupamento ou Clusterização .....	23
<b>3.5.1 Método Hierárquico .....</b>	<b>24</b>
<b>3.5.2 Método de Particionamento.....</b>	<b>26</b>
3.5.2.1 O algoritmo <i>K-Means</i> .....	26
3.5.2.2 O algoritmo <i>K-Medoids</i> .....	26
<b>3.5.2.3 Método Baseado em Densidade.....</b>	<b>27</b>
<b>3.5.2.4 Método baseado em Restrições de Contiguidade.....</b>	<b>28</b>

4 Ferramentas para Mineração de Dados Geográficos .....	30
4.1 GeoMiner .....	30
4.2 Padrão .....	33
4.3 INGENS .....	34
<b>4.3.1 O Modelo de Objetos do <i>INGENS</i> .....</b>	<b>36</b>
<b>4.3.2 Minerando Dados no INGENS.....</b>	<b>37</b>
5 Conclusão .....	38
Bibliografia.....	40

## Lista de Figuras

FIGURA 1 – Relacionamento espacial do tipo Rio <i>cruza</i> Floresta.....	10
FIGURA 2 – Relacionamentos Topológicos entre dois objetos do tipo área num espaço bi-dimensional .....	10
FIGURA 3 - Processo de Descoberta de Conhecimento [Han 01a].....	12
FIGURA 4 – Fundindo dois <i>clusters</i> no algoritmo <i>CURE</i> .....	25
FIGURA 5 – Dois clusters encontrados pelo <i>DBSCAN</i> .....	28
FIGURA 6 – Representação da relação espacial de vizinhança por matriz e grafo .....	29
FIGURA 7 – Arquitetura do GeoMiner .....	31
FIGURA 8 – Exemplo de extração de conhecimento usando <i>generalização</i> .....	31
FIGURA 9 – Exemplo de extração de conhecimento usando <i>classificação</i> .....	32
FIGURA 10 – Exemplo de extração de conhecimento usando <i>regras de associação</i> ...	32
FIGURA 11 - Arquitetura do Sistema Padrão.....	34
FIGURA 12 – Arquitetura do INGENS em três camadas.....	35

## Resumo

A Descoberta de Conhecimento em Bases de Dados (DCBD) é uma área de pesquisa voltada para a extração de conhecimento importante, porém oculto em bases de dados, onde a dimensionalidade, a complexidade ou a quantidade de dados armazenados inviabiliza a análise realizada pelos métodos tradicionais empregados nos sistemas gerenciadores de banco de dados. Esses métodos retornam apenas a informação explicitamente armazenada, não sendo capazes de deduzir ou induzir conhecimento. Em bases de dados geográficos (BDG) este problema é ainda mais complexo, devido ao volume de informações que é ainda maior que em bases de dados convencionais e pela característica espacial dos dados que precisa ser considerada, já que pode ser de grande importância na descoberta do conhecimento. A DCBD é um processo interativo, que alia a intuição e o conhecimento dos especialistas da aplicação à eficiência de técnicas inteligentes de extração de informação. A mineração de dados é a fase mais automatizada deste processo. Entretanto, a realização totalmente automatizada e de propósito geral da descoberta de conhecimento ainda está distante, pois existem vários problemas que ainda necessitam da orientação do usuário para guiar o processo. A Descoberta de Conhecimento em Bases de Dados Geográficos (DCBDG) é um tipo particular de descoberta, pois está ligada à extração de características e padrões espaciais interessantes, à identificação de relacionamentos entre dados espaciais e não-espaciais, restrições entre objetos geográficos e outras características não explicitamente armazenadas nestes bancos de dados. Para realizar o processo de DCBDG, novos algoritmos de Mineração de Dados foram criados e outros foram estendidos ou adaptados para suportar dados espaciais. Este trabalho de pesquisa busca descrever algumas das principais técnicas de mineração de dados e dentro destas técnicas, investigar alguns algoritmos que suportam dados espaciais. Busca também, fazer um levantamento das ferramentas que implementam essas técnicas para dados espaciais.

### **Palavras-chave:**

Descoberta de Conhecimento, Sistemas de Informação Geográfica, Bases de dados Geográficos, Mineração de Dados.

## **Abstract**

The Knowledge Discovery in Databases (KDD) is a research area which extracts either important as well as implicit knowledge from databases whose dimension, complexity and amount of stored data makes the data analyses unfeasible by traditional methods used in conventional DBMS (Database Management Systems). These methods just recover explicit information stored in the databases and they are unable to deduce or induce knowledge by itself. In Geographic Databases this problem is still more complicated because the data amount is larger than conventional databases due to its spatial characteristics that have to be considered in the knowledge discovery process. The KDD is an interactive process which combines the intuition as well as the knowledge of specialists with the efficiency of intelligent techniques to extract information. The Data Mining (DM) is the most automated phase of this process. However, it is not yet capable to work alone, once many problems still need the user's orientation to guide the knowledge discovery process. The knowledge discovery in geographic databases is a particular kind of knowledge once it find out interesting characteristics as well as spatial patterns; relationships between spatial and non spatial data; geographic objects constraints and other features not explicitly stored in these databases. To discover knowledge in geographic databases, some new data mining algorithms were created and others were extended or become adopted to support spatial data. This research work will describe some of the most popular data mining techniques and for some of that inquire algorithms that endure spatial data. It also presents some software tools that implement these techniques to spatial data.

### **Key words**

Knowledge Discovery, Geographic Information Systems, Geographic Databases, Data Mining.

# 1 Introdução

Os Sistemas de Informação Geográfica (SIG) são sistemas especiais, capazes de armazenar, manipular e analisar dados geográficos, ou seja, objetos e fenômenos do mundo real em que a localização geográfica é uma característica inerente e indispensável para tratá-los [CAM 96].

Os SIG manipulam dados descritivos, também chamados convencionais, como, por exemplo, a população e a área de um município; e dados espaciais, oferecendo uma estrutura consistente para análises e consultas envolvendo esses dois tipos de dado [GUT 94]. Com relação aos dados espaciais, os SIG permitem o armazenamento de informações de localização geográfica relativas a projeções cartográficas e escalas<sup>1</sup> específicas, além de características estruturais, geométricas e topológicas de entidades pertencentes a um domínio de aplicação.

As entidades do mundo real, também conhecidas como objetos ou feições geográficas [BOG 01], estão fortemente relacionadas. Por exemplo, ferrovias podem cruzar rios, rios podem atravessar florestas, estados são formados por municípios, postos de combustível estão próximos a rodovias, e assim por diante. Tais associações, denominadas relacionamentos espaciais, são alguns dos aspectos que diferenciam os SIG dos sistemas de informação convencional. Alterações na geometria de qualquer objeto  $x$  podem afetar a geometria ou o comportamento de objetos vizinhos. Através dessas associações é possível avaliar a influência que um objeto exerce sobre outro com o qual está geograficamente relacionado.

O crescimento acelerado do volume de dados armazenados tornou o processo de aquisição de conhecimento útil em bases de dados digitais inviável pelo método tradicional [BAS 01]. Nesse método, os analistas familiarizavam-se com os dados e então desenvolviam uma interface para o usuário interagir com estes dados. Entretanto, esse processo tornou-se caro e demorado, além de envolver o caráter subjetivo do analista e não apresentar todo o conhecimento que o conjunto dos dados poderia oferecer. Isso levou ao desenvolvimento de teorias e ferramentas automatizadas para auxiliar na extração de conhecimento implícito, porém útil para o usuário.

A Descoberta de Conhecimento em Bases de Dados (DCBD ou *KDD-Knowledge Discovery in Databases*) é um processo complexo e que está concentrado em identificar relacionamentos e informações úteis, que estão implícitas no repositório de dados [SAN 01]. A área de estudo responsável por essa temática está relacionada ao desenvolvimento de métodos e técnicas para descobrir algum conhecimento significativo, porém implícito em bases de dados.

O processo global de descoberta de conhecimento, que é dividido em várias etapas, inclui a gestão de algoritmos de mineração de dados utilizados para extrair e interpretar padrões dos dados. As ferramentas de DCBD utilizam uma diversidade de algoritmos para identificar relacionamentos e padrões que estão implícitos nos dados. Estes representam conhecimento acerca da Base de Dados (BD) explorada e das entidades nela contidas. Decidir se os padrões encontrados refletem ou não conhecimento útil, é uma das etapas que necessitam da participação do usuário.

---

<sup>1</sup> relação entre a dimensão dos elementos representados em um mapa e sua grandeza correspondente medida sobre a superfície da Terra.

Existem várias técnicas que são empregadas em mineração de dados (etapa do processo de DCBD), dentre as quais podemos citar agrupamento, classificação e regras de associação. A identificação de agrupamentos tem motivado muitas pesquisas na área e grande parte dos algoritmos de mineração de dados é baseada nessa técnica.

Os grandes progressos conseguidos até ao momento na área da DCBD restringem-se quase que exclusivamente [KOP 97] à exploração de dados armazenados em bases de dados relacionais. Existe, contudo, em várias bases de dados organizacionais uma dimensão espacial cuja semântica não é interpretada pelos algoritmos tradicionais de mineração de dados.

Em bases de dados geográficos (BDG), a descoberta de conhecimento é um tipo particular de descoberta, pois está ligada à extração de características e padrões espaciais interessantes, como a identificação de relacionamentos entre dados espaciais e não-espaciais, as restrições entre objetos geográficos e outras características que não estão explicitamente armazenadas nesses bancos de dados.

As características espaciais dos dados que precisam ser consideradas envolvem operações mais complexas e demoradas, mas que podem ser de fundamental importância para o processo de descoberta do conhecimento.

A análise de dados espaciais com o objetivo de descoberta de conhecimento requer a utilização de técnicas específicas, que permitam a inclusão da semântica espacial. As técnicas existentes [KOP 96] baseiam-se em algoritmos de mineração de dados (etapa do processo de descoberta de conhecimento), um pouco diferentes dos tradicionais, capazes de incluir a semântica espacial no processo de descoberta de conhecimento, ou, na integração de SIG com ferramentas de descoberta de conhecimento, que permitam a manipulação dos dados espaciais e não-espaciais.

A principal diferença entre a descoberta de conhecimento em bases de dados convencionais e a descoberta de conhecimento em bases de dados geográficos está nos relacionamentos espaciais existentes entre as entidades do mundo real [NEV 01].

Os relacionamentos espaciais podem ser classificados de várias formas, segundo alguns autores. Borges [BOR 97] agrupa-os em topológicos, métricos, de ordem e *fuzzy*. Parent [PAR 98] categoriza-os em topológicos, orientados (ou de ordem), métricos e de agregação. Faria [FAR 98] classificá-os em métricos, topológicos, de orientação e de localização.

Os relacionamentos de orientação ou direcionais [FAR 98, BOR 97] verificam se existe alguma relação de orientação entre dois objetos geométricos. Eles descrevem a ordem como os objetos estão posicionados uns em relação aos outros [CAM 96], partindo, normalmente, de um marco de referência que determina a direção na qual o objeto está localizado. Exemplos desse tipo de relacionamento são: *acima, abaixo, ao norte, ao sul, a leste, a oeste, à esquerda e à direita*.

Métricos são os relacionamentos de distância que descrevem quão afastado um objeto está em relação ao outro ou a um marco de referência [CLE 94]. Eles dependem de definições métricas no sentido de parametrizar o *quanto é perto ou longe*, o que dependerá das circunstâncias e das entidades geográficas relacionadas [BOR 97]. Exemplos de relacionamentos métricos são: *distância e comprimento*.

Os relacionamentos *fuzzy* são aqueles que tratam de entidades geográficas que não possuem limites bem definidos. Um lago, por exemplo, pode ter sua área bastante

modificada entre uma estação de seca e uma estação de chuvas. Muitos riachos chegam a desaparecer durante períodos de seca [LIS 97].

Relacionamentos topológicos são aqueles que se referem à posição relativa dos objetos no espaço onde estão contidos. Eles descrevem se dois objetos interceptam-se ou não, e qual a forma de interseção existente [CLE 93], [EGE 93]. Esse tipo de relacionamento espacial é preservado sob transformações como rotação, escala e translação. Na literatura, podem ser encontrados vários métodos que buscam definir um conjunto significativo de relacionamentos topológicos, os quais resultam nos seguintes tipos de relação entre dois objetos espaciais [CLE 94], [EGE 94]: *disjoint*, *touches*, *overlaps*, *equal*, *inside*, *contains*, *covers* e *coveredBy*. Esses relacionamentos expressam ligações entre objetos espaciais do tipo ponto, linha e polígono. A figura 1 ilustra um exemplo de relacionamento topológico de cruzamento entre os objetos geográficos do tipo rio e floresta.



FIGURA 1 – Relacionamento espacial do tipo Rio *cruza* Floresta

A figura 2 ilustra, geograficamente, como os relacionamentos topológicos entre objetos geográficos do mundo real, representados espacialmente por polígonos, são traduzidos para um sistema computacional.

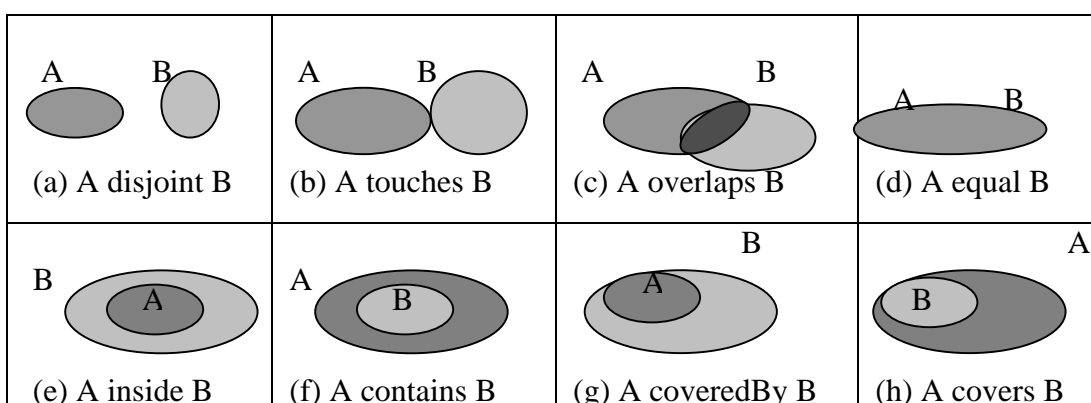


FIGURA 2 – Relacionamentos Topológicos entre dois objetos do tipo área num espaço bi-dimensional

Uma vez que, em bases de dados espaciais, um objeto pode afetar ou ser afetado pelo comportamento do objeto vizinho, as relações de vizinhança são de grande importância para o processo de descoberta de conhecimento. Esses relacionamentos,

normalmente, não estão armazenados na base de dados, pois são calculados por funções específicas, que manipulam dados espaciais. Por isso, os algoritmos de mineração de dados precisam ser capazes de avaliar a dependência espacial, utilizando a informação sobre relacionamentos de vizinhança.

Este trabalho está organizado da seguinte forma: o Capítulo 2 apresenta uma visão geral do processo de descoberta de conhecimento em bases de dados.

O Capítulo 3 aborda as diferentes técnicas utilizadas na mineração de dados espaciais e alguns algoritmos que implementam essas técnicas.

No Capítulo 4, são descritas algumas ferramentas que auxiliam no processo de descoberta de conhecimento em bases de dados geográficos e, o Capítulo 5, por sua vez, apresenta as conclusões.

## 2 Descoberta de Conhecimento em Bases de Dados

A necessidade de recolher e armazenar dados de diversos tipos, formatos e origem, superou a capacidade humana de analisar, sintetizar e extrair conhecimento a partir desses dados. Enquanto as BD fornecem as ferramentas necessárias ao armazenamento e a utilização de grandes quantidades de dados, a compreensão e a análise dos mesmos requer a utilização de ferramentas apropriadas, que automatizem o processo de análise dos dados e descoberta de conhecimento [FAY 96].

Os princípios associados a DCBD conjugam fundamentos provenientes de diversas áreas, tais como a inteligência artificial, a aprendizagem automática, o reconhecimento de padrões e a estatística. As aplicações de DCBD integram teorias, métodos e algoritmos provenientes destas diferentes áreas, tendo como objetivo a extração de conhecimento a partir de grandes BD.

Os algoritmos utilizados para procurar padrões nos dados são denominados algoritmos de mineração de dados ou *Data Mining* (DM). O processo global de DCBD, que se desenvolve em várias etapas, inclui a gestão dos algoritmos de DM e a interpretação dos padrões encontrados pelos mesmos. A interpretação dos padrões pode dar suporte à tomada de decisão.

### 2.1 Etapas do Processo de Descoberta de Conhecimento

O processo de DCBD [Han01], conforme já mencionado, tem como principal objetivo extrair regras e informações implícitas de grandes banco de dados. Para isso, quatro etapas sucessivas foram definidas (conforme figura 3): limpeza e integração dos dados; seleção e transformação dos dados; mineração e avaliação; e apresentação do conhecimento. Essas etapas podem ser generalizadas em três grandes fases: pré-processamento, mineração e pós-processamento de dados.

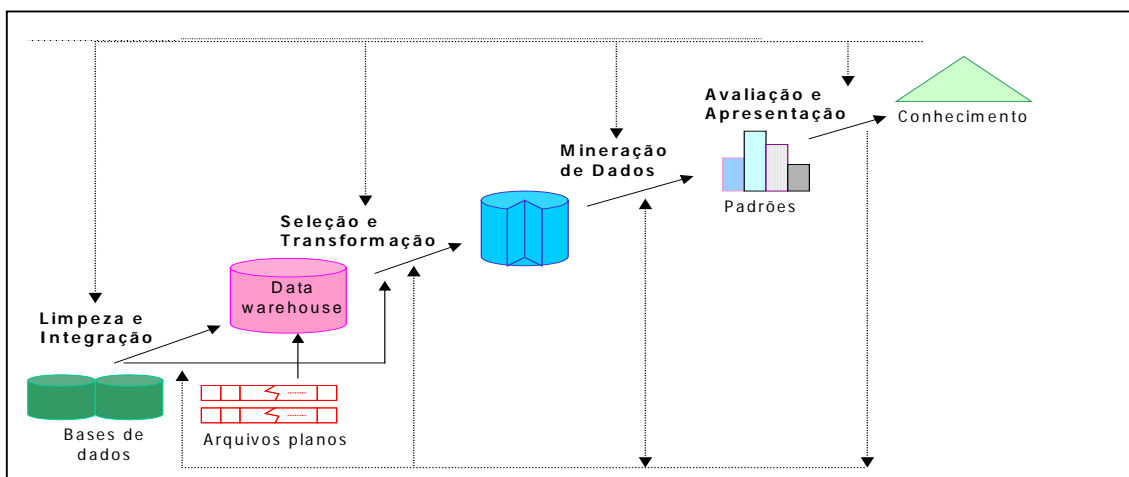


FIGURA 3 - Processo de Descoberta de Conhecimento [Han 01a]

O *pré-processamento* é uma das fases mais demoradas do processo de DCBD, e, segundo pesquisas, consome cerca de 80 % dos esforços necessários para concluir todo o processo. Nessa fase são realizadas as seguintes tarefas:

- determinação dos objetivos da descoberta: define-se claramente o problema;

- limpeza dos dados: eliminação de ruído e inconsistência dos dados;
- integração dos dados: podem ser combinados dados de múltiplas fontes;
- seleção de dados: os dados relevantes para a mineração de dados são identificados e agrupados, gerando uma amostra do banco de dados;
- transformação de dados: conversão dos dados para um formato interpretável pelas ferramentas de mineração de dados.

A *mineração de dados* é a etapa em que se definem os padrões a serem usados para a extração do conhecimento e selecionam-se os algoritmos de mineração, que serão aplicados sobre os dados selecionados. Essa etapa destaca-se de tal forma dentre as etapas do processo de KDD, que levou alguns autores a considerá-la como sinônimo de descoberta de conhecimento [BAS 01].

O *pós-processamento* é realizado através das etapas de avaliação e apresentação dos padrões, que são responsáveis pela identificação e análise dos padrões interessantes que representam conhecimento, bem como, a forma como o conhecimento extraído será apresentado ao usuário.

## 2.2 Tipos de Conhecimento

Após a execução de todas as etapas do processo de DCBD, diferentes tipos de conhecimento podem ser extraídos. Segundo ADDRIANS [ADR 97], o conhecimento descoberto pode ser classificado da seguinte forma:

- *conhecimento superficial*: consiste na informação que pode ser facilmente recuperada através de uma ferramenta de consulta;
- *conhecimento multi-dimensional*: é a informação que permite a análise dos dados através de ferramentas de processamento analítico. Esse tipo de conhecimento pode ser rapidamente extraído e, na maioria das vezes, é obtido por ferramentas de consulta como, por exemplo, SQL (*Structure Query Language*);
- *conhecimento oculto*: é a informação que pode ser encontrada através da aplicação de algoritmos de reconhecimento de padrões ou aprendizado de máquina. Linguagens de consulta como SQL também poderiam ser utilizadas para a extração desse tipo de conhecimento, porém consumiriam muito tempo de processamento;
- *conhecimento profundo*: é a informação que está armazenada na base de dados, mas só pode ser localizada se alguém informar onde ela está contida. A diferença entre conhecimento oculto e conhecimento profundo é que; o primeiro, pode ser encontrado por ferramentas de DCBD e; o segundo, somente com a indicação de pistas que indicam ou apontem ao conhecimento implícito.

## 2.3 Formas de Aquisição de conhecimento

O núcleo de um sistema de DCBD é formado pelos algoritmos de extração de padrões. Existem diferentes formas de extrair padrões da base de dados e, algumas delas estão relacionadas abaixo [FEL 97]:

- *correlação*: um elemento da base de dados exerce alguma influência sobre outro elemento. O valor dessa influência pode ser previamente especificado;
- *dependência*: há uma dependência entre dois elementos, sendo que a alteração das características de um elemento pode afetar o comportamento do outro. Por exemplo, um parto normal ou cesariana sempre é dependente do sexo da pessoa (feminino);
- *detecção de sequência*: existe uma dependência em relação ao tempo. Isso ocorre quando um procedimento precede o outro, ou quando ele somente pode ser repetido após um intervalo mínimo de tempo;
- *classificação*: o algoritmo detecta os padrões e descreve conceitos. Ao contrário da forma de descrição de conceitos, o algoritmo é quem identifica a classe a qual o elemento pertence;
- *regressão linear*: o algoritmo mapeia um item de dado para uma variável de precisão com valor real, permitindo que os dados sejam discriminados através da combinação de atributos de entrada;
- *detecção de desvio*: o algoritmo procura por elementos ou ocorrências que estão fora do conjunto de dependências, sequências ou descrições de conceitos, ou seja, os elementos que estão fora do Padrão. Ele detecta anomalias na base de dados, evidenciando problemas de qualidade e fraudes.

## 2.4 Dificuldades e Desafios da DCBD

A total automatização do processo de DCBD ainda está distante, pois inúmeros problemas ainda precisam da intervenção do usuário para ser solucionados [FEL 97]. Essa área tem evoluído através de sistemas projetados e implementados para fins específicos, podendo ser utilizados em várias bases de dados, mas onde o objetivo da descoberta seja semelhante. Por outro lado, os sistemas mais genéricos dependem da intervenção do usuário, exigindo deste um conhecimento prévio e significativo sobre os dados, para que algum conhecimento possa, de fato, ser encontrado.

Alguns problemas estão relacionados com as decisões que precisam ser tomadas durante o processo de descoberta de conhecimento como, por exemplo, a representação do conhecimento extraído, a definição das prioridades do conhecimento descoberto e a seleção do método de mineração de dados mais adequado.

Segundo FAYAD [FAY 96], além dos problemas já mencionados, existem outros fatores que podem dificultar o êxito do processo de DCBD. Alguns destes fatores estão relacionados abaixo:

- *volume da base de dados*: as bases de dados estão cada vez maiores, acarretando inúmeras combinações entre os dados, podendo resultar numa grande variedade de padrões, combinações e hipóteses;
- *complexidade e dimensionalidade*: quanto maior a base de dados, maior é o número de atributos e relacionamentos entre eles. Isso aumenta a possibilidade do algoritmo encontrar padrões falsos. Quanto mais complexa a base de dados, mais eficiente precisa ser o algoritmo para extrair conhecimento;

- *dados inconsistentes, inválidos ou com ruído*: como as bases de dados não foram projetadas para extrair conhecimento através de técnicas de aprendizado de máquina, muitos atributos importantes podem não estar presentes na base de dados ou ter valores nulos. Além disso, os atributos existentes podem estar com valores errados ou serem redundantes;
- *representação do conhecimento*: o conhecimento descoberto precisa ser compreendido pelo usuário para evitar que este interprete mal o conhecimento extraído;
- *dados constantemente alterados*: a natureza dinâmica dos dados faz com que eles sejam constantemente alterados, podendo levar a conclusões prévias e errôneas, pois as variáveis medidas podem ter sido removidas ou modificadas.

## 2.5 Mineração de Dados

A mineração de dados é uma etapa do processo de DCBD, que consiste na aplicação de algoritmos (veja Capítulo 3) específicos, que tenham uma limitação aceitável de eficiência computacional e que sejam capazes de produzir uma enumeração particular de padrões [FAY 96]. Esses algoritmos utilizam técnicas de aprendizado indutivo sobre bases de dados e são capazes de extrair conhecimento através de exemplos, aplicando métodos iterativos por repetidas vezes.

### 2.5.1 Tipos de Aprendizado em Bases de Dados

A informação contida nas bases de dados deve ser recuperada de forma rápida e correta. Essa informação pode não estar contida explicitamente na base de dados, podendo ser *inferida*. Há duas técnicas principais de inferência de dados [HAL 2000], [DOM 01]: dedução e indução. A dedução é uma consequência lógica das informações contidas na base de dados, isto é, a informação é extraída da base de dados a partir da utilização de operadores dos próprios SGBD. Ela resulta em descrições corretas em relação ao mundo descrito na base de dados.

A indução é uma técnica de inferência que generaliza as informações contidas na base de dados. Na aplicação da indução, a base de dados é percorrida em busca de padrões ou regularidades, que são combinações de valores para certos atributos que compartilham características comuns. Cada regularidade forma uma regra, prevendo o valor de um atributo com base em outros atributos.

O aprendizado indutivo consiste na criação de um modelo, onde os objetos e eventos similares são agrupados em classes. Para cada classe é criado um conjunto de regras que caracterizam o comportamento dos elementos de cada uma. Segundo [AVI 98], existem duas técnicas de aprendizado indutivo:

- *aprendizado supervisionado ou aprendizado de exemplos*: nesta técnica são fornecidas classes e exemplos de cada classe ao sistema, o qual precisa encontrar a descrição (propriedades comuns nos exemplos) de cada classe. Neste tipo de aprendizado existe um “professor” que guia o processo de aprendizado. O professor, na realidade, é o conhecimento prévio dos conceitos ou classes que estão sendo descritas pelo conjunto de exemplos de treinamento;

- *aprendizado não-supervisionado ou aprendizado por observação*: o sistema precisa descobrir a classe dos objetos através das propriedades que os mesmos têm em comum. Neste tipo de aprendizado, o indutor analisa os exemplos fornecidos e tenta determinar se alguns deles podem ser agrupados, formando *clusters*. Após a criação dos agrupamentos, normalmente, é necessária uma análise para determinar o que cada agrupamento significa no contexto do problema que está sendo analisado.

Ainda com relação aos tipos de aprendizado, existe o aprendizado de máquina e o aprendizado em mineração de dados. Um sistema de aprendizado de máquina utiliza informações de um conjunto de treinamento, ou seja, uma amostra de dados cuidadosamente selecionados [AVI 98]. A amostra é gerada de acordo com a técnica de aprendizado utilizada. Se esta técnica for de aprendizado supervisionado, o sistema busca as descrições das classes definidas pelo usuário e, se for não-supervisionada, o sistema gera um conjunto de novas classes juntamente com suas descrições.

Um sistema de aprendizado em mineração de dados busca descrições de dados em uma base inteira e não em uma amostra [HAL 2000], resultando em um processo mais complexo e demorado.

Apesar das estruturas de aprendizado de máquina e aprendizado em mineração de dados serem muito semelhantes, existem algumas diferenças importantes. Uma delas é que uma base de dados é construída de acordo com a necessidade da aplicação e não com as necessidades da mineração de dados. Por isso, as propriedades ou os atributos que simplificariam a tarefa de aprendizado em mineração de dados podem estar ausentes, o que não acontece no aprendizado de máquina, onde todas as informações ou atributos necessários são fornecidos ao conjunto de treinamento.

## **2.5.2 Métodos de Mineração de Dados**

Na literatura podem ser encontrados vários métodos para realizar a mineração de dados, os quais atendem a diferentes propósitos, cada qual com suas vantagens e desvantagens. A escolha de um desses métodos depende do contexto e do domínio da aplicação, bem como, do tipo de conhecimento que se deseja encontrar. Alguns dos principais métodos utilizados na etapa de mineração de dados estão descritos nas próximas seções.

### **2.5.2.1 Árvores de Decisão**

Segundo SOUZA [SOU 98], árvore de decisão é uma forma simples de classificar exemplos em um número finito de classes. Ela consiste de nodos que representam os nomes dos atributos, ligações/arcos que representam os valores dos atributos e as folhas, que correspondem às diferentes classes a que pertencem as entidades.

Uma árvore de decisão tem a função de particionar recursivamente um conjunto de treinamento até que cada subconjunto obtido deste particionamento contenha casos de uma única classe. Um objeto é classificado seguindo o caminho da raiz da árvore até a folha, enquanto as suas características satisfazem as ligações.

Depois de construída a árvore de decisão, é possível gerar árvores mais otimizadas e específicas, através da *poda*, que reduz o número de nodos internos e a complexidade da árvore.

Uma grande vantagem das árvores de decisão é que elas podem ser aplicadas a grandes conjuntos de dados e possibilitam uma visão real da natureza do processo de decisão, de forma que, o resultado do algoritmo possa ser facilmente interpretado pelo usuário.

As árvores de decisão também permitem derivar regras (chamadas regras de produção ou decisão), as quais são geradas, percorrendo o trajeto do nodo raiz até uma folha da árvore. Como as regras são equivalentes às árvores de decisão, a derivação de regras é interessante quando as árvores crescem muito.

Como exemplos de algoritmos baseados em árvores de decisão podemos citar o ID3 e o C4.5.

### 2.5.2.2 Redes Neurais

As redes neurais artificiais foram desenvolvidas a partir de uma tentativa de criar em computador, utilizando estudos do sistema nervoso biológico, um modelo computacional que simule a estrutura e o funcionamento do cérebro humano, buscando a chamada Inteligência Natural [AVI 98].

Cada neurônio artificial possui um ou mais sinais (valores) de entrada e um sinal de saída, sendo que a saída de um neurônio pode servir de entrada a diversos outros neurônios. Um neurônio, basicamente, tem a função de avaliar os valores de entrada; calcular o total de valores de entrada combinados; comparar o total com um valor limiar e determinar o que será a saída. Enquanto a operação de cada neurônio é razoavelmente simples, a conexão de um conjunto de neurônios pode gerar procedimentos complexos.

Existem vários tipos e formas de redes neurais que são aplicáveis a descoberta de conhecimento. Essas redes são diferenciadas pelo tipo de conexão entre os neurônios, o número de camadas de neurônios e o tipo de treinamento utilizado.

O resultado encontrado pelas redes neurais em relação às árvores de decisão e outros métodos é praticamente o mesmo. Entretanto, elas trabalham melhor com dados que contêm ruído.

As redes neurais têm algumas desvantagens quando aplicadas na etapa de mineração de dados. Além do processo de aprendizado ser muito lento, quando comparado com outros sistemas de aprendizado, o resultado gerado torna o conhecimento praticamente incompreensível ao usuário.

### 2.5.2.3 Algoritmos Genéticos

Um algoritmo genético é um procedimento de busca, baseado em mecanismos de seleção natural. Ele utiliza um conjunto de descrições candidatas, também chamadas de população ou organismos, e aumenta gradualmente a qualidade dessa população através da construção de novas descrições, criadas a partir das melhores descrições da população corrente [BAR 96]. As novas descrições geradas formam a segunda geração e, novamente, as melhores descrições são recombinadas para formar a próxima geração e assim sucessivamente.

Novos indivíduos são criados usando-se dois operadores principais de combinação genética: *crossover* e *mutação*. O *crossover* se dá pela aproximação dos cromossomos dos dois indivíduos pais, que trocam partes de seus cromossomos, resultando em dois cromossomos diferentes, mas que ainda guardam influências dos pais [SIL 97].

#### 2.5.3.4 Regras de Decisão

As regras são condições do tipo *if-then* que são sucessivamente generalizadas de forma que resumem o conteúdo da base de dados.

Dentre as vantagens do uso de regras está a facilidade na interpretação dos resultados; facilidade de incorporação de conhecimento que fica explícito nas regras e; facilidade de armazenamento das regras numa base de conhecimento.

## 2.6 Descoberta de Conhecimento em Bancos de Dados Geográficos

Um caso particular da DCBD diz respeito à exploração de dados georreferenciados, isto é, dados que incluem referências a objetos geográficos, localizações ou partes de uma divisão territorial. Ela refere-se à extração de padrões espaciais e características interessantes como relacionamentos espaciais e relacionamentos existentes entre dados espaciais e dados descritivos, a construção de uma base de conhecimento espacial e a descoberta de conhecimento não explicitamente armazenado na base de dados [KOP 97].

A principal diferença entre a análise dos dados espaciais e dados não-espaciais está associada ao fato das entidades geográficas endereçadas poderem ser afetadas por características de entidades vizinhas. A influência mútua que duas entidades exercem entre si depende de fatores como a topologia, a distância e a direção existente entre elas.

Em bases de dados tradicionais, os dados estão relacionados, mas são independentes. Em BDG, os dados são interdependentes, pois estão geograficamente relacionados uns aos outros como, por exemplo, um estado (Rio Grande do Sul) está dentro de um país (Brasil).

Segundo Koperski [KOP 97], a interdependência dos dados causa problemas na descoberta de conhecimento porque os algoritmos de mineração consideram os dados como independentes. Esse problema levou pesquisadores da área de SIG e KDD a estender as técnicas tradicionais de mineração de dados para suportar dados espaciais.

A semântica associada à localização dos objetos do mundo real e a análise dessas localizações fazem com que a utilidade do conhecimento obtido no processo de descoberta de conhecimento seja largamente melhorada através da integração de dados espaciais e dados não-espaciais [SAN 01].

O processo de DCBDG não se aplica somente aos SIG, mas nas áreas de sensoriamento remoto, em bases de dados de imagens e de muitas outras áreas onde são utilizados dados espaciais.

A Descoberta de Conhecimento em Bases de Dados Geográficas (DCBDG) refere-se ao processo de extração de padrões ou regularidades espaciais nos dados, relacionamentos existentes entre dados espaciais e dados não-espaciais, ou outras características implícitas em BDG. Este processo também desempenha um papel fundamental na percepção das características não-espaciais associadas aos dados espaciais.

### 2.6.1 Principais Tarefas e Abordagens

As tarefas tradicionalmente associadas ao processo de DCBDG incluem [KOP 95] [EST 98] [HAN 01]:

- *descrição da distribuição espacial dos dados e sua caracterização espacial*: a caracterização espacial de um conjunto de objetos consiste na descrição das propriedades espaciais e não-espaciais comuns aos objetos analisados. Nesta caracterização, não são consideradas apenas as propriedades dos objetos alvo do estudo, mas também as propriedades dos objetos vizinhos. Esta tarefa permite determinar o conjunto de registos (atributo, valor) e o conjunto de objetos para os quais a frequência relativa de incidência nesse conjunto, e nos seus vizinhos, é diferente da frequência relativa verificada nos demais registos da BD;
- *verificação das características não-espaciais em regiões geográficas através da análise espacial discriminante*: a análise espacial discriminante permite contrastar padrões espaciais de dados não-espaciais, comparando a variação dos atributos não-espaciais em diversas regiões geográficas (uma regra discriminante compara, por exemplo, o número de mortes causadas por neoplasias em diversas regiões geográficas);
- *estabelecimento de relações entre dados espaciais e entre dados espaciais e não-espaciais usando associação espacial*: a associação espacial permite identificar a relação existente entre um conjunto de objetos espaciais e um conjunto de dados não-espaciais, ou entre dois conjuntos de dados espaciais, definindo a associação (implicação) que existe entre os mesmos. Uma regra de associação espacial deve integrar pelo menos um predicado espacial, que pode estar associado a relações do tipo direção, distância ou topologia;

## 3 Técnicas/Tarefas e Algoritmos de Mineração de Dados Geográficos

As principais técnicas de mineração de dados podem ser classificadas em: generalização, associação espacial, aproximação e agregação, classificação e *clusterização*.

Na literatura podem ser encontrados vários algoritmos que implementam as diferentes técnicas de mineração de dados. A maior parte dos algoritmos de mineração de dados implementam a técnica de *agrupamento*, razão pela qual essa técnica é descrita, neste trabalho, em maior nível de detalhamento.

Alguns desses algoritmos foram estendidos para suportar dados espaciais. As próximas seções descrevem, em diferentes níveis de detalhe, as técnicas de mineração de dados espaciais e alguns algoritmos que implementam essas técnicas.

### 3.1 Generalização

Os algoritmos para descoberta de conhecimento baseados em *generalização* abstraem um grande conjunto de dados relevantes, partindo de um nível conceitual baixo para um nível mais elevado, extraindo conhecimento sobre os dados generalizados.

Na área de agricultura, por exemplo, *agrocere*s e *brascalb* podem ser generalizadas para o conceito de milho, que por sua vez pode ser generalizado para o conceito de grãos. Os dados espaciais podem ser generalizados da mesma forma. Por exemplo: os municípios podem ser generalizados em estados, os estados podem ser generalizados em países e estes em continentes.

Em [LU 93] foi investigado o processo de DCBDG baseado em generalizações. A estratégia implementada integra a indução orientada a atributos não-espaciais e a sua posterior caracterização espacial, generalizando os respectivos dados espaciais.

A generalização parte do conceito de hierarquia, que pode ser definida pelo usuário ou gerada automaticamente pelo sistema a partir da análise dos dados. Em bases de dados geográficos, podem ser definidas duas hierarquias; uma, para dados espaciais e outra, para dados convencionais [EST 01].

Os dados espaciais são generalizados, inicialmente, de acordo com o seu tipo (exemplos: rio, rodovia, estado, etc), através da seleção de um conjunto de dados relevantes, usando uma linguagem de consulta. Um outro tipo de generalização é reduzir, por exemplo, um objeto representado espacialmente por uma superfície para um único ponto. Da mesma forma, os relacionamentos entre os objetos espaciais e não-espaciais são generalizados.

Quando a generalização ocorre sobre os dados não-espaciais, os próximos passos são executados:

- os valores dos atributos nas tabelas são substituídos por valores generalizados. Por exemplo: o valor 9 graus de temperatura em uma tupla pode ser enquadrado

em um intervalo de 0 a 10 graus. Cenouras e batatas podem ser generalizadas em um nível de abstração mais alto como, por exemplo, vegetais;

- os valores que não foram generalizados são removidos da tabela;
- tuplas idênticas são fundidas em uma única tupla e um contador armazena em cada nova tupla, a quantidade de tuplas agrupadas.

Esses passos são executados, até que todos os atributos estejam no nível de generalização desejado.

## 3.2 Classificação

A técnica de *classificação* consiste no enquadramento dos dados, armazenados na base de dados explorada, dentro de classes pré-definidas [SAN 01]. O modelo de classificação construído pelo algoritmo de mineração permite determinar em qual classe cada elemento da base de dados se enquadra.

A classificação permite encontrar regras que dividem o conjunto de objetos em grupos, de forma que um determinado objeto seja caracterizado pelo comportamento do grupo onde está inserido. Os objetos podem ser classificados não somente pelas características espaciais mas também pelos atributos descritivos ou funções espaciais.

A técnica de classificação mais utilizada está associada à construção de árvores de decisão. Como os dados geográficos integram objetos espaciais e descrições não-espaciais dos mesmos, a classificação espacial permite encontrar regras para as classes com base nas propriedades não-espaciais e nas relações espaciais existentes entre os elementos classificados.

O processamento das relações espaciais existentes entre os objetos pode ser extremamente dispendioso, quando analisado numa perspectiva computacional [KOP 95]. Para acelerar este processo, as regras encontradas são generalizadas e avaliadas, buscando selecionar as mais interessantes.

Com relação aos algoritmos que implementam essa técnica, Ester [EST 97] estendeu o algoritmo *ID3* [QUI 86] para a classificação de objetos espaciais armazenados em BDG. Nesta abordagem, a detecção das relações espaciais entre objetos utiliza o conceito de vizinhança na forma de predicados do tipo *perto\_de* ( $a,b$ ), *dentro\_de* ( $a,b$ ), de forma a suportar os três principais tipos de relacionamentos espaciais (topológicos, direcionais e de ordem).

## 3.3 Regras de Associação

Uma regra de associação é a descrição geral de um conjunto de dados relacionados. Em bases de dados geográficas, essas regras referem-se aos objetos relacionados geograficamente.

O objetivo das regras de associação ou regras associativas é descobrir itens de uma transação que implicam a presença de outros itens na mesma transação. O algoritmo mais conhecido que implementa regras de associação é o APRIORI. Ele foi estendido para suportar dados espaciais.

As regras associativas são formadas por predicados; um antecedente e um conseqüente. O formato dessas regras pode ser expresso na forma  $X \rightarrow Y$ ; onde  $X$ , representa o antecedente e  $Y$ , o conseqüente. Dentro dessas regras existem duas probabilidades: o *grau de confiança* e o *grau de suporte* ou suporte mínimo. O *grau de confiança* indica a probabilidade de ocorrência do conseqüente em relação ao antecedente, ou seja, indicando a força ou a confiabilidade da regra. Ele é determinado pela divisão do número de transações (registros) da base de dados que admitem a regra de associação pelo número de transações que suportam somente o corpo (conseqüente) da regra [HAN 95].

O *grau de suporte* indica a probabilidade da regra ocorrer em relação à base de dados inteira. Ele é definido pela divisão do número de transações que suportam a regra pelo número total de transações.

Um exemplo de regra de associação é aquele em que 90 % dos clientes que compram os produtos pão e manteiga, na mesma transação, também compram leite.

Uma regra de associação espacial descreve a implicação de uma ou um conjunto de feições sobre um outro conjunto de feições em uma base de dados espaciais. “Os grandes rios estão cercados por florestas” é uma regra de associação espacial.

As regras de associação espacial podem ser descritas através de conjuntos de predicados, mas pelo menos um, seja o antecedente ou o conseqüente, precisa envolver objetos espaciais [KOP 95]. Na regra  $is\_a(x, house) \wedge close\_to(x, beach) \rightarrow is\_expensive(x)$  (90%), o antecedente  $is\_a$  e  $close\_to$  são espaciais. Esse exemplo expressa que 90% das casas localizadas próximas a praias são caras. Já no exemplo  $is\_a(x, gas\_station) \rightarrow close\_to(x, highway)$  (75%), os dois predicados são espaciais. Essa regra expressa que 75% dos postos de combustível estão localizados próximos a rodovias.

As técnicas de classificação e agrupamento são capazes de identificar relacionamentos espaciais e não-espaciais, quando os objetos estão espacialmente localizados na mesma região [KOP 95]. Entretanto, elas não são capazes de descobrir regras ou padrões de objetos espaciais localizados em diferentes regiões geográficas e não conseguem extrair conhecimento de dados espaciais e não-espaciais conjuntamente, já que o agrupamento é realizado separadamente para dados espaciais e não-espaciais. As regras de associação são uma técnica que pode complementar as demais suprindo esta falta, já que os predicados da regra podem ser formados por dados espaciais e não-espaciais.

Os predicados espaciais podem representar os relacionamentos topológicos *disjoint*, *intersects*, *inside*, *contais*, *touches*, *covers*, *covered-by*, *equal* e *crosses*; os relacionamentos de orientação como *left*, *right*, *nort*, *east*, etc ou conter uma informação de distância como *close\_to* e *far\_away*.

Considerando que cada objeto geográfico tem um atributo *tipo* que o caracteriza e um atributo *geometria* que corresponde à parte espacial do objeto geográfico, os algoritmos que implementam essa técnica realizam as seguintes tarefas para extrair regras de associação espacial:

- a) seleciona os dados e os atributos relevantes, através de uma linguagem de consulta e armazena o resultado em uma tabela no banco de dados. Essa etapa pode ser realizada com uma linguagem de consulta espacial como, por exemplo, a GMQL implementada pela ferramenta *GeoMiner* (veja Capítulo 4);

- b) cria uma tabela *T1*, generalizando todos os objetos geográficos de acordo com o seu tipo. Por exemplo: os objetos Jacuí, Atlântico e Patos podem ser generalizados em *rios*, *oceanos* e *lagoas*, respectivamente;
- c) gera uma tabela *T2* com todos os objetos geográficos que estão em *T1* e que estão próximos ou relacionados, ou seja, agrupa os objetos e os predicados que descrevem os relacionamentos desses objetos;
- d) gera uma tabela *T3*, agrupando as tuplas que são iguais em *T2* e cria um atributo para indicar o número de vezes que tuplas iguais se repetem. Após, elimina dessa tabela todas as linhas onde o número de ocorrências de cada relacionamento for inferior ao percentual mínimo definido, ou seja, elimina as tuplas com o número de ocorrência menor que o suporte mínimo estipulado;
- e) Com base na tabela *T3* são geradas as regras de associação espacial. O atributo número de ocorrência de cada linha idêntica na tabela *T3* representa o grau de confiança.

Detalhes sobre esse tipo de técnica estão em [KOP 95].

### 3.4 Agrupamento ou Clusterização

O *agrupamento* ou *clusterização* consiste em identificar coleções de objetos semelhantes. O agrupamento dos dados pode ser baseado em funções de distância, que podem ser especificadas para diferentes contextos da aplicação. Pode-se, por exemplo, agrupar casas de uma área, de acordo com sua categoria, área construída e localização geográfica.

A tarefa básica da *clusterização* é agrupar um conjunto de objetos em subconjuntos, de acordo com os critérios apropriados [NEV 01]. Esses subconjuntos agrupam elementos que têm um alto grau de semelhança ou similaridade, enquanto que, quaisquer elementos pertencentes a grupos distintos tenham pouca semelhança entre si.

Uma característica que torna o *clustering* uma das técnicas mais utilizadas é a sua habilidade de identificar estruturas diretamente dos dados, sem que haja um conhecimento prévio dos mesmos. Essa técnica tem sido bastante utilizada em análise exploratória de dados espaciais e em procedimentos de regionalização. Os algoritmos de *clustering* com restrição de contiguidade espacial, por exemplo, são utilizados com o objetivo de agrupar áreas homogêneas em regiões contíguas. Eles podem ser aplicados para agrupar unidades espaciais como setores censitários em regiões maiores, reduzir erros associados ao posicionamento geográfico e diminuir o custo de análise dos dados.

Segundo Neves [NEV 01], os critérios mais comuns adotados na técnica de agrupamento são a homogeneidade e a separação. A primeira, refere-se a objetos pertencentes a um mesmo *cluster*, onde os objetos são o mais similares possível. Na segunda, os objetos de diferentes *clusters* devem ser o mais distintos possível.

A qualidade dos *clusters* gerados depende de uma série de definições estabelecidas pelo usuário como, por exemplo, escolha dos atributos, medidas de dissimilaridade, critérios de agrupamento, escolha do algoritmo e definição do número de *clusters*. A dissimilaridade normalmente é utilizada por essa técnica para avaliar o grau de semelhança entre dois objetos durante o processo de agrupamento. Muitas vezes, essa medida é apresentada como sendo a distância entre dois objetos.

Existem vários algoritmos que implementam essa técnica como: *k-means*, *Agnes*, *Clarans*, *Cure*, etc. Para escolher o algoritmo adequado, alguns fatores devem ser considerados:

- *objetivo da aplicação*: se a aplicação envolve um supermercado que deseja escolher o melhor ponto para construir uma filial, o algoritmo de *clusterização* escolhido deve ser um que permita encontrar a menor distância entre o ponto sugerido para a construção do supermercado e os clientes. Sempre que a aplicação envolver a menor distância entre o centro do *cluster* em relação aos demais objetos, HAN [HAN 94] sugere que sejam usados os algoritmos *k-means* e *k-medoids*. Já em aplicações onde os dados espaciais são imagens de satélite ou mapas de vegetação, uso do solo, etc, é aconselhável o uso de algoritmos baseados em densidade, os quais criam *clusters* mais uniformes em termos de densidade, cor, pixels, e assim por diante.
- *velocidade X qualidade*: o algoritmo de *clusterização* ideal, para qualquer aplicação, deveria atender a dois critérios: velocidade de execução e qualidade dos *clusters* gerados [HAN 94]. O problema é que os algoritmos que geram *clusters* de boa qualidade são incapazes de ler grandes bases de dados. Para solucionar esse problema, HAN sugere que se comprima ou diminua o tamanho da base de dados e então se execute o algoritmo sobre a base condensada. O desafio, nesse caso, é reduzir a base sem que haja perda de informação importante a ponto de não prejudicar a geração dos *clusters*;
- *características dos dados*: as características dos dados a serem agrupados devem ser consideradas durante a escolha do algoritmo.
- *domínio dos atributos*: se os atributos dos objetos são numéricos, a clusterização é facilitada. Se forem binários, categóricos ou ordinais, a clusterização se complica, já que a maioria dos algoritmos que implementam essa técnica foi desenvolvida para o uso de atributos numéricos;
- *ruído*: na escolha do algoritmo, deve ser observado o percentual de dados com ruído, pois alguns algoritmos são muito sensíveis ao ruído e isso pode afetar a qualidade do *cluster* gerado.

Os algoritmos de *clustering* podem ser classificados em algumas categorias principais: hierárquicos, de particionamento ou realocação interativa, de densidade e de restrições de contiguidade. Essas categorias serão descritas nas próximas seções, incluindo alguns algoritmos que implementam cada uma delas.

### 3.5.1 Método Hierárquico

O método hierárquico cria uma decomposição da base de dados na forma de árvore, dividindo-a recursivamente em conjuntos de dados menores. Essa divisão pode ser feita de duas formas: *top-down* e *bottom-up* [HAN 01]. Alguns autores chamam essa divisão de *divisivos* e *aglomerativos*, respectivamente [NEV 01].

Na abordagem *top-down*, o processo inicia com todos os objetos no mesmo *cluster*, o qual vai sendo dividido sucessivamente, até que cada *cluster* contenha um único elemento. Na forma *bottom-up*, cada objeto é um *cluster* e, a cada passo do procedimento, os dois clusters mais próximos (similares) são fundidos até que, ao final, exista somente um grande *cluster*, contendo todos os objetos. Ele é chamado de hierárquico porque admite obter vários níveis de agrupamento.

O algoritmo hierárquico aglomerativo é executado de acordo com os seguintes passos:

Passo 1: iniciar com  $n$  *clusters*, cada um contendo um objeto.

Passo 2: calcular a dissimilaridade entre os objetos.

Passo 3: procurar o par de *clusters* com menor dissimilaridade.

Passo 4: recalculer a dissimilaridade do *cluster* fundido com os demais *clusters*.

Passo 5: repetir os passo 3 e 4  $n-1$  vezes.

O cálculo da dissimilaridade entre *clusters* no algoritmo hierárquico utiliza um ponto central para representar o *cluster*, definido pelos valores médios dos atributos dos objetos membros de cada grupo. A dissimilaridade entre dois *clusters* é igual a menor dissimilaridade existente entre dois objetos quaisquer, um objeto pertencente a cada um dos *clusters* envolvidos. Uma desvantagem desse método é que ele produz *clusters* de forma alongada, originando o efeito de *encadeamento*.

Embora os métodos hierárquicos sejam empregados com sucesso em aplicações biológicas, por exemplo, não existe uma revisão dos *clusters* durante a execução do procedimento, ou seja, no método hierárquico aglomerativo, uma vez realizada a fusão de dois objetos dentro de um mesmo *cluster*, os objetos não podem mais ser separados, permanecendo no mesmo *cluster* até o final do procedimento. De forma análoga, no hierárquico divisivo, uma vez que dois objetos foram separados, eles nunca mais serão agrupados no mesmo *cluster*.

AGNES e DIANA são dois algoritmos de agrupamento hierárquico. O primeiro é um algoritmo do tipo *bottom-up*, enquanto DIANA é do tipo *top-down*. Em ambos os algoritmos, o número de *clusters* pode ser utilizado como condição de término da execução. Se as decisões de agrupamento ou separação tomadas em cada passo da execução não forem bem escolhidas, os *clusters* gerados podem ser de má qualidade.

*CURE* e *CHAMELEON* são dois algoritmos que utilizam princípios mais complexos de agrupamento e separação. O *CURE* é um algoritmo de *clusterização* que usa princípios mais sofisticados no agrupamento dos *clusters*. Ao invés de utilizar o ponto mais central ou um objeto qualquer para representar o *cluster*, um número de objetos que estão melhor espalhados são selecionados para representar cada *cluster*. A cada passo do algoritmo, dois *clusters* com o par de pontos mais próximos são fundidos em um único *cluster*. Se houver mais de um ponto representativo por grupo, então todos os pontos ou apenas o *centróide* representarão o *cluster*. Essa abordagem permite ao algoritmo *CURE* ajustar geometrias de objetos espaciais. A figura 4 ilustra um exemplo do algoritmo *CURE*.

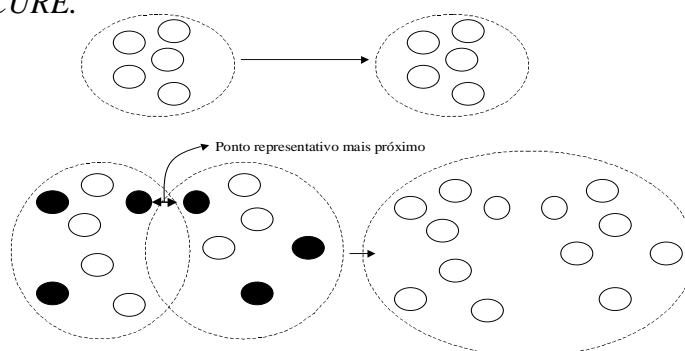


FIGURA 4 – Fundindo dois *clusters* no algoritmo *CURE*

O algoritmo *CHAMELEON* é do tipo *top-down* e busca melhorar a qualidade do *cluster*, usando critérios mais elaborados durante o agrupamento. Dois *clusters* são unidos quando a interconectividade e a proximidade do *cluster* gerado for muito semelhante à conectividade interna e à proximidade dos dois *clusters* antes de sua união. Esse algoritmo é mais eficiente que o *CURE* no que se refere à descoberta de *clusters* com formato arbitrário e densidade variada.

### 3.5.2 Método de Particionamento

Os métodos de particionamento buscam encontrar a melhor partição dos  $n$  objetos em  $k$  grupos. Normalmente os  $k$  *clusters* encontrados são de melhor qualidade do que os  $k$  *clusters* produzidos pelos métodos hierárquicos. Estes métodos apresentam um maior desempenho e por isso os algoritmos que implementam esse método têm sido mais investigados e utilizados [NG 94]. Os métodos de particionamento mais utilizados são baseados em um ponto central (média dos atributos dos objetos – *K-means*) ou em um objeto representativo para o *cluster* (*k-medoids*).

#### 3.5.2.1 O algoritmo *K-Means*

O *k-means* é um algoritmo bastante difundido e muito utilizado em *Sensoriamento Remoto*, com a finalidade de executar procedimentos de classificação não supervisionada de imagens de satélite. Ele exige a definição prévia do número de *clusters* e do posicionamento do centro de cada *cluster*  $k$  no espaço de atributos. O centro do *cluster* é o chamado *centróide*, que é o ponto médio mais central do *cluster*. Esse algoritmo é sensível ao ruído, mas em termos de performance é relativamente eficiente para grandes bases de dados.

Os passos básicos do algoritmo *k-means* são:

Passo 1: seleção de  $n$  objetos para serem centros iniciais dos  $k$  *clusters*.

Passo 2: cada objeto é associado a um *cluster*, para o qual a dissimilaridade entre o objeto e o centro do *cluster* é menor que as demais.

Passo 3: os centros dos *clusters* são recalculados, redefinindo cada um, em função dos atributos de todos os objetos pertencentes ao *cluster*.

Passo 4: retorna ao passo 2 até que os centros dos *clusters* se estabilizem.

A cada interação, os objetos são agrupados em função do centro do *cluster* mais próximo e, por consequência, os centros dos *clusters* são reavaliados (passo 3). Isso provoca no espaço, o deslocamento dos centros médios. O algoritmo é interrompido quando as médias não mais são deslocadas, ou há uma insignificante realocação de objetos entre os *clusters*.

#### 3.5.2.2 O algoritmo *K-Medoids*

A diferença básica em relação ao *k-means* está na utilização de um objeto representativo, chamado *medoid*, localizado mais ao centro do *cluster*, ao invés de um centro médio. Por essa razão, este método é menos sensível ao ruído que o *k-means*. Entretanto, resulta num maior tempo de processamento.

Os objetos são randomicamente selecionados para serem o centro dos *clusters*. O algoritmo de *clustering* PAM (*Partitioning Around Medoids*), baseado em *k-medoid*, realiza a cada passo uma busca exaustiva pela troca de um dos  $k$  *medoids*, previamente

selecionados, por um dos demais ( $n-k$ ) objetos que minimize as dissimilaridades entre os  $k$  *medoids* e os membros dos  $k$  *clusters*.

O algoritmo PAM funciona efetivamente com pequenas bases de dados. Para manipular grandes bases de dados, HAN [HAN 95] sugere o uso do CLARA (*Clustering LARge Applications*). Assim, ao invés de utilizar toda a base de dados, pode-se usar uma amostra da base sobre a qual o algoritmo PAM é aplicado para selecionar os *medoids*. A média de dissimilaridade é calculada sobre toda a base de dados. Dessa forma, várias amostras são coletadas da base de dados e em seguida se aplica o PAM sobre cada amostra. Então, o CLARA pode ser aplicado sobre o melhor *cluster* gerado para cada amostra.

A eficiência do CLARA depende do tamanho da amostra. Enquanto PAM pesquisa o melhor  $k$ -*medoids* numa base de dados, CLARA procura pelo melhor  $K$ -*medoid* entre várias amostras da base de dados.

Para melhorar a qualidade e a escalabilidade do CLARA foi criado o algoritmo CLARANS (*Clustering LARarge Applications based upon RANdomicized Search*) [NG 94]. Na busca pelo melhor objeto central do *cluster*, o CLARANS tenta encontrar a melhor solução, buscando randomicamente o melhor  $K$  central e tentando substituí-lo por um outro objeto, também escolhido randomicamente, entre os demais  $n-k$  objetos. Se nenhuma solução melhor for encontrada, permanece a solução anterior.

### 3.5.2.3 Método Baseado em Densidade

Os métodos baseados em densidade podem ser utilizados para eliminar o ruído dos dados e descobrir *clusters* com formatos arbitrários, separando regiões de objetos de alta e baixa densidade [HAN 01]. Assim, pode ser usado em bases de dados espaciais. Alguns algoritmos baseados nesse método são o *DBSCAN*, o *DENCLUE* e o *OPTICS*.

O *DBSCAN* requer que dois parâmetros iniciais sejam informados: o raio (distância entre um objeto e seus vizinhos) e o objeto central, chamado *MinPts*. Como a qualidade dos *clusters* gerados por esse algoritmo depende desses parâmetros, o usuário é responsável por selecionar os melhores parâmetros possíveis [EST 01].

Os *clusters* gerados pelo *DBSCAN* seguem algumas regras, as quais estão relacionadas abaixo:

- um objeto pertence a um *cluster*  $k$  somente se estiver localizado no raio de algum objeto central do *cluster*;
- um objeto central  $o$ , no raio de um outro objeto central  $o_i$  qualquer, precisa pertencer ao mesmo *cluster*  $k$ ;
- um objeto não central  $o$ , no raio de algum objeto central  $o_1... o_i$ , onde  $i > 0$  precisa pertencer ao mesmo *cluster* cujo objeto central esteja entre  $o_1... o_i$ ;
- um objeto não central  $o$  que não estiver no raio de nenhum objeto central é considerado ruído.

Para gerar os *clusters* o *DBSCAN* testa o raio de cada ponto da base de dados. Se o raio de um objeto  $p$  contém mais de um ponto central (*MinPts*), então um novo *cluster* é criado para o objeto  $p$ . Os objetos no raio de  $p$  são então adicionados ao novo *cluster*. Durante o processo, um objeto central que já pertence a um *cluster* pode ser encontrado

em outro cluster. Nesse caso, os dois *clusters* são agrupados em um só e o processo se encerra quando nenhum novo ponto for adicionado a qualquer *cluster*. A figura 5 ilustra um exemplo desse algoritmo.

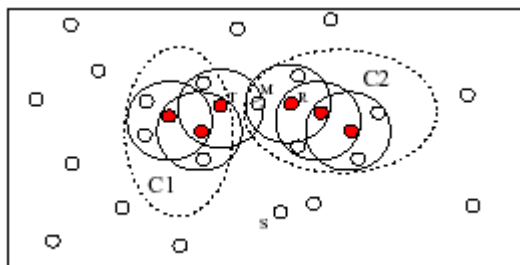


FIGURA 5 – Dois clusters encontrados pelo *DBSCAN*

Embora o *DBSCAN* seja capaz de descobrir *clusters* em dados com ruído ele é sensível aos dois parâmetros de entrada (o raio e o ponto central). Para descobrir os *clusters* ele depende do usuário informar os parâmetros adequados, o que muitas vezes pode consumir muito tempo pela necessidade de executar os passos várias vezes até chegar a um resultado aceitável. Para superar essa dificuldade, o algoritmo *OPTICS* (*Ordering Points To Identify the Clustering Structure*) foi proposto. Ele requer a entrada dos mesmos parâmetros que o *DBSCAN*. Entretanto, em vez de produzir *clusters* com base nos parâmetros raio e ponto central, o *OPTICS* produz um conjunto de dados ordenados, de modo que o resultado do agrupamento para qualquer valor inferior ao raio e similar ao valor do ponto central, possam ser facilmente visualizados e computados.

O *OPTICS* cria um conjunto de objetos ordenados na base de dados, armazenando a distância mais central e a distância mais adequada para cada objeto. Essa informação é suficiente para extrair todos os *clusters* baseados em densidade em relação a qualquer raio  $r$  que seja menor que a distância  $r'$ , usada na geração da ordem dos objetos.

O *DENCLUE* é um algoritmo baseado em um conjunto de funções de distribuição da densidade. Ele funciona da seguinte forma:

- cada ponto é caracterizado por uma função matemática que descreve o impacto ou influência desse ponto dentro da vizinhança;
- a densidade do espaço em questão é representada pela soma dos pontos de influência;
- o *cluster* é definido matematicamente pela identificação dos pontos mais densos.

#### 3.5.2.4 Método baseado em Restrições de Contiguidade

Na maioria dos estudos envolvendo o uso de métodos de *clustering* aplicados à mineração de dados espaciais, é empregada a distância euclidiana<sup>2</sup> para medir as dissimilaridades entre os objetos espaciais. Segundo GORDON [GOR 96], existem três

<sup>2</sup> é a função usada em um Sistema de Coordenadas Cartesianas para medir a distância entre dois pontos P1 e P2 de coordenadas  $(x_1, y_1)$  e  $(x_2, y_2)$ , respectivamente.

abordagens para o agrupamento com restrições de contiguidade espacial. Na primeira, o processo de *clusterização* é realizado considerando, inicialmente, os atributos não-espaciais e, posteriormente, os *clusters* são reavaliados, observando as relações de vizinhança entre os objetos. Assim, objetos similares agrupados em um *cluster* no primeiro estágio, mas sem contiguidade espacial, serão separados no segundo estágio, formando regiões distintas.

Este tipo de abordagem permite identificar, entre os dois estágios, se os objetos similares estão espalhados ou não por toda área de abrangência do estudo, o que pode ser utilizado para uma rápida avaliação da dependência espacial entre os objetos. Outro aspecto importante é que pelo primeiro estágio (regionalização), fica garantido que objetos similares estejam na mesma região. O inconveniente dessa abordagem é a falta de controle sobre o número de regiões resultantes, pois em casos de pequena dependência espacial, haverá tendência a produzir muitas regiões.

Na segunda abordagem, a dissimilaridade dos objetos é avaliada, considerando simultaneamente, a posição geográfica dos objetos e seus atributos não-espaciais. Para isso, são utilizadas as coordenadas do centróide do objeto espacial. Essa abordagem é implementada pelo sistema *SAGE (Spatial Analysis in a GIS Environment)* em seu algoritmo de regionalização. Ele utiliza um procedimento de *clustering* baseado no algoritmo de particionamento *k-means*, cujo objetivo é formado por três critérios: homogeneidade (regiões formadas por objetos similares, considerando atributos descritivos); compacidade (as coordenadas dos objetos membros estão próximas) e igualdade (a soma dos valores de um determinado atributo, considerando todos os objetos membros como, por exemplo, população, são semelhantes em todas as regiões).

Na terceira abordagem a informação espacial é utilizada por dispositivos auxiliares do tipo *grafo ou matriz* para representar a relação de vizinhança entre os objetos. No caso da matriz, denominada matriz de contiguidade, cada elemento indica se dois objetos são contíguos ou não. De forma equivalente, quando são utilizados grafos, cada objeto é representado por um vértice ou nó. A figura 6 ilustra a representação da estrutura espacial de uma matriz *M* e seu grafo correspondente.

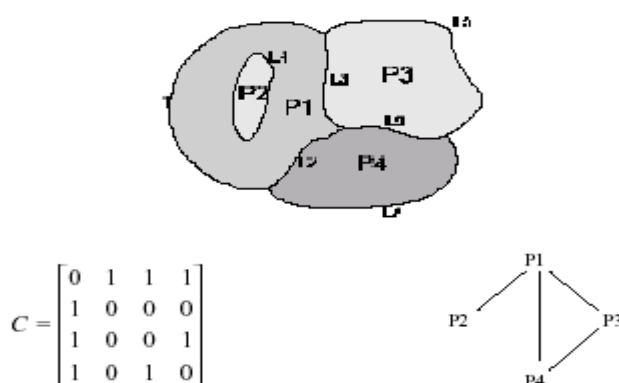


FIGURA 6 – Representação da relação espacial de vizinhança por matriz e grafo

Dentro da terceira abordagem, os algoritmos de *clustering* tradicionais precisarão ser adaptados para o uso em procedimentos com restrição de contiguidade espacial. Nos algoritmos hierárquicos aglomerativos, por exemplo, dois *clusters* mais similares são agrupados somente se existem dois objetos contíguos, um em cada *cluster*.

## 4 Ferramentas para Mineração de Dados Geográficos

Atualmente, existem poucas ferramentas para descoberta de conhecimento em BDG. Alguns algoritmos que implementam a etapa de mineração de dados já foram estendidos para suportar dados espaciais, porém existem poucas ferramentas que implementam esses algoritmos.

Entre as ferramentas encontradas estão o *GeoMiner*, o *Padrão* e o *INGENS*. Cada uma tem características próprias para suportar dados espaciais e não-espaciais.

### 4.1 GeoMiner

O *GeoMiner* é um protótipo de software para descoberta de conhecimento em bases de dados geográficos que foi desenvolvido em 1995, na Universidade de Simon Fraser no Canadá. Ele é uma extensão do *DBMiner* (ferramenta para descoberta de conhecimento em bases de dados convencionais) [HAN 96].

A seleção dos dados no *GeoMiner* é realizada através de uma linguagem de consulta espacial *GMQL* (*GeoMining Query Language*), que é uma extensão da SQL Espacial para mineração de dados espaciais.

```
Discover RAE
inside RS
from road r, wather w, limit l
in relevance to town t
where g_close_to (t.geometry, x.geometry) and x in {r,w,l}
      and t.type="large" and r.type in {highway}
      and w.type in {sea, ocean, river, lake}
      and l.region1 in "RS" and l.region2 in "Brazil"
```

Neste exemplo de consulta espacial, são recuperados todos os grandes rios, lagos, mares, oceanos, estradas e limites municipais das cidades da região RS que pertençam à região Brasil. A variável  $x$  representa as variáveis  $r$ ,  $w$  e  $l$ . O predicado *g\_close\_to* generaliza os relacionamentos espaciais *adjacent\_to*, *containts* e *close\_to*.

A arquitetura do *GeoMiner* é formada por três grandes módulos: uma *interface gráfica* para o usuário interagir com o sistema e visualizar o resultado da mineração de dados; o módulo de *descoberta de conhecimento* e o módulo de *acesso e processamento aos dados espaciais* (veja figura 7).

Através da *interface*, o usuário pode visualizar o resultado da mineração na forma de mapas, gráficos ou tabelas, bem como interagir com o sistema.

A descoberta de conhecimento se dá no *GeoMiner* através cinco sub-módulos [HAN 97]: o *Geo-Characterizer*, o *Geo-Comparator*, o *Geo-Associator*, o *Geo-Classifer* e o *Geo-Cluster-Analyser*. Todos os módulos extraem conhecimento da base de dados através da *GMQL*, a qual é utilizada tanto para selecionar os dados da base espacial como para extrair o conhecimento usando alguma técnica (classificação, generalização, agrupamento, etc) de mineração de dados. Para cada técnica, a *GMQL* oferece comandos específicos, conforme descrito a seguir.

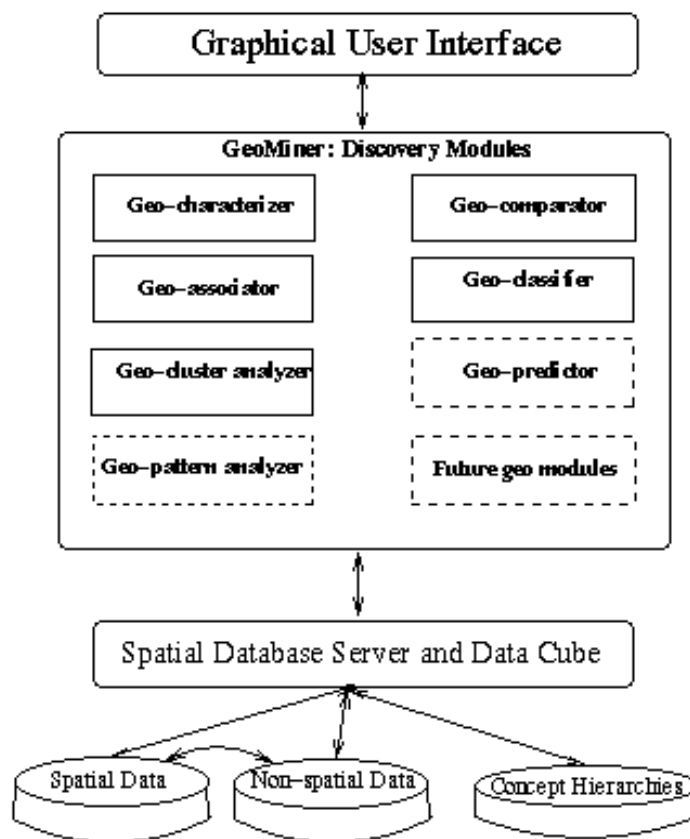


FIGURA 7 – Arquitetura do GeoMiner

O *Geo-Characterizer* tem como principal objetivo extrair conhecimento da base de dados espacial usando a técnica de *generalização* dos atributos espaciais e não-espaciais, representando de forma diferente o processo realizado para cada tipo de dado. A extração de conhecimento desse módulo é dada através da cláusula *characteristics*, conforme ilustra a figura 8.

```

Mine spatial characteristics
As "Brazil.temperature and precipitation distribution"
Analyze geo
in relevance to temperature, precipitation
From weather_probe
Where time_period = "summer" and year=1996 and area_name="Brazil"

```

FIGURA 8 – Exemplo de extração de conhecimento usando *generalização*

O *Geo-Comparator* é baseado na técnica de *classificação*, comparando objetos de uma classe de dados com os objetos de outra classe. A abordagem utilizada compara uma classe (a classe de origem) com as demais classes existentes (classes de contraste). O que diferencia esse módulo do *Geo-Classifier*, é que no primeiro as classes de dados são definidas na consulta e no último as classes são geradas pelo sistema. A cláusula utilizada pela GMQL para comparar as classes é *comparison*, conforme ilustra a figura 9.

No exemplo, o *GeoMiner*, inicialmente, recupera os dados relevantes (que incluem temperatura, precipitação e o nome da região) de duas regiões (Brasil e Argentina) no ano de 1996. Em seguida, os dados coletados são divididos em duas classes de contraste: Brasil e Argentina.

```

Mine spatial comparison
As "Precipitation: Brazil versus Argentina"
For "Brazil"
Where area_name="Brazil"
Versus "Argentina"
Where area_name="Argentina"
Analyze precipitation, geo
In relevance to temperature, area_name
From weather_probe
wheree time period = "july" and year=1996

```

FIGURA 9 – Exemplo de extração de conhecimento usando *classificação*

O *Geo-Associator* busca um conjunto de *regras de associação* espacial que estão implícitas na base de dados. Essas regras definem a dependência probabilística que pode existir entre os dados espaciais ou entre os dados espaciais e os não-espaciais. Elas são construídas com base em predicados espaciais e predicados não-espaciais. Por exemplo: se um estado (Rio Grande do Sul) é adjacente a um grande rio e outro estado (Santa Catarina) também é adjacente a este mesmo rio, pode-se dizer que o rio faz a divisa entre os dois estados, caracterizando uma associação espacial. A cláusula utilizada pela GMQL para encontrar as regras é *associations*, conforme ilustra a figura 10.

```

Mine spatial associations
As "large_PARANAtown"
In relevance to water.name, states.area_name
From towns, water, states, provinces
Where towns.population > 25000 and
    Towns.geo inside provinces.geo and
    Provinces.area_name = "PARANÁ" and
    G_close_to (towns.geo, water.geo, 10 "km") and
    Water.area > 3 and
    G_close_to (towns.geo, states.geo, 75 "km") and
    States.area_name = "Brazil"

```

FIGURA 10 – Exemplo de extração de conhecimento usando *regras de associação*

No exemplo da figura 10, o objetivo é encontrar associações espaciais entre grandes cidades (com população maior que 25 mil habitantes) localizadas no estado do Paraná (que esteja dentro do país Brasil), que tenham grandes rios (com mais de 3 KM de extensão) e que estejam a no máximo 10 km da cidade.

O *Geo-Classifier* é um módulo que implementa árvores de decisão para fazer a *classificação* dos objetos espaciais. Os objetos são classificados pelo sistema com base nos atributos não-espaciais, relacionamentos espaciais, predicados e/ou funções espaciais. A classificação espacial permite encontrar regras para as classes com base na

espacialidade dos objetos e os relacionamentos espaciais entre elas. As relações espaciais entre os objetos são identificadas pelo uso de técnicas de aproximação e vizinhança. As relações encontradas são transformadas em predicados do tipo *perto\_de(x,y)*, que posteriormente são utilizadas na construção das árvores de decisão.

O *Geo-Cluster-Analyser* é uma técnica de *agrupamento* que tem como principal vantagem o fato de permitir a busca de classes de dados, sem que haja um conhecimento prévio do domínio da aplicação. Esse módulo implementa o algoritmo *CLARANS*, criando diferentes *clusters* para os dados espaciais e não-espaciais.

Os módulos Geo-predictor e Geo-pattern analyser não foram implementados.

Segundo HAN em [HAN 97], apenas o *Geo-Cluster-Analyser* foi baseado em um algoritmo específico, o *CLARANS*. Os demais módulos de DCBDG do *GeoMiner* não foram baseados em um algoritmo específico, mas em vários algoritmos que foram estudados pelos pesquisadores do grupo e adaptados para .

## 4.2 Padrão

O *Padrão*, segundo Santos [SAN 01] é um sistema de DCBDG que busca identificar padrões ou relacionamentos implícitos, existentes entre dados geográficos e dados não-geográficos. O *Padrão* foi desenvolvido na Universidade do Minho, em Portugal, em [SAN 01]. Ele é baseado em um sistema de raciocínio espacial qualitativo, que permite a incorporação da componente *espacial* no processo de descoberta de conhecimento. Sua arquitetura é formada por três componentes principais, ilustrados na figura 11.

- *repositório de dados e conhecimento*: este repositório de dados integra três bases de dados: a base de dados geográfica (que armazena os dados espaciais e os relacionamentos entre eles), a base de conhecimento espacial (que armazena os mecanismos de raciocínio espacial qualitativo, que permitem a inferência de relações espaciais desconhecidas) e a base de dados não-geográfica (que contém os dados descritivos, que ao serem integrados com os dados espaciais, permitirão a identificação de padrões e relacionamentos implícitos entre dados geográficos e os dados convencionais);
- *análise de dados*: este componente realiza a análise dos dados armazenados no repositório de dados, aplicando algumas etapas do processo de descoberta de conhecimento (seleção dos dados, tratamento, pré-processamento, processamento da informação geo-espacial, mineração de dados e interpretação dos resultados). A análise de dados foi implementada na ferramenta *Clementine* e possui uma interface externa desenvolvida em *Visual Basic*, que auxilia o *Clementine* no processo de inferência da informação espacial. Esse módulo combina regiões cujas relações espaciais são desconhecidas, mas que serão identificadas pelas regras de inferência qualitativa. Os padrões encontrados na interpretação dos resultados são armazenados numa base de dados de padrões (BDP);
- *visualização dos resultados*: este componente recupera o resultado da descoberta de conhecimento da base de padrões e apresenta-o na forma de mapas, onde cada mapa representa uma determinada região geográfica.

O *GEOMEDIA PROFESSIONAL* da Intergraph [INT 99] foi utilizado para apresentar graficamente o resultado da descoberta.

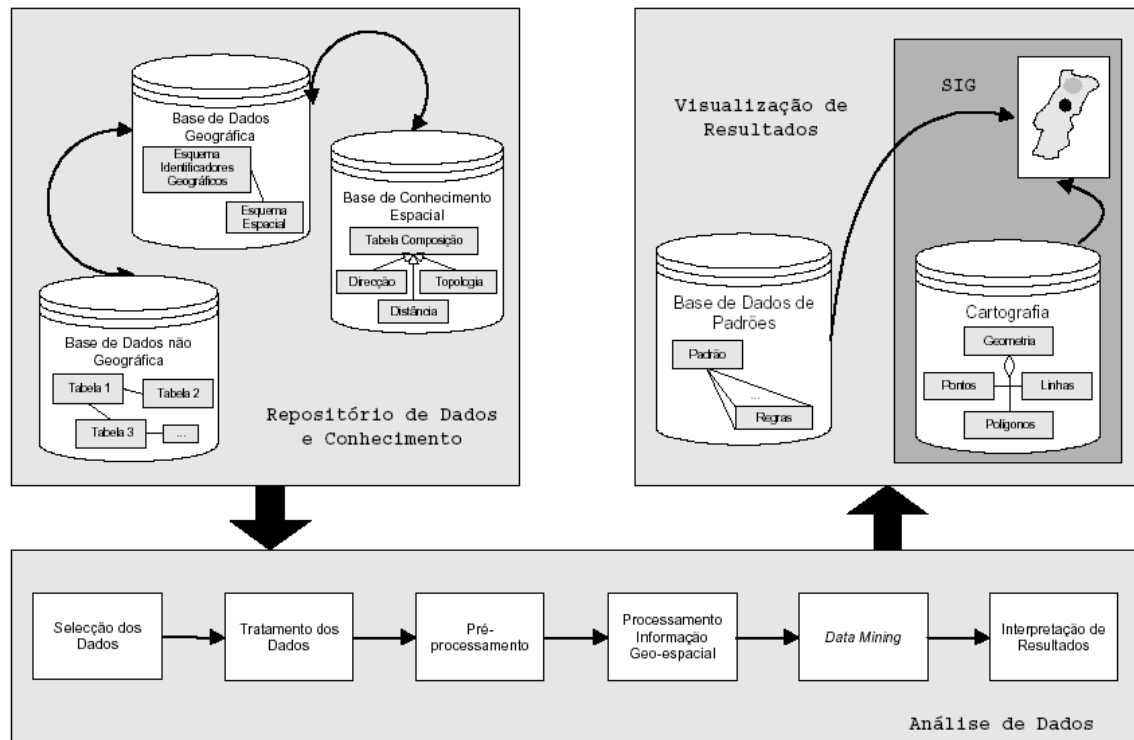


FIGURA 11 - Arquitetura do Sistema Padrão

A ferramenta *Clementine* implementa vários algoritmos de mineração de dados não-espaciais como: o *k-means* (para clusterização), o *APRIORI*, o *C5.0* e o *GRI-Generalised Rule Induction* (para regras de associação). Os diferentes algoritmos podem ser utilizados pelo sistema PADRÃO de acordo com o que se deseja classificar, prever ou associar. Um módulo adicional foi incorporado à ferramenta *Clementine* para minerar dados espaciais, porém o algoritmo empregado não foi mencionado.

O Padrão integra um conjunto de ferramentas, de diferentes fornecedores, e assim como o *GeoMiner*, não possui uma versão para testes.

### 4.3 INGENS

O *INGENS* (*Inductive Geographic information System*) é um protótipo de SIG, desenvolvido em 2000 na Università degli Studi em Bari, na Itália. Ele integra ferramentas de aprendizado de máquina que auxiliam os usuários na interpretação de mapas topográficos. O módulo que implementa um ou mais algoritmos de aprendizado indutivo gera modelos de objetos geográficos a partir de exemplos [MAL 02] [MAL 02a]. A figura 12 ilustra a arquitetura do *INGENS*.

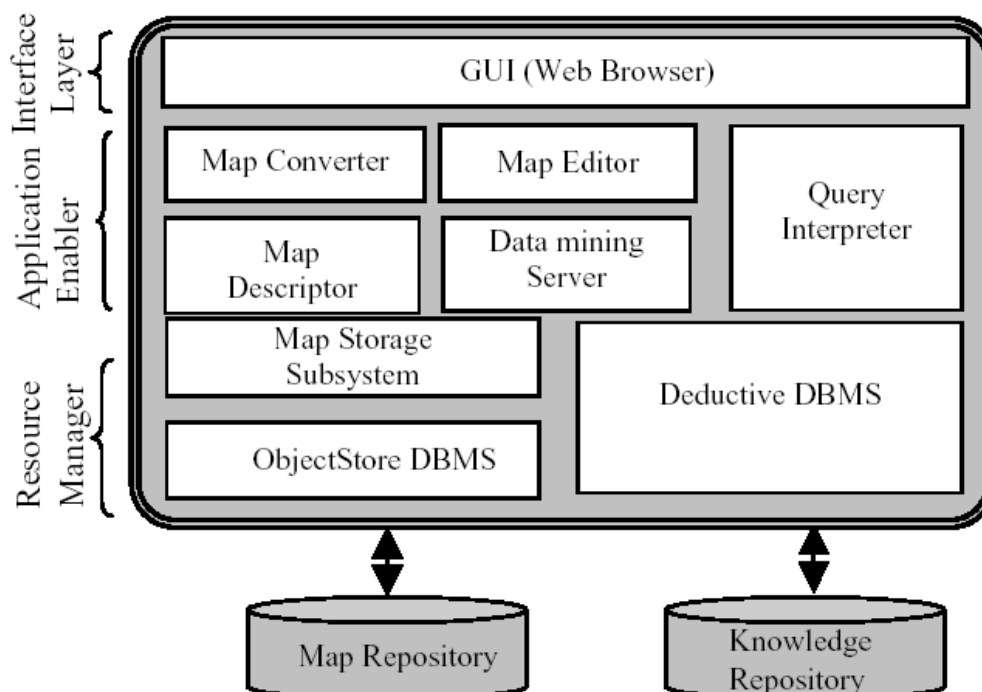


FIGURA 12 – Arquitetura do INGENS em três camadas [MAL 02a]

O GUI é uma interface gráfica através da qual o usuário interage com o INGENS. Essa interface foi construída em *Java* e pode rodar em qualquer *browser web* interoperável com *Java*. Ela pode ser utilizada por usuários de vários níveis:

- *DBA* – administradores de bases de dados;
- *especialistas* – ensinam ao sistema operações sobre objetos geográficos que não estão explicitamente modeladas na base de dados;
- *finais casuais* – acessam a base de dados ocasionalmente, buscando diferentes informações a cada instante. Esses usuários não precisam ter um conhecimento aprofundado sobre o funcionamento do sistema.
- *atualizadores* de mapas – fazem as atualizações no repositório de mapas.

O *Map Descriptor* é uma aplicação que gera automaticamente a descrição lógica de alguns objetos geográficos.

O *Data Mining Server* implementa alguns algoritmos de mineração de dados, que podem ser executados concorrentemente por múltiplos usuários, para descobrir, previamente, padrões úteis e desconhecidos sobre os dados geográficos.

Para fazer consultas no repositório de mapas (*Map Repository*), o *INGENS* oferece o *Query Interpreter*, que realiza a seleção dos dados através da *SDMOQL* (*Spatial Data Mining Object Query Language*), que é uma extensão da linguagem *OQL*, para mineração de dados espaciais. O *Query Interpreter* dispara a consulta ao repositório de mapas após obter a estrutura dos dados através do *Map Descriptor*.

O *Map Converter* é responsável por importar e exportar mapas de vários formatos como, por exemplo, *DXF*.

O *Map Editor* auxilia o *Map Converter* na conversão dos mapas. Esse módulo é necessário porque dependendo dos formatos gerados, algumas informações podem ser perdidas e, por isso, precisam ser ajustadas manualmente, usando um editor de mapas. Por exemplo: os limites de um município podem não estar totalmente conectados após a conversão de um formato de dados para outro. Através do *Map Editor*, o usuário pode editar o mapa no novo formato gerado e fazer a ligação das arestas não conectadas.

O *Map Storage Subsystem* armazena as funções de inclusão, alteração e exclusão usadas na manipulação dos dados. Esse módulo é o único caminho pelo qual o usuário pode acessar os dados geográficos.

O módulo *Deductive DBMS* usa o *XSB-basedDbms* relacional para expressar o conhecimento descoberto e o armazena no *Knowledge Repository*.

O *Object Store DBMS* armazena os atributos descritivos relacionados aos dados geográficos.

### 4.3.1 O Modelo de Objetos do *INGENS*

Os elementos geográficos existentes no mundo real dividem-se, basicamente, em nove categorias: hidrografia, transporte, infra-estrutura, limites, pontos de referência, altimetria, vegetação, localidades e edificações. Através dessas categorias é possível representar qualquer região geográfica.

A mesma região geográfica pode ter várias representações em cada categoria, dependendo da escala. No *INGENS*, os mapas que representam a mesma região geográfica, porém em diferentes escalas, são armazenados separadamente no *Map Repository*, mas estão relacionados entre si. Os mapas estão armazenados no formato raster (como imagens), onde cada mapa é dividido em células quadradas de mesmo tamanho. Cada célula da matriz possui um par de coordenadas x,y e as características daquela célula. Cada célula está relacionada com todas as demais células da matriz.

Através de um modelo de objetos (especificado em *UML-Unified Modeling Language*) [BOO 2000], o *INGENS* permite que o usuário especifique a sua aplicação no nível conceitual, ou seja, o modelo da aplicação. O *INGENS* foi validado com a base cartográfica do Instituto Militar de Geografia (IGMI) da Itália e é muito semelhante ao modelo de objetos da Primeira Divisão de Levantamento do Exército Brasileiro, situado em Porto Alegre.

O modelo topológico, que caracteriza os relacionamentos entre os objetos geográficos, está dividido em duas hierarquias: lógico e físico. Na hierarquia física, os objetos espaciais estão descritos como ponto, linha e polígono. Em mapas de diferentes escalas, o mesmo objeto *rio*, por exemplo, pode ser representado como linha ou polígono. Os objetos do tipo ponto são representados por um par de coordenadas x,y. Uma linha é caracterizada por uma lista de vértices e um polígono por um conjunto de linhas conectadas pelas suas extremidades. O modelo topológico do *INGENS* implementa os relacionamentos topológicos definidos em [EGE 94], que são: *disjoint*, *contains*, *inside*, *covers*, *covered-by*, *overlaps*, *equal*, *crosses* e *touches*.

A hierarquia lógica expressa a semântica dos objetos geográficos, independentemente da sua representação física. Ela é uma generalização das nove categorias citadas anteriormente, também conhecidas, em SIG, como temas. Os

elementos do mundo real são representados como classes e os relacionamentos topológicos são implementados como funções.

### 4.3.2 Minerando Dados no INGENS

Através do módulo *Data Mining Server*, o *INGENS* permite que usuários especialistas ensinem ao sistema como identificar objetos geográficos não explicitamente modelados na base de dados.. Ele é baseado na técnica de *classificação*, permitindo a definição recursiva de conceitos por conta própria e descobrir conceitos de dependência que auxiliam na interpretação de problemas com mapas. Essa técnica é implementada pelo algoritmo *ATRE* (*Apprendimento di Teorie Ricorsive da Esempi*). Também implementa regras de associação espacial.

Embora sinais e símbolos do mapa correspondam a conceitos genéricos de rios e limites municipais, por exemplo, eles são compartilhados pelo criador de mapas e o usuário dos mapas. Outros objetos geográficos relevantes para a descoberta de conhecimento podem não estar explicitamente modelados e, por isso, não serem incluídos no processo de classificação. Neste caso, o aprendizado indutivo pode suportar usuários sofisticados que gerem definições operacionais desses objetos, a partir de uma lista de observações, que são descritas em uma linguagem lógica. Essas observações são interpretadas através de uma série de funções como: *color* (verifica a cor do objetos), *altitude* (busca a altitude do objeto), *distance* (calcula a distância entre dois objetos), *point\_to\_region* (retorna o relacionamento espacial entre um objeto do tipo ponto e outro do tipo polígono), entre outras.

A principal característica do *INGENS* é permitir que o usuário de SIG defina um conjunto de classes para os objetos geográficos de interesse e o próprio SIG aplique as técnicas de aprendizado de máquina. Essas definições podem recuperar novas instâncias da base de dados ou auxiliar na definição de consultas.

Maiores detalhes sobre o *INGENS* e sua estrutura e funcionamento podem ser encontrados em [MAL 02].

## 5 Conclusão

Um caso particular da DCBD diz respeito à exploração de dados referenciados geograficamente, isto é, dados que incluem referências a objetos que têm uma localização com relação a superfície terrestre. A análise destes dados impõe a verificação do aspecto espacial associado aos mesmos (posições relativas, adjacências, direções, distâncias, etc.), e da sua influência nos demais dados explorados, já que um objeto geográfico pode ser afetado pelo comportamento dos objetos vizinhos.

A análise de dados espaciais com o objetivo de descoberta de conhecimento requer a utilização de técnicas específicas, capazes de incluir a semântica espacial, implícita na posição e dimensão dos objetos geográficos. Até o presente momento, estas técnicas têm visado o desenvolvimento de novos algoritmos de DM (ou a adaptação de algoritmos já existentes), e a integração de Sistemas Gerenciadores de Bancos de Dados Geográficos, ou SIG, com ferramentas de descoberta de conhecimento. Estes últimos permitem a manipulação dos dados espaciais, e conseqüente transferência dos resultados para uma ferramenta de descoberta de conhecimento.

Na literatura podem ser encontrados vários algoritmos para mineração de dados espaciais. Grande parte desses algoritmos implementa a técnica de agrupamento, gerando *clusters* de objetos espaciais e não-espaciais com características semelhantes ou que estejam localizados na mesma região geográfica. A maioria dos algoritmos propostos é baseada em agrupamento. O *k-media*, *k-medoid*, *PAM* e vários outros são algoritmos baseados nesta técnica. Isso se deve ao fato do conhecimento extraído de BDG referir-se aos objetos que estão geograficamente próximos, ou apresentarem algum relacionamento espacial do tipo topologia, direção e orientação. Neste caso, a técnica de agrupamento é a que mais se adapta para encontrar objetos espaciais localizados na mesma região geográfica.

Quanto às ferramentas, o *Padrão* não incluiu o desenvolvimento de novos algoritmos de DM adaptados à componente espacial dos dados. Ele busca essencialmente o aproveitamento das capacidades de análise exploratória de dados encontrados pela ferramenta de DCBD *Clementine*.

Segundo Santos [SAN 01], o sistema *Padrão* baseia-se na constatação de que a componente espacial associada aos dados georreferenciados não é incorporada no processo de descoberta de conhecimento, já que os algoritmos tradicionalmente utilizados para explorar os dados, não incluem mecanismos que lhes permitam raciocinar em termos espaciais.

Quanto ao *GeoMiner*, ele é uma extensão do *DBMiner* que extrai conhecimento de bases de dados geográficos através das técnicas de agrupamento, classificação e regras associativas. No entanto, essa ferramenta foi um protótipo desenvolvido em uma tese de doutorado e não sofreu melhorias, não saindo de um protótipo. Os desenvolvedores sequer têm uma versão demo da ferramenta para uma análise mais detalhada.

O *INGENS* é uma arquitetura de software significativamente melhor, pois uma ferramenta de SIG foi adaptada para fazer DCBDG. Foi criado um novo módulo que possibilita fazer mineração de dados geográficos diretamente da base espacial. Esta ferramenta também não está disponível para realizar estudos mais aprofundados.

No que se refere a mineração de dados geográficos, não existe, atualmente, uma ferramenta disponível. O *Padrão* e o *GeoMiner* são dois protótipos que não estão disponíveis, nem no meio comercial, nem no meio acadêmico. O INGENS é um SIG que possui um módulo de mineração de dados encapsulado e para ser usado é necessário adquirir o SIG. Ainda assim, o INGENS é um sistema proprietário e sua licença está vinculada à quantidade de vezes que a mineração é realizada.

Com relação aos algoritmos para mineração de dados geográficos, vários foram propostos, porém não implementados.

## Bibliografia

- [ADR 97] ADRIAANS, P.; ZANTINGE, D. Data Mining. Harlow: Addison-Wesley, 1997.
- [BAS 01] BASSALO, G. H. M. **Integração de Modelos Conceituais para Sistemas de Informação Geográfica voltada à preparação de esquemas de Bancos de Dados Geográficos para utilização em Ferramentas de Descoberta de Conhecimento**. Porto Alegre: PPGC da UFRGS, 2001. (TI-1025).
- [BOG 01] BOGORNY, V. **Incorporando suporte a restrições espaciais de caráter topológico ao modelo abstrato do consórcio Open GIS**. Porto Alegre: PPGC da UFRGS, 2001. Dissertação de Mestrado.
- [AVI 98] AVILA, B. C. Data Mining: ESCOLA REGIONAL DE INFORMÁTICA DA SBC, 6, 1998, Blumenau. **Anais...** Blumenau: SBC, 1998. p. 87-106.
- [BOO 2000] BOOCH, G.; RUMBAUGH, J.; JACOBSON, I.. **The Unified Modeling Language User Guide**. Massachusetts, 1999.
- [BOR 97] BORGES, K. A. V. **Modelagem de dados geográficos: uma extensão do modelo OMT para aplicações geográficas**. Belo Horizonte: Fundação João Pinheiro, 1997. Dissertação de Mestrado.
- [CAM 96] CAMARA, G. et al. **Anatomia de Sistemas de Informação Geográfica**. Campinas, Instituto de Computação, UNICAMP, 1996.
- [CLE 93] CLEMENTINI, E.; DI FELICE, P.; VAN OSTERN, P. A small set of formal topological relationships for end-user interaction. In: ABEL, D; OOI, B.C. (Eds.). **Advances in Spatial Databases**. [S.l.]: Springer-Verlag, 1993. p. 277-295. (Lecture Notes in Computer Science, v. 692)
- [CLE 94] CLEMENTINI, E.; SHARMA, J.; EGENHOFER, M. Modeling topological spatial relations: strategies for query processing. **Computers & Graphics**, Oxford, v.18, n.6, p. 815-822, Nov./Dec. 1994.
- [DOM 01] DOMINGUES, M. L. C. S. **Estudo sobre técnicas de mineração de dados utilizando aprendizado não-supervisionado**. PPGC da UFRGS, 2001. Trabalho Individual
- [EGE 93] EGENHOFER, M. A model for detailed binary topological relationships. **Geomatica**, [S.l.], v.47, n.3-4, p. 261-273, 1993.
- [EGE 94] EGENHOFER, M.; CLEMENTINI, E.; FELICE, P. di. Topological relations between regions with holes. **International Journal of Geographical Information Systems**, [S.l.], v.8, n.2, p. 129-144, 1994.
- [EST 97] ESTER, M.; KRIEGEL, J.; SANDER, J.; XU, X. Spatial data mining: a database approach. **Proceedings...** International Symposium on Large Spatial Databases, Berlin, Germany, p.47-66, 1997.

- [EST 98] ESTER, M.; KRIEGEL, H. P.; SANDER, J.; XU, X. Clustering for mining in large spatial databases. *KI-Journal, Special Issue on Data Mining*, 1:18-24,1998.
- [EST 01] ESTER, M.; KRIEGEL, H.; SANDER, J. **Algorithms and applications for spatial data mining**. Geographic Data Mining and Knowledge Discovery, Taylor and Francis, 2001.
- [FAR 98] FARIA, G. **Um Banco de dados espaço-temporal para desenvolvimento de aplicações em Sistemas de Informação Geográfica**. Campinas: Instituto de Computação – Universidade Estadual de Campinas, 1998. Dissertação de Mestrado.
- [FAY 96] FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. **From data mining to discovery knowledge in databases**. *AI Magazine*, 3(17): 37-54. 1996.
- [FEL 97] FELDENS, M. A. **Engenharia da descoberta de conhecimento em bases de dados: estudo e aplicação na área da saúde**. Porto Alegre: PPGC da UFRGS, 1997. (Dissertação de Mestrado).
- [GOR 96] GORDON, A.D. A survey of constrained classification. *Computational Statistics & Data Analysis*, v. 21, n., p. 17-29, 1996.
- [GUT 94] Güting, H. R. An Introduction to Spatial Database Systems. **VLDB Journal**, [S.l.], v.10, n.4, p. 357-399, 1994.
- [HAL 2000] HALMENSCHLAGER, C. **Utilização de agentes na descoberta de conhecimento**. Porto Alegre: PPGC da UFRGS, 2000. (TI-955).
- [HAN 01] HAN, J.; KAMBER, M.; TUNG, A. K. H. Spatial clustering methods in data mining: a Survey. School of Computing Science, Simon Fraser University, Burnaby, BC Canada, v5a156, 2001.
- [HAN 01a] [Han e Kamber, 2001] J. Han e M. Kamber. *Data Mining: Concepts and Techniques*. Morgan Kaufmann Publishers, 2001.
- [HAN 94] HAN, J.; FU, Y.; HUANG, Y.; CAI, Y.; CERCONE, N. DBLearn: a system prototype for knowledge discovery in relational databases. **In Proceedings...** on ACM SIGMOD International Conference on Management of Data, 516. ACM, 1994.
- [HAN 95] HAN J.; FU Y. Discovery of multiple-level association rules from large databases. **Proceedings...** Int. Conference Very Large Data Bases, p. 420-431, Zurich, Switzerland, 1995.
- [HAN 96] HAN J. et. Al. DBMiner: a system for mining knowledge in large relational databases. **Proceedings...** Int. Conference on Data Mining and Knowledge Discovery (KDD'96), p.250-255, Portland, Oregon, 1996.

- [HAN 97] HAN J.; KOPERSKI, K.; STEFANVIC, N. GeoMiner: a system prototype for spatial data mining. **Proceedings...** ACM-SIGMOD Int. Conference on Management of Data (SIGMOD'97), Tucson, AR, 1997.
- [INT 99] [Intergraph, 1999b] Intergraph. Geomedia Professional v3, Reference Manual. Intergraph Corporation, 1999.
- [KOP 95] KOPERSKI, K.; HAN, J. Discovery of Spatial Association Rules in Geographic Information Databases, 1995. **Proceedings...** 4<sup>th</sup> Int. Symp. On Large Spatial Databases (SSD'95), Portland, ME, p.47-66.
- [KOP 96] KOPERSKI, K.; ADHIKARY, J.; HAN, J. Spatial Data Mining: Progress and Challenges. **In Proceedings...**SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, Montreal, 1996.
- [KOP 97] KOPERSKI, K.; HAN, J.; ADHIKARI, J. Mining knowledge in geographical data. In COMM. ACM, 1997.
- [LIS 97] LISBOA, F. J. Modelos Conceituais de Dados Para Sistemas de Informações Geográficas. Exame de Qualificação. Porto Alegre: CPPGCC da UFRGS, 1997.
- [LU 93] LU, W.; HAN, J.; OOI, B. C. Discovery of general knowledge in large spatial databases. **In Proceedings...** Far East Workshop on Geographic Information Systems, 275-289, Singapura, 1993.
- [MAL 02] MALBRA, D.; ESPOSITO, F. LANZA, F. LISI, F. A.; APPICE, A. Empowering a GIS with inductive learning capabilities: the case of INGENS. PERGAMON, Università degli studi, Bari, Italy, 2002.
- [MAL 02a] MALEBRA, D., APPICE, A., VACCA, N. SDMOQL: An OQL-based Data Mining Query Language for Map Interpretation Tasks. **Proceedings...** Workshop on Database Technologies for Data Mining (DTDM'02) , in conjunction with the VIII International Conference on Extending Database Technology (EDBT'02), Prague, Czech Republic, 25-27, 2002
- [NEV 01] NEVES, M. C.; FREITAS, C. C.; CAMARA, G. **Mineração de Dados em Grandes Bancos de Dados Geográficos**. INPE, 2001. Relatório Técnico.
- [NG e HAN 94] NG, R. T.; HAN, J. Efficient and Effective Clustering Methods for Spatial Data Mining. In: Twentieth International Conference on Very Large Data Base, Santiago, 1994.
- [PAR 98] [PAR 98] PARENT, C. et al. Modeling spatial data in the MADS conceptual model. In: INTERNATIONAL SYMPOSIUM ON SPATIAL DATA HANDLING, 1998. **Proceedings...** Canada: [s.n.], 1998.
- [QUI 86] QUINLAN, J. R. Induction of decision trees. Machine Learning, n 1, p. 81-106, 1986.

- [SOU 98] SOUZA, M. S. et al. Data Mining: a database perspective. In: INTERNATIONAL CONFERENCE ON DATA MINING, 1998, Rio de Janeiro: COPPE/UFRJ, 1998.
- [SAN 01] SANTOS, M. Padrão: um sistema de descoberta de conhecimento em bases de dados georreferenciadas. Universidade do Minho, 2001. Tese de doutorado.
- [SIL 97] SILVA, N. da. Introdução aos algoritmos genéticos. In: I Oficina de Inteligência Artificial, 1997.