COPPE/UFRJ

METODOLOGIA PARA MINERAÇÃO DE REGRAS DE ASSOCIAÇÃO MULTINÍVEIS INCLUINDO PRÉ E PÓS-PROCESSAMENTO

Custódio Gouvêa Lopes da Motta

Tese de Doutorado apresentada ao Programa de Pós-graduação em Engenharia Civil, COPPE, da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários à obtenção do título de Doutor em Engenharia Civil.

Orientador: Nelson Francisco Favilla Ebecken

Rio de Janeiro
Março de 2010

METODOLOGIA PARA MINERAÇÃO DE REGRAS DE ASSOCIAÇÃO MULTINÍVEIS INCLUINDO PRÉ E PÓS-PROCESSAMENTO

Custódio Gouvêa Lopes da Motta

TESE SUBMETIDA AO CORPO DOCENTE DO INSTITUTO ALBERTO LUIZ COIMBRA DE PÓS-GRADUAÇÃO E PESQUISA DE ENGENHARIA (COPPE) DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE DOUTOR EM CIÊNCIAS EM ENGENHARIA CIVIL.

Examinada por:

_____
Prof. Nelson Francisco Favilla Ebecken, D.Sc.

_____
Prof. Alexandre Gonçalves Evsukoff, Dr.

_____
Prof. Carlos Cristiano Hasenclever Borges, Ph.D.

_____
Dr. Gilberto Carvalho Pereira, D.Sc.

_____
Profª. Myriam Christina de Aragão Costa, D.Sc.

RIO DE JANEIRO, RJ - BRASIL
MARÇO DE 2010

Motta, Custódio Gouvêa Lopes da

Metodologia para Mineração de Regras de Associação Multiníveis Incluindo Pré e Pós-Processamento/ Custódio Gouvêa Lopes da Motta – Rio de Janeiro: UFRJ/COPPE, 2010.

XIII, 90 p.: il.; 29,7 cm.

Orientador: Nelson Francisco Favilla Ebecken

Tese (doutorado) – UFRJ/ COPPE/ Programa de Engenharia Civil, 2010.

Referencias Bibliográficas: p. 75-77.

1. Mineração de Dados. 2. Regras de Associação. 3. Visualização de Dados. I. Ebecken, Nelson Francisco Favilla. II. Universidade Federal do Rio de Janeiro, COPPE, Programa de Engenharia Civil. III. Titulo.

À meu filho Lucas.

AGRADECIMENTOS

Ao Professor Nelson Francisco Favilla Ebecken pela amizade, ensinamentos e orientação precisa e objetiva.

Ao CNPq – Conselho Nacional de Desenvolvimento Científico e Tecnológico e à UFJF – Universidade Federal de Juiz de Fora pelo indispensável suporte financeiro.

Aos Professores Alexandre Gonçalves Evsukoff, Gilberto Carvalho Pereira, Carlos Cristiano Hasenclever Borges e Myriam Christina de Aragão Costa por aceitarem participar da Banca de avaliação desta tese.

Ao empresário Silvino de Castro Matos e ao analista de sistemas Eduardo Lúcio Alves Martins pela disponibilização dos dados e esclarecimentos relevantes para realização das aplicações práticas deste trabalho.

Ao Professor Vinicius Gomes Fortes e aos meus ex-alunos Victor Marques de Assis e Vinícius Vieira pelo competente apoio técnico.

À minha mulher Rachel pelo incentivo permanente e compreensão pelas horas extras dispensadas a este trabalho.

Aos meus colegas da UFJF, em especial aos Professores Ever, Hélio, Marcelo Bernardes, Raul, Rubens e Wilhelm, pelo incentivo permanente.

Aos meus parentes, especialmente meus irmãos Marta, Margô e Bráulio que sempre mostraram sua preocupação e incentivo para a conclusão deste trabalho.

Aos meus amigos Carlos Carreira e Tadeu pelo incentivo paralelo, que tornou possível a continuidade e conclusão desta tese.

Resumo da Tese apresentada à COPPE/UFRJ como parte dos requisitos necessários para a obtenção do grau de Doutor em Ciências (D.Sc.)

METODOLOGIA PARA MINERAÇÃO DE REGRAS DE ASSOCIAÇÃO MULTINÍVEIS INCLUINDO PRÉ E PÓS-PROCESSAMENTO

Custódio Gouvêa Lopes da Motta

Março/2010

Orientador: Nelson Francisco Favilla Ebecken

Programa: Engenharia Civil

O presente trabalho tem por objetivo desenvolver um sistema inteligente para apoiar a geração e a pós-análise das regras de associação entre itens selecionados de uma base de dados, através da projeção desses itens no espaço bidimensional. Foi criada uma função para o cálculo das distâncias entre pares de itens, bem como métodos de hierarquização e otimização dessas distâncias em estruturas de dados, que viabilizam a geração de projeções com boa qualidade visual. Uma projeção alternativa complementar apresenta os grupos formados pelas estruturas hierárquicas. Esse sistema foi incluído em uma metodologia iterativa proposta para mineração de regras de associação, que usa múltiplos níveis de abstração dos itens e abrange desde a pré-análise da base de dados, até o pós-processamento das regras descobertas. A aplicação da metodologia sobre uma base de dados real mostrou a importância de cada um de seus sistemas componentes na construção das possíveis decisões em relação aos diversos aspectos do problema.

Abstract of Thesis presented to COPPE/UFRJ as a partial fulfillment of the requirements for the degree of Doctor of Science (D.Sc.)

METHODOLOGY FOR MINING MULTILEVEL ASSOCIATION RULES INCLUDING PRE AND POST-PROCESSING

Custódio Gouvêa Lopes da Motta

Março/2010

Advisor: Nelson Francisco Favilla Ebecken

Department: Civil Engineering

This study aims to develop an intelligent system to support the generation and post-analysis of association rules between items selected from a database, through the projection of these items in two-dimensional space. A function was created to calculate distances between pairs of items as well as methods of ranking and optimization of these distances in data structures that enable the generation of projections with good visual quality. A complementary alternative projection shows the groups formed by hierarchical structures. This system was included in an iterative methodology proposed for mining association rules, which uses multiple levels of abstraction of the items and ranges from pre-analysis of the database until the post-processing of discovered rules. Applying the methodology on a real database showed the importance of each of its component systems in the construction of possible decisions on the various aspects of the problem.

ÍNDICE

FIGURAS

TABELAS

LISTAGENS

CAPÍTULO 1
INTRODUÇÃO

A descoberta de conhecimento em bases de dados, também chamada de KDD (*Knowledge Discovery in Databases*), pode ser definida como o processo de identificação de padrões embutidos nos dados. Além disso, os padrões identificados devem ser válidos, novos, potencialmente úteis e compreensíveis (FAYYAD, PIATETSKY-SHAPIRO & SMITH, 1996a).

As pesquisas relativas a este processo ganharam rápido crescimento a partir da última década, motivadas pela evolução da tecnologia que vem permitindo a coleta, o armazenamento e o gerenciamento de quantidades cada vez maiores de dados (FAYYAD, PIATETSKY-SHAPIRO & SMITH, 1995, 1996b).

Outra razão deste crescimento é a ampliação das áreas de aplicações de KDD. Como exemplos de áreas de aplicações, podem ser citadas (CUROTTO, 2003): bancária (aprovação de crédito), ciências e medicina (descoberta de hipóteses, predição, classificação, diagnóstico), comercialização (segmentação, localização de consumidores, identificação de hábitos de consumo), engenharia (simulação e análise, reconhecimento de padrões, processamento de sinais e planejamento), financeira (apoio para investimentos, controle de carteira de ações), gerencial (tomadas de decisão, gerenciamento de documentos), *Internet* (ferramentas de busca, navegação, extração de dados), manufatura (modelagem e controle de processos, controle de qualidade, alocação de recursos) e segurança (detecção de bombas, icebergs e fraudes).

O processo de descoberta de conhecimento em base de dados envolve diversas etapas, destacando-se a seguinte seqüência (FAYYAD, PIATETSKY-SHAPIRO & SMITH, 1996a):

1. Consolidação de dados: onde os dados são obtidos a partir de diferentes fontes (arquivos texto, planilhas ou bases de dados) e consolidados numa única fonte.

2. Seleção e pré-processamento: nesta etapa, diversas transformações podem ser aplicadas sobre os dados, como reduzir o número de amostras, de

atributos ou de intervalos de atributos, normalizar valores etc., de forma a obter, no final, um conjunto de dados preparados para utilização dos algoritmos de mineração.

3. Mineração de dados ou DM (*Data Mining*): é a etapa de extração de padrões propriamente dita, onde, primeiramente, é feita a escolha da tarefa de mineração conforme os objetivos desejáveis para a solução procurada, isto é, conforme o tipo de conhecimento que se espera extrair dos dados. A Figura 1.1 ilustra as tarefas de mineração organizadas em atividades preditivas e descritivas.

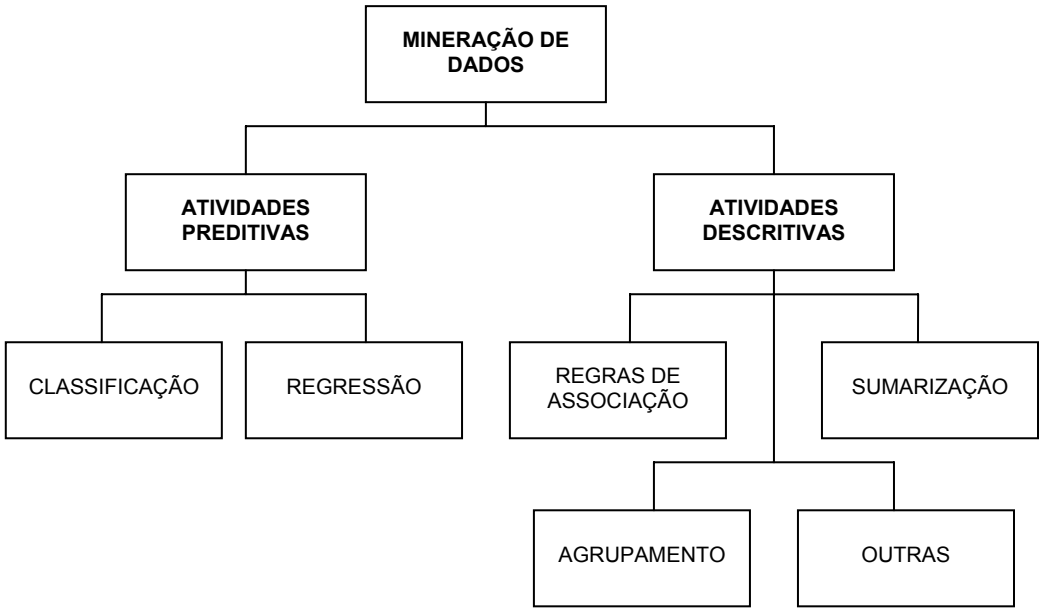


Figura 1.4. Tarefas de Mineração de Dados (REZENDE *et al*, 2003)

As atividades preditivas buscam identificar a classe de uma nova amostra de dados, a partir do conhecimento adquirido de um conjunto de amostras com classes conhecidas. Já as atividades descritivas trabalham com um conjunto de dados que não possuem uma classe determinada, buscando identificar padrões de comportamento comuns nestes dados.

Em seguida, é escolhido o algoritmo que atenda a tarefa de mineração eleita e que possa representar satisfatoriamente os padrões a serem encontrados. Os algoritmos de mineração mais comuns são: Algoritmos Estatísticos,

Algoritmos Genéticos, Árvores de Decisão, Regras de Decisão, Redes Neurais Artificiais, Algoritmos de Agrupamento e Lógica *Fuzzy*.

A mineração de dados é, na verdade, uma atividade interdisciplinar pela diversidade de tecnologias que podem estar envolvidas. A Figura 1.2 sintetiza as disciplinas envolvidas com DM.

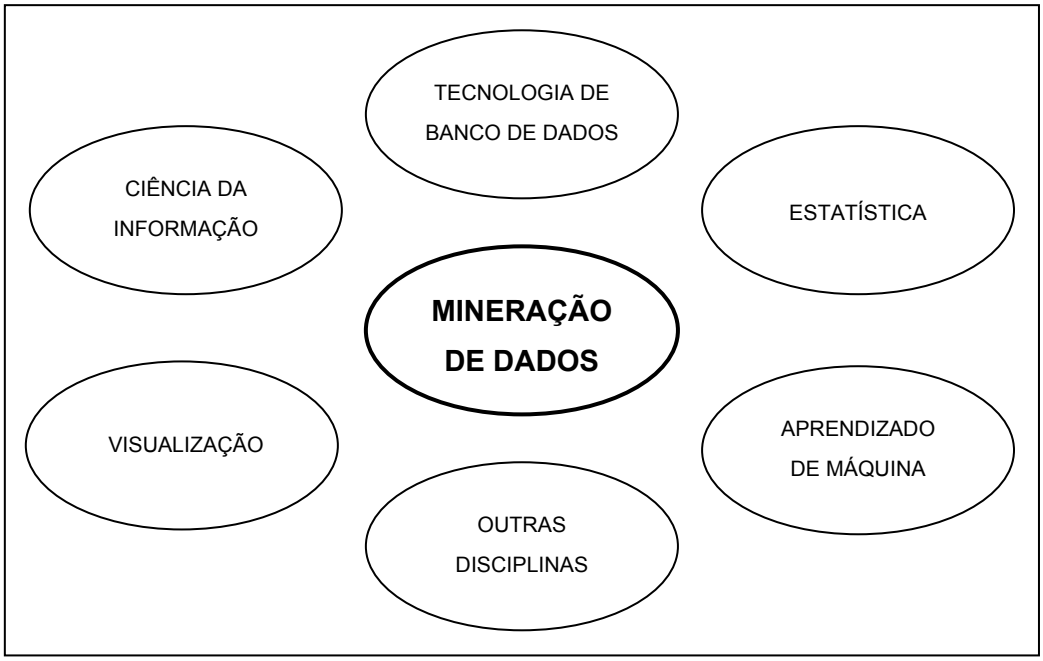


Figura 1.5. Disciplinas envolvidas com Mineração de Dados (HAN & KAMBER, 2006)

4. Avaliação e interpretação: nesta etapa são avaliados o desempenho do processo e a qualidade dos padrões extraídos, bem como verificada a facilidade de interpretação desses padrões.

Deve-se destacar que o processo de KDD ocupa apenas uma posição no ciclo de solução do problema, não se esgotando por si só. Este ciclo é também conhecido como ciclo virtuoso e é apresentado na Figura 1.3.

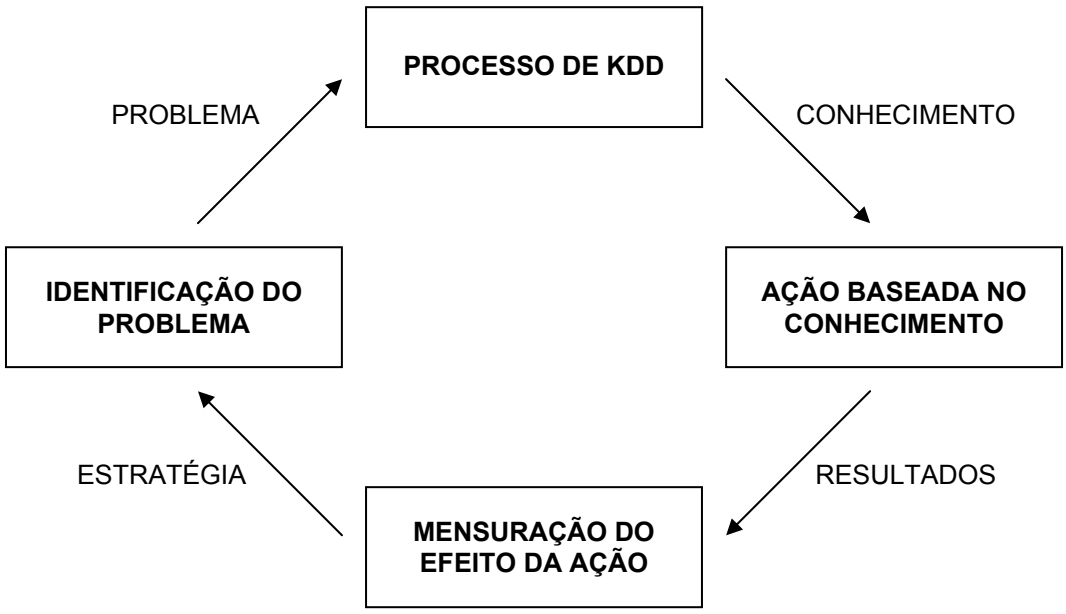


Figura 1.6. Ciclo Virtuoso (SILVER, 1998)

A utilização do conhecimento obtido no processo de KDD é realizada através de um sistema inteligente ou de um ser humano como forma de apoio à tomada de decisão.

Entende-se como inteligente um sistema computacional que possui habilidades inteligentes e sabe como elas modelam tarefas específicas. Entre essas habilidades, está a de usar conhecimento para resolver problemas (REZENDE, 2003).

O objetivo principal desta tese é o desenvolvimento um sistema inteligente que utiliza um método de cálculo das distâncias entre itens selecionados de uma base de dados de transações, criando estruturas de dados que possibilitam a visualização das associações entre esses itens no espaço bidimensional, como forma de apoio à pós-análise do usuário (seja um profissional em mineração ou um especialista do domínio).

Os mapas gerados pela projeção dos itens como pontos no plano facilitam muito a interpretação do usuário para solução de diversos tipos de problemas, desde os mais técnicos, como a identificação dos itens envolvidos em regras de associação que possam ser realmente interessantes ou a análise dos vizinhos mais próximos, até outros mais práticos, como o planejamento da planta baixa de uma loja, a melhor organização dos itens nas prateleiras, a confecção de catálogos de produtos etc.

Para a geração dos mapas foi necessária a criação de uma hierarquização das distâncias nas estruturas de dados, dirigida pelo usuário e baseada nas freqüências de

ocorrências dos itens na base de dados. Adicionalmente, foi desenvolvida uma otimização das distâncias dos itens pertencentes a cada estrutura hierárquica distinta, permitindo uma maior clareza na visualização dos itens.

Entretanto, as dificuldades encontradas na pesquisa com uma base de dados real, tornaram necessária a tarefa de pré-análise dos dados e o desenvolvimento de mais dois sistemas, sendo um para preparação dos dados e outro para edição dos itens de interesse organizados em múltiplos níveis de abstração (edição multinível).

O produto final resultou em uma metodologia para mineração de regras de associação multiníveis constituída de duas etapas: na primeira é realizada a pré-análise e a preparação dos dados e, na segunda, podem ser executadas as tarefas de edição multinível, mineração das regras de associação, cálculo das distâncias, mapeamento da projeção dos itens no espaço bidimensional e pós-análise através da exploração do mapa e de análise subjetiva. Com essa subdivisão, fica facilitada a execução iterativa da segunda etapa, onde, dependendo dos objetivos do usuário, pode-se omitir a realização de algumas de suas tarefas.

A relevância do presente trabalho reside em concorrer para o atendimento de duas necessidades claras da mineração de dados e, em especial, da mineração de regras de associação:

1ª. O processo de mineração de dados como um todo (constituído de pré-processamento, mineração de dados e pós-processamento) é quase sempre iterativo. Geralmente, muitas execuções precisam ser realizadas para conseguir resultados finais satisfatórios, que depois são incorporados às tarefas operacionais do mundo real (LIU, 2008).

2ª. Embora, desde a introdução formal de mineração de regras de associação, a maioria dos trabalhos publicados tem-se concentrado em criar técnicas mais rápidas e eficazes para geração das regras, pouco tem sido feito, relativamente, para investigar, em profundidade, as implicações das análises dessas regras (NICHOLAS & ZHAO, 2009).

Neste sentido, as principais contribuições desta tese são:

1ª. A proposta de uma metodologia iterativa e flexível, que vai desde a pré-analise dos dados, passa pela geração de regras de associação multiníveis e chega até a uma nova forma gráfica para apoiar a pós-analise.

2ª. Um método de cálculo das distâncias entre itens, seguido de uma organização hierárquica e uma otimização dessas distâncias nas estruturas de dados. Esse método foi criado para viabilizar um mapeamento coerente da projeção dos itens no plano, o que permite diversas formas de exploração e em muito facilita a pós-análise.

Os assuntos tratados nesta tese estão organizados da seguinte forma:

No Capítulo 2, é apresentado um estudo geral da atividade descritiva de mineração de regras de associação.

O Capítulo 3 trata das questões relativas à visualização de regras de associação. Inicia com uma apresentação das saídas gráficas de alguns sistemas de mineração de regra de associação, resume dois métodos de projeção multidimensional e finaliza descrevendo as características do sistema de projeção escolhido para ser usado nas aplicações desta tese.

A criação de uma função de distância e os algoritmos de geração e de transformação (hierarquização, otimização e agrupamento) das estruturas de dados que armazenam as distâncias calculadas são detalhados no Capítulo 4.

O Capítulo 5 propõe uma metodologia iterativa para mineração de regras de associação multiníveis e descreve os sistemas que a constituem, realizando um estudo de caso sobre uma base de dados real.

As considerações finais do trabalho desenvolvido são descritas no Capítulo 6.

Como complemento, dois anexos são incluídos para apresentar as imagens geradas para o ajuste da hierarquização e da otimização das projeções de itens no R^2.

CAPÍTULO 2
REGRAS DE ASSOCIAÇÃO

Mineração de regras de associação é talvez o mais importante modelo inventado e amplamente estudado pelas comunidades de banco de dados e de mineração de dados. Seu objetivo é encontrar todas as relações de co-ocorrências entre itens de dados, chamadas associações (LIU, 2008).

Este capítulo apresenta uma visão geral da tarefa descritiva de mineração de regras de associação, desde os conceitos iniciais, passando pelo funcionamento do algoritmo considerado o mais importante da área, descrevendo, em seguida, os tipos de regras de associação e finalizando com uma discussão de alguns recursos que podem facilitar o trabalho do minerador, tanto na obtenção das regras, quanto na identificação daquelas mais interessantes.

Embora essa mineração venha sendo aplicada, atualmente, em bases de dados das mais diversas áreas (medicina, indústria, segurança, engenharia etc.), serão utilizados exemplos da área do comércio, que é considerada a sua aplicação mais tradicional e ficou conhecida como análise de cestas de compras. O objetivo principal dessa análise é descobrir quais produtos (itens) são comprados juntos pelos consumidores.

2.1 Conceitos Básicos

A seguir, é apresentada uma seqüência de conceitos básicos, amplamente utilizada na literatura sobre o assunto, que formaliza o problema de mineração de regras de associação (AGRAWAL, IMIELINSKI & SWAMI, 1993, AGRAWAL & SRIKANT, 1994, HAN & KAMBER, 2006, LIU, 2008).

Seja a base de dados $T = \{t_1, t_2, ..., t_n\}$ contendo um conjunto de n transações.

Seja, agora, o conjunto de m itens $I = \{i_1, i_2, \mathbf{...}, i_m\}$ disponíveis para constituir cada transação $t_i \in T$, tal que $t_i \subseteq I$.

Um conjunto de itens é chamado de um *itemset*. Se um *itemset* possui k itens, ele é um *k-itemset*.

Sejam dois *itemsets* A e B, tais que $A \subseteq I$ e $B \subseteq I$ e não possuem itens em comum, isto é, $A \cap B = \emptyset$. Diz-se que uma transação t_i contem, por exemplo, o *itemset* A se e somente se $A \subseteq t_i$.

Uma regra de associação é uma implicação da forma: $A \rightarrow B$, onde $A \subseteq I$, $B \subseteq I$ e $A \cap B = \emptyset$. Neste caso, lê-se A implica em B, onde A é chamado antecedente e B é o conseqüente da regra.

O objetivo geral da mineração é encontrar, a partir do conjunto de transações T, todas as regras que associem a presença de um *itemset*, A, por exemplo, com qualquer outro (B, C etc.). Entretanto, como I possui um total de m itens, o espaço de busca para todas as regras é, teoricamente, exponencial ($O(2^m)$), pois todos os itens podem constituir *itemsets*.

Na prática, nem todos os itens de I estão presentes nas transações de T e outros, ocorrem em um número muito baixo de transações. Essa esparsialidade é aproveitada pelos métodos de mineração, tornando-os viáveis e eficientes.

Neste sentido, foram criadas duas medidas para as regras, conhecidas como suporte e confiança, que são calculadas a partir da freqüência de ocorrência dos *itemsets* envolvidos na regra.

A freqüência de um *itemset*, também conhecida como contagem ou contagem de suporte, denotada por c, é o número de transações de T que contêm este *itemset*.

O suporte S de uma regra de associação, $A \rightarrow B$, é a percentagem de transações que contêm $A \cup B$ (ou A e B) em relação ao total de transações n de T. O suporte, então, pode ser visto como a probabilidade de ocorrência do *itemset* $A \cup B$ em T e é assim calculado:

$$S(A \rightarrow B) = P(A \cup B) = \frac{c(A \cup B)}{n}. \qquad (2.1)$$

O suporte indica, portanto, a freqüência relativa das regras, podendo ser usado para compará-las, isto é, regras com valores altos para o suporte podem ser interessantes e merecem a atenção, por se destacarem quantitativamente das demais. Por outro lado, as de suporte muito baixo, podem representar somente uma ocorrência ao acaso. Em última análise, o suporte representa a aplicabilidade da regra.

A confiança C de uma regra de associação, $A \rightarrow B$, é a percentagem de transações que contêm $A \cup B$ (ou A e B) em relação a todas as transações de T que contêm A. Neste caso, C é dado pela probabilidade condicional $P(B|A)$ ou probabilidade de ocorrência de B, quando A ocorre. Seu cálculo pode ser feito da seguinte forma:

$$C(A \rightarrow B) = P(B \mid A) = \frac{P(A \cup B)}{P(A)} = \frac{c(A \cup B)}{c(A)}. \qquad (2.2)$$

A confiança indica a capacidade de predição das regras. As regras com valores altos de confiança se destacam qualitativamente das demais, pelo nível de certeza de ocorrência do conseqüente da regra, a partir dos casos onde o seu antecedente ocorre. Já as regras com confiança baixa não fornecem segurança de predição e, por isso, são de uso limitado.

É bastante comum nos algoritmos de mineração de regras de associação a adoção de limites pré-estabelecidos pelo usuário para o suporte e para a confiança, conhecidos, respectivamente, como suporte mínimo (*sup-min*) e confiança mínima (*conf-min*), reduzindo a amplitude do problema, que, desta forma, passa ser assim enunciado:

Dada uma base de dados T contendo um conjunto de n transações, o problema de mineração consiste em descobrir todas as regras de associação fortes em T.

Uma regra de associação forte é aquela que possui suporte e confiança maiores ou iguais, respectivamente, aos limites pré-estabelecidos pelo usuário de suporte mínimo e de confiança mínima.

Após a mineração, é comum apresentar as regras fortes com o seguinte formato:

$A \rightarrow B$ [suporte, confiança]

Um *itemset* freqüênte é aquele que satisfaz ao suporte mínimo, isto é, sua freqüência é maior ou igual ao produto do *sup-min* pelo total de transações de *T*.

Um conjunto de *k-itemsets* é denotado por C_k e um conjunto de *k-itemsets* freqüentes, por L_k.

2.2 Algoritmo Apriori

O algoritmo Apriori representa um dos maiores avanços na tecnologia de mineração de dados (HASTIE, TIBSHIRANI & FRIEDMAN, 2001).

Em sua proposta de criação (AGRAWAL & SRIKANT, 1994), destaca-se a utilização da propriedade Apriori para redução do espaço de busca dos *itemsets* freqüentes.

A propriedade Apriori determina que "todo subconjunto não vazio de um *itemset* freqüente também é freqüente" (HAN & KAMBER, 2006) e é verificada facilmente, pois, se o *itemset B* é um subconjunto do *itemset* freqüente *A* ($S(A) \geq sup\text{-}min$), então, todas as transações que contêm *A* também contêm *B*, logo, $S(B) \geq S(A) \geq sup\text{-}min$, isto é, *B* também é freqüente.

A mineração de regras de associação é um processo realizado em dois passos:

1º. Encontrar todos os *itemsets* freqüentes.

2º. Gerar todas as regras de associação fortes a partir dos *itemsets* freqüentes.

Deve-se destacar que o desempenho geral da mineração de regras de associação é determinado pelo primeiro passo, sendo o segundo passo bem mais simples. Desta forma, o uso do algoritmo Apriori para encontrar os *itemsets* freqüentes, torna-se fundamental.

O exemplo a seguir ilustra o funcionamento do processo:

1º Passo (Figura 2.1):

Seja encontrar todos os *itemsets* freqüentes na base de dados abaixo, considerando um *sup-min* de 50%, isto é, contagem de *sup-min* = 2.

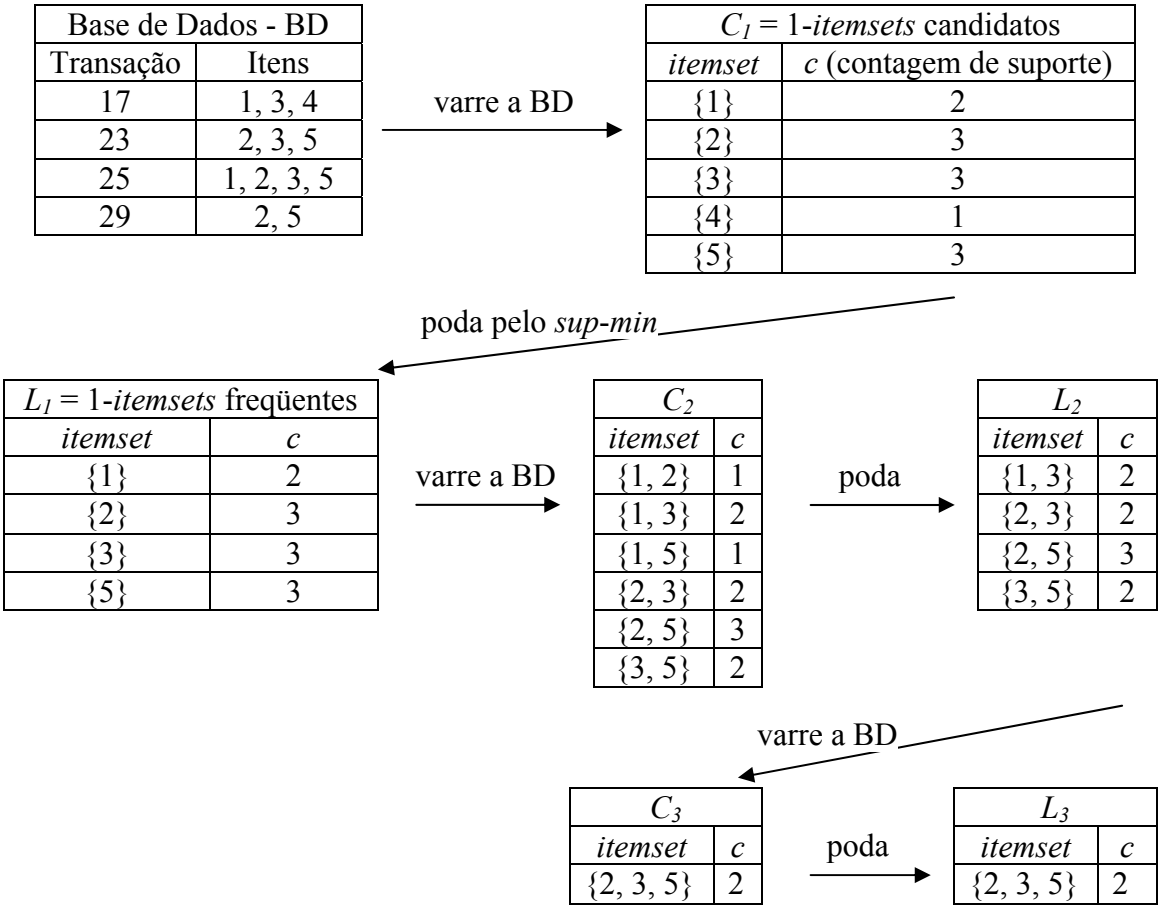


Figura 2.1. Execução do 1º Passo – encontrar todos os *itemsets* freqüentes

2º Passo:

Para gerar as regras de associação deve-se, inicialmente, distinguir todos os subconjuntos não-vazios *sc* de cada *itemset* freqüente *F*. No exemplo, o *itemset* freqüente *F* = {2, 3, 5} possui os seguintes subconjuntos não-vazios: {2, 3}, {2, 5}, {3, 5}, {2}, {3} e {5}.

Em seguida, produzir todas as regras com o formato "$sc \rightarrow (F - sc)$" e que satisfaça a *conf-min*.

No exemplo, produzindo as possíveis regras do *itemset* freqüente *F* = {2, 3, 5} e calculando as suas confianças, tem-se:

$C(2, 3 \rightarrow 5) = 2 / 2 * 100 = 100\%$

$C(2, 5 \rightarrow 3) = 2 / 3 * 100 = 67\%$

$C(3, 5 \rightarrow 2) = 2 / 2 * 100 = 100\%$

$C(2 \rightarrow 3, 5) = 2 / 3 * 100 = 67\%$

$C(3 \rightarrow 2, 5) = 2 / 3 * 100 = 67\%$

$C(5 \rightarrow 2, 3) = 2 / 3 * 100 = 67\%$

Considerando uma *conf-min* = 80%, são identificadas as seguintes regras fortes:

$2, 3 \rightarrow 5$ [50%, 100%]

$3, 5 \rightarrow 2$ [50%, 100%]

A execução do 1º passo representou o funcionamento do algoritmo Apriori, que tem por objetivo encontrar todos os *itemsets* freqüentes, usando geração de candidatos, dados um banco de dados de transações *T* e o limite de suporte mínimo (*sup-min*).

A Listagem 2.1, a seguir, apresenta o algoritmo Apriori (AGRAWAL & SRIKANT, 1994, HAN & KAMBER, 2006):

```
(1)  L1 = acha_1-itemsets_freqüentes(T);
(2)  para (k = 2; Lk-1 ≠ ø; k++){
(3)      Ck = gera_candidatos(Lk-1);
(4)      para cada transação t ∈ T {   // leitura de T para contagem
(5)          Ct = subconjunto(Ck, t); //recebe os subconjuntos candidatos de t
(6)          para cada candidato cd ∈ Ct
(7)              c(cd)++;
(8)      }
(9)      Lk = {cd ∈ Ct | c(cd) ≥ sup-min};
(10) }
(11) retorna L = ∪k Lk;
```

Listagem 2.1. Algoritmo Apriori

A partir do algoritmo Apriori, a grande maioria dos esforços tem se concentrado em duas frentes de pesquisa: (1ª) desenvolver variantes do algoritmo que ofereçam algum tipo de vantagem, especialmente, em relação ao desempenho e à escalabilidade e (2ª) encontrar medidas que possam identificar as regras realmente interessantes.

2.2.1 Evolução dos Algoritmos

Tem sido desenvolvido um grande número de algoritmos de mineração de regras de associação, que se diferem na eficiência computacional e nos requisitos de memória. Entretanto, os seus conjuntos de regras resultantes são os mesmos, pois se baseiam na definição de regras de associação, isto é, dado um conjunto de transações *T*, um suporte mínimo e uma confiança mínima, o conjunto de regras de associação existentes em *T* é único (LIU, 2008).

Os algoritmos criados posteriormente possuem, em geral, a mesma estrutura do Apriori, isto é, usam como base a propriedade Apriori (HAN & KAMBER, 2006).

Como exemplos desses algoritmos, tem-se (ALVES, 2007):

- Apriori-TID: faz somente 2 varreduras da base de dados, mas o consumo de memória principal é bastante elevado.

- Apriori-Híbrido: usa o Apriori nas etapas iniciais e, a partir de um determinado momento, passa a usar o Apriori-TID.

- DHP (Dynamic Hashing and Pruning): reduz o número de candidatos e progressivamente, reduz também a base de dados, melhorando, desta forma, o tempo de sua leitura.

- Partition: divide a base de dados em pequenas partes que podem ser processados na memória principal. Varre a base de dados somente duas vezes: na primeira, gera todos os *itemsets* potencialmente freqüentes e na segunda, calcula os suportes de todos os candidatos e identifica os *itemsets* freqüentes.

- SEAR (Sequential Efficient Association Rules): idêntico ao Apriori, mas experimentando o armazenamento dos candidatos numa árvore de prefixos.

- SPEAR (Sequential Partitioning Efficient Association Rules): semelhante ao SEAR, utilizando ainda a técnica do particionamento da base de dados. Faz somente duas varreduras na base de dados.

- DIC (Dynamic *itemset* Counting): divide a base de dados em partes de mesmo tamanho, que possam ser processadas na memória principal. Em leituras sucessivas destas partições, processa dinamicamente candidatos de tamanhos variados, objetivando assim reduzir o número de acesso aos dados.

- ALL-MFS (ALL Maximal Frequent Sets), MAXMINER, GENMAX e outros: usam a estratégia de buscar diretamente os *itemsets* freqüentes maximais e, em seguida, extrair seus subconjuntos.

- ECLAT (Equivalence Class Transformation), MAXECLAT, CLIQUE e MAXCLIQUE: utilizam uma estratégia de buscar tanto todos os *itemsets* freqüentes como os *itemsets* freqüentes maximais, dependendo do formato da base de dados, da técnica de decomposição adotada e do método de busca escolhido. Fazem uma única varredura na base de dados e são mais eficientes quando os *itemsets* freqüentes a serem descobertos são longos.

- FP-Growth (frequent-pattern growth): busca os *itemsets* freqüentes sem a geração de candidatos. Em duas varreduras na base de dados, calcula a freqüência de cada item, ordenando-os decrescentemente e monta uma estrutura de dados em árvore, chamada FP-tree. A partir daí, passa a minerar os *itemsets* freqüentes na árvore e não na base de dados.

- A-CLOSED e CHARM: utilizam uma estratégia de condensação dos dados e fazem somente uma varredura na base de dados.

Os algoritmos apresentados anteriormente constituem as principais abordagens seqüenciais para o problema. Existem ainda, outros que usam estratégias paralelas para mineração dos *itemsets* freqüentes, que não serão detalhados, por fugirem do objetivo do presente trabalho.

2.2.2 Interessabilidade das Regras

Um dos principais problemas com a análise do conjunto de regras de associação mineradas é que, geralmente, são produzidas muitas regras (dezenas de milhares, ou mais), tornando difícil para o usuário encontrar aquelas que são realmente úteis. Este é o chamado problema de interessabilidade (LIU, 2008).

As medidas de interessabilidade de uma regra de associação podem ser classificadas em duas categorias: subjetivas e objetivas. As medidas subjetivas se concentram em encontrar padrões interessantes baseados em um determinado conjunto de crenças do usuário (especialista do domínio), enquanto as objetivas medem a interessabilidade em termos de probabilidades (NICHOLAS & ZHAO, 2009).

As duas medidas de interessabilidade objetivas mais usuais são o suporte e a confiança, pois a maioria dos algoritmos de mineração emprega a estrutura suporte-confiança, conforme a formulação do algoritmo Apriori. Apesar de usar limites mínimos de suporte e de confiança para podar as regras desinteressantes, muitas regras fortes, que não interessam ao usuário, ainda podem ser produzidas (HAN & KAMBER, 2006).

Diversos estudos têm proposto a aplicação de medidas objetivas adicionais e/ou realizado testes comparativos entre elas.

Uma dessas medidas que merece destaque é a correlação, cuja utilização tem se tornado bastante comum, a ponto de se propor o seguinte formato para as regras de associação (HAN & KAMBER, 2006):

$$A \rightarrow B \text{ [suporte, confiança, correlação]}$$

Uma medida simples de correlação que tem sido muito utilizada, chamada *lift*, é assim calculada:

$$lift(A \rightarrow B) = \frac{P(A \mid B)}{P(B)} = \frac{P(A \cup B)}{P(A)P(B)} \tag{2.3}$$

onde:

- se *lift*($A \rightarrow B$) > 1, A e B são positivamente correlatos, isto é, a ocorrência de um implica na ocorrência do outro;

- se *lift*($A \rightarrow B$) = 1, A e B são independentes, isto é, não existe correlação entre eles, e

- se *lift*($A \rightarrow B$) < 1, a ocorrência de A é negativamente correlata com a ocorrência de B.

2.3 Tipos de Regras de Associação

Os três principais critérios para classificar as regras de associação são (HAN & KAMBER, 2006):

- Quanto aos valores manuseados, uma regra de associação pode ser:

  - Booleana: onde são associadas a presença ou a ausência de itens. Ex.: congelados, frutos do mar → vinho [2%, 71%].

  - Quantitativa: associa itens quantitativos ou atributos. Nestes casos, os valores quantitativos são discretizados. Ex.:
    idade(X, "40...50") e renda(X, "5000...7000") → compra(X, "vinho") [4%, 85%], onde X representa os consumidores.

- Quanto às dimensões dos dados envolvidos, considerando que cada predicado distinto é uma dimensão, uma regra pode ser:

  - Unidimensional, quando usa somente um predicado. Ex.:
    compra(X, congelados) → compra(X, vinho) [5%, 78%]
    ou, simplesmente
    congelados → vinho [5%, 78%]

  - Multidimensional, quando usa mais de um predicado. Ex.:
    idade(X, "40...50") e renda(X, "5000...7000") → compra(X, "vinho") [4%, 85%]

- Quanto aos níveis de abstração dos itens envolvidos, pode-se ter:

  - Regra de associação de um único nível: Ex.:
    congelados → vinho [5%, 78%], considerando que os itens congelados e vinho estão no mesmo nível de abstração, podendo representar conjuntos de subitens, isto é, congelados = {batata palito, brócolis etc.} e vinho = {vinho alemão, vinho argentino etc.}.

  - Regra de associação multinível: Ex.:
    congelados → vinho francês [1%, 43%], neste caso, o item congelados está num nível de abstração mais elevado do que vinho francês, que é um subitem do item vinho.

2.4 Alternativas de Mineração de Regras de Associação

A complexidade real da mineração de regras de associação reside na quantidade de dados. No ambiente da análise de cestas de compras, por exemplo, essa quantidade é sempre muito grande, tanto em número de itens, quanto em transações. O fato de ter que trabalhar com enormes volumes de dados, se reflete no aumento da complexidade da tarefa de mineração propriamente dita, bem como na descoberta das regras interessantes, configurando verdadeiros campos de pesquisas da área.

O próprio algoritmo Apriori, cujo processamento gera uma grande quantidade de *itemsets* candidatos e tem a necessidade de varrer repetidamente a base de dados, tem motivado o desenvolvimento de diversos algoritmos alternativos.

Neste sentido, alguns recursos adicionais podem ser utilizados pelo minerador, na busca da redução dessa complexidade, sendo, os mais importantes, descritos a seguir.

2.4.1 Uso de Restrições

O uso de restrições permite que seja especificado o conhecimento desejado como resultado da mineração das regras, de acordo com as intenções do minerador,

tornando o processo mais efetivo. A mineração baseada em restrições encoraja a mineração exploratória e a análise (HAN & KAMBER, 2006).

As principais formas de restrições são:

- Restrição de itens: usada quando interessam somente as regras formadas por um conjunto selecionado de itens. Neste caso, pode-se criar novas bases de dados, tanto de itens quanto de transações, contendo somente os itens de interesse.

- Restrição de transações: transações que não contém nenhum *k-itemset* freqüente não podem conter nenhum (*k*+1)-*itemset* freqüente. Essas transações são dispensadas de leituras futuras para *j-itemsets* onde $j > k$.

- Restrição de tempo: como as transações são normalmente colecionadas em ordem cronológica e o padrão de mineração de regras é feito por intervalos fixos de tempo (mês a mês, por exemplo), o uso dessa restrição é bastante comum. Neste caso, as transações realizadas em intervalos de tempo definidos pelo usuário constituem subconjuntos da base de dados de transações. A restrição de tempo também é usada em outras análises, como, por exemplo, planejar as vendas que antecedem um determinado feriado, com base nas vendas das quinzenas que antecederam este mesmo feriado, nos três últimos anos ou descobrir conhecimentos pela comparação da evolução dos negócios no primeiro trimestre dos cinco últimos anos.

- Restrição de *itemsets* ou de regras: especifica a forma das regras a serem mineradas como, por exemplo, o número máximo de itens do antecedente ou do conseqüente das regras.

- Restrição de nível de abstração: nas bases de dados onde os itens estão representados em mais de um nível de abstração, é interessante executar a mineração do mais alto para o mais baixo nível, fazendo uso deste tipo de restrição.

- Restrição de interessabilidade: os limites de suporte e confiança mínimos são as formas mais usuais dessa restrição. Nos casos onde são usadas outras medidas de interessabilidade, limitações semelhantes podem ser adotadas.

2.4.2 Uso de Múltiplos Suportes Mínimos (LIU, 2008)

O suporte mínimo (*sup-min*) é o elemento chave que torna viável a mineração de regras de associação, limitando o espaço de busca, através da poda do total de *itemsets* e, consequentemente, das regras geradas.

Por outro lado, com a sua utilização, assume-se, implicitamente, que todos os itens são da mesma natureza ou têm freqüências semelhantes na base de dados de transações, o que não acontece na prática. Num supermercado, por exemplo, a natureza dos bens duráveis ou caros (eletrodomésticos, móveis etc.) faz com que eles tenham freqüências muito menores que os demais (alimentos, higiene etc.). Além disso, outra característica da natureza desses bens duráveis é que, geralmente, possuem boas margens de lucro, tornando a análise de suas associações de interesse do mercado. Observa-se também que, em diversas aplicações envolvendo somente itens de mesma natureza, alguns deles aparecem muito mais frequentemente do que outros.

Com essa grande variação nas freqüências dos itens, a definição do limite de *sup-min* pode gerar o chamado problema do item raro:

- Se o *sup-min* for muito elevado, as regras que envolvem itens de baixa freqüência ou itens raros não serão encontradas.

- Se o *sup-min* for muito baixo, os itens freqüentes serão associados uns aos outros de todas as maneiras possíveis, gerando uma enorme quantidade de regras de conhecimento comum e sem utilidade, podendo causar, até, uma explosão combinatória na mineração, tornando-a inviável.

Assim, se por um lado é importante descobrir as regras contendo itens menos freqüentes, por outro é necessário fazê-lo sem permitir que itens altamente freqüentes produzam regras sem sentido, nem provocar a explosão combinatória.

A conclusão é que a utilização de um único *sup-min* para todo o conjunto de dados é insuficiente para conseguir capturar as naturezas inerentes ou itens de diferentes freqüências.

Uma solução seria dividir os dados em vários blocos menores, cada um contendo apenas os itens de freqüências semelhantes e, em seguida, minerar esses blocos separadamente, com diferentes valores de *sup-min*. Entretanto, esta solução apresenta a deficiência de não minerar *itemsets* ou regras com itens de blocos diferentes.

A melhor solução é permitir que o usuário especifique múltiplos suportes mínimos, isto é, que ele possa especificar um suporte mínimo diferente para cada item. Assim, diferentes *itemsets* devem satisfazer a diferentes suportes mínimos, dependendo de quais itens pertencem a cada *itemset*. O objetivo deste modelo é encontrar *itemsets* que contêm itens raros e evitar que os itens freqüentes causem a geração de muitos *itemsets* sem sentido.

É comum usar o modelo que usa múltiplos suportes mínimos junto com a restrição de determinados itens com alta freqüência, evitando, desta forma, a formação de *itemsets* que contenham tanto itens muito freqüentes, quanto itens raros. Esse é um recurso bastante útil na prática, porque em muitas aplicações, o interesse está apenas em encontrar determinados tipos de *itemsets* ou de regras.

CAPÍTULO 3
VISUALIZAÇÃO DE REGRAS DE ASSOCIAÇÃO

A necessidade de encontrar uma forma de saída gráfica de qualidade que possa representar com a maior clareza possível as regras de associação mineradas de uma base de dados motivou a realização do estudo a seguir.

Inicialmente, são descritas algumas abordagens gráficas para regras de associação. Na seqüência, são apresentados dois métodos de projeção multidimensional. Finalizando, são detalhados o funcionamento e alguns recursos do sistema de projeções adotado.

3.1 O Estado da Arte

A forma de apresentação de resultados usada pela maioria dos sistemas de mineração é feita através de uma listagem das regras descobertas. Como o número de regras é, geralmente, muito elevado, essa forma de apresentação dificulta o trabalho do minerador ou do especialista do domínio na identificação das regras que são realmente interessantes.

Esforços distintos têm sido dispensados na contribuição da melhoria de solução deste problema, entre eles verifica-se a evolução dos sistemas de visualização de regras de associação.

A seguir, são apresentadas as saídas gráficas de alguns desses sistemas.

- Técnica de Matriz ou de Grade (Figura 3.1): é um gráfico 3D para visualizar a força das associações entre pares de itens, posicionando-os em dois eixos e representando o suporte no terceiro eixo. Nesta técnica, as regras a serem visualizadas precisam ser selecionadas pelo usuário e o número de regras apresentadas na mesma imagem é muito restrito (HAO *et al*, 2001).

Figura 3.1. Técnica de Matriz – Visualizador de Associações do SGI MineSet.

- Técnica de Grafo (Figura 3.2): dispõe as associações em um grafo. Nesta técnica, quando o número de itens cresce muito, o processamento torna-se muito lento. Além disso, as imagens podem não representar claramente associações fortes entre os itens (HAO *et al*, 2001).

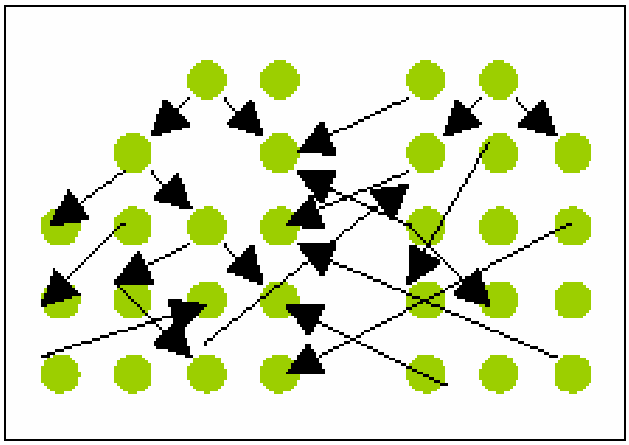


Figura 3.2. Técnica de Grafo.

- Gráfico de Regra (Figura 3.3): apresenta as regras de associação entre itens selecionados, onde cada item é representado por uma pequena esfera, cujo tamanho corresponde ao seu suporte (HAN & KAMBER, 2007).

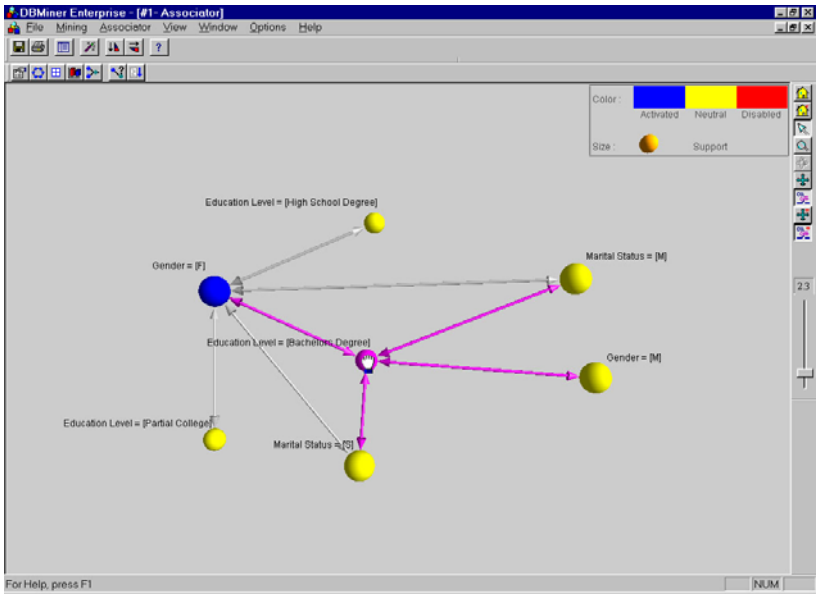
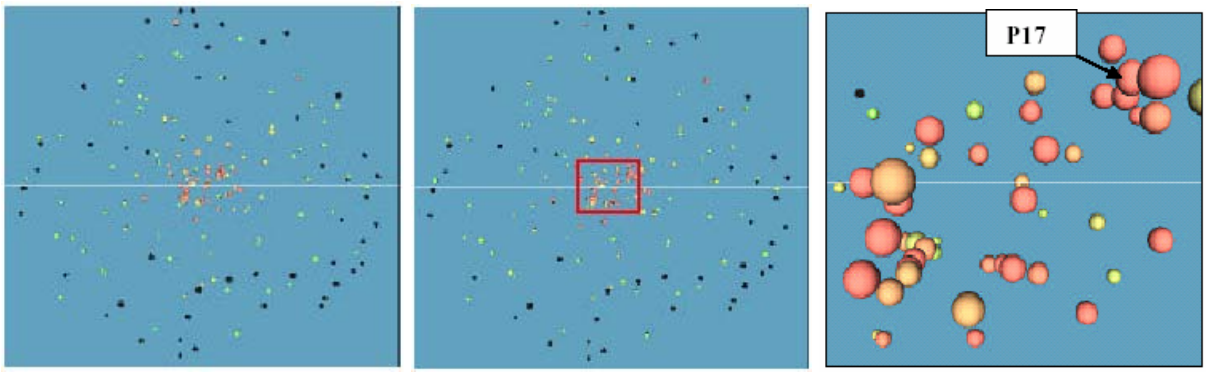


Figura 3.3. Gráfico de Regra - Visualizador de Regras do DBMiner Enterprise.

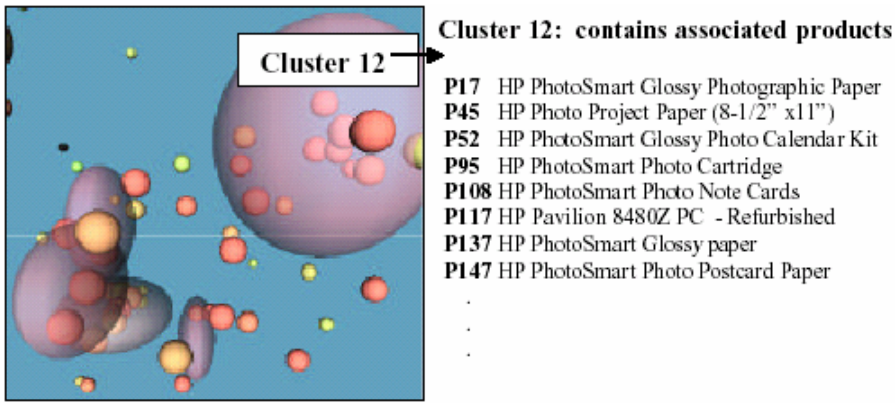
- Técnica de distância: diversos tipos de imagens podem ser obtidos com o uso desta técnica. Para ilustrar, são apresentadas duas abordagens, sendo uma de uso mais genérico e outra, mais específica. A primeira é o sistema DAV (*Directed Association Visualization*) (HAO *et al*, 2001), cujas principais características são:

  - os itens são posicionados em um espaço limitado por uma superfície esférica visual (3D),
  - a distância entre cada par de itens corresponde ao suporte da regra,
  - usa setas para indicar a direção das associações e a cor da extremidade da seta representa o nível de confiança e
  - agrupamentos contendo itens altamente relacionados podem ser visualizados em superfícies elipsoidais.

  O DAV fornece diversas imagens, como a distribuição de itens na esfera (Figura 3.4-a), seleção de uma área de mineração (Figura 3.4-b), ampliação da área selecionada (Figura 3.4-c), agrupamento auto-organizado (Figura 3.4-d), direção das associações (Figura 3.5) e perfil de clientes (Figura 3.6).



(a) (b) (c)



(d)

Figura 3.4. Sistema DAV: Distribuição de Itens na Esfera (a), Seleção de Área de Mineração (b), Ampliação da Área Selecionada (c), Agrupamento Auto-organizado (d).

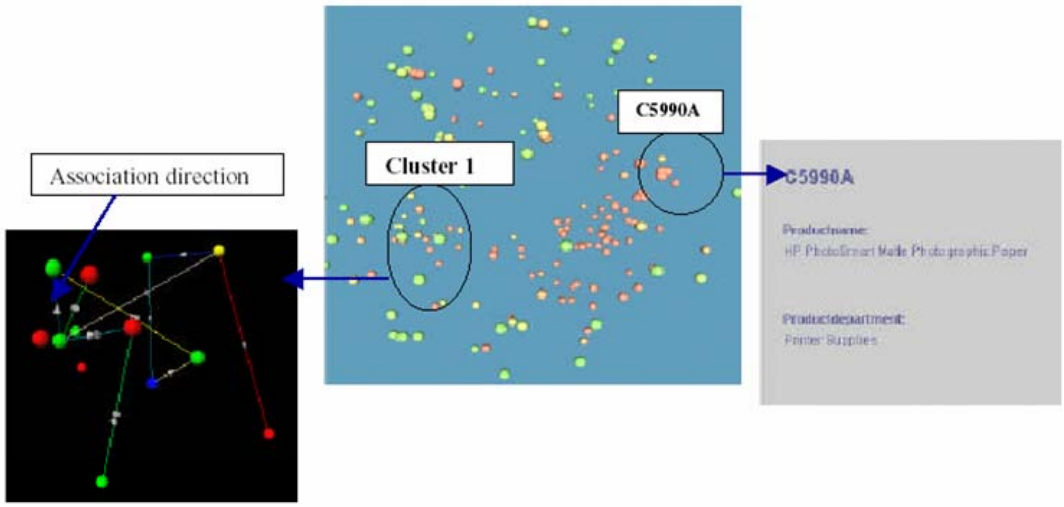


Figura 3.5. Sistema DAV: Direção das Associações.

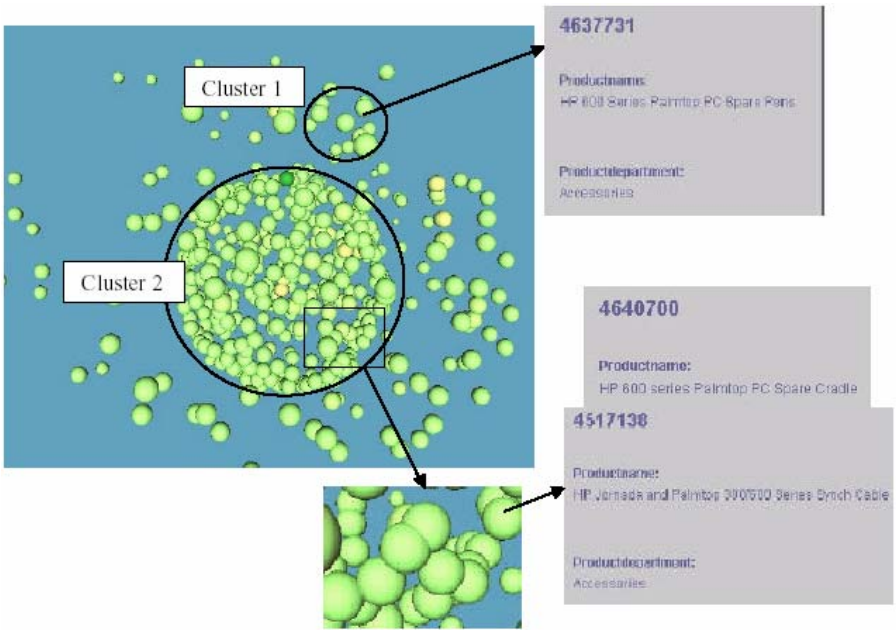


Figura 3.6. Sistema DAV: Perfil de Clientes.

A segunda abordagem é aplicada na distribuição espacial dos itens em prateleiras (CHEN & LIN, 2007). Realiza uma mineração de associação multinível para explorar as relações entre produtos, subcategorias e categorias de produtos, identificando quais itens devem ser disponibilizados, onde eles devem ser exibidos na estante e quanto espaço da estante deve ser alocado para cada um. O cálculo dos espaços alocados na estante se baseia na otimização do lucro dos *itemsets* freqüentes. Em relação à localização dos itens na estante e à distância entre eles, procura atender, se possível, a alguns chamados princípios básicos, que consideram: alocar itens do mesmo *itemset*, mesma subcategoria e mesma categoria, tão próximos quanto possível. Adicionalmente, adota a distância entre itens inversamente proporcional à suas freqüências (suportes). O resultado do processo é a visão bidimensional das distribuições das categorias (Figura 3.7), das subcategorias (Figura 3.8) e dos itens na estante (Figura 3.9).

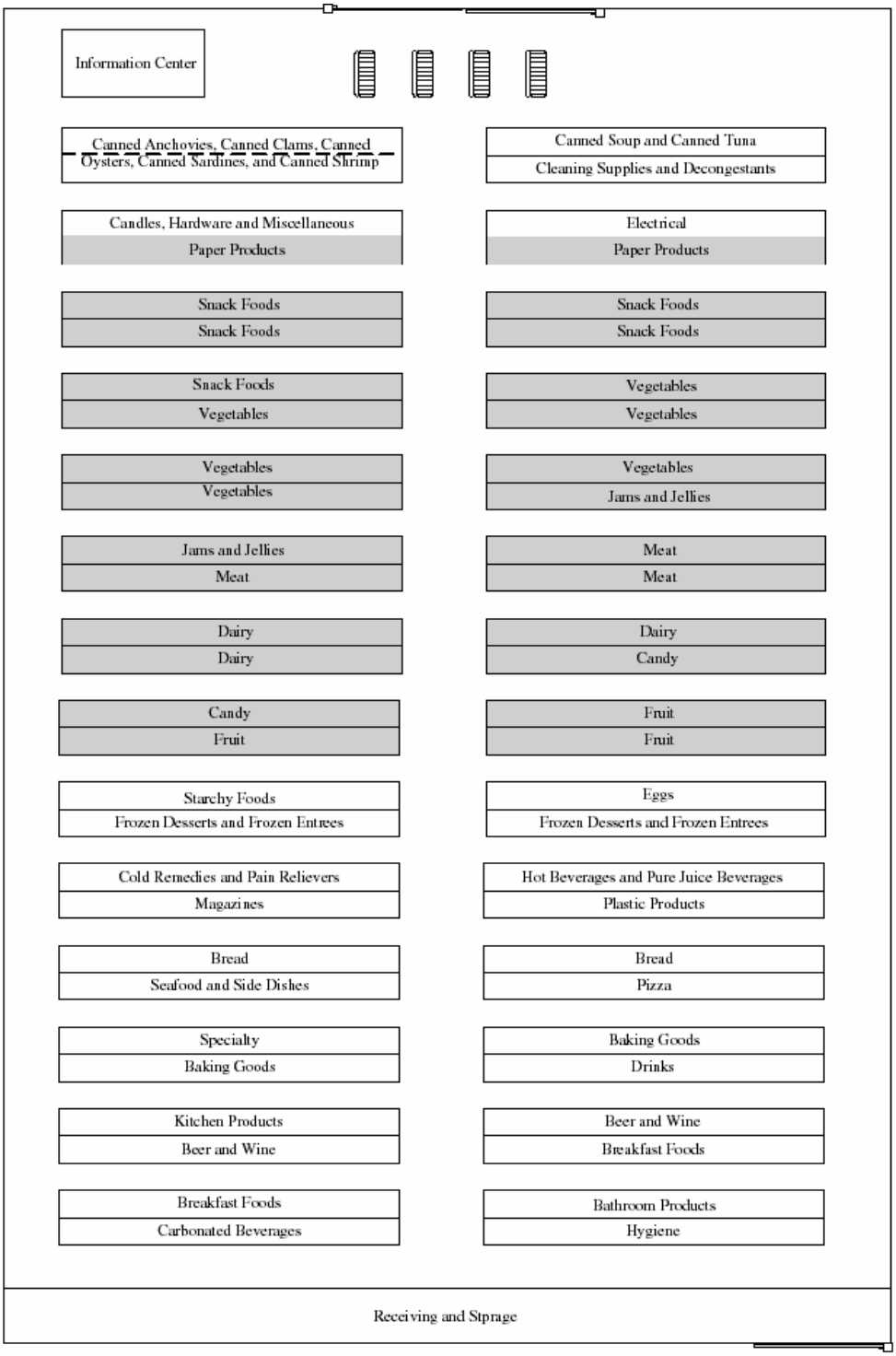


Figura 3.7. Técnica de Distancia: Distribuição das Categorias de Produtos.

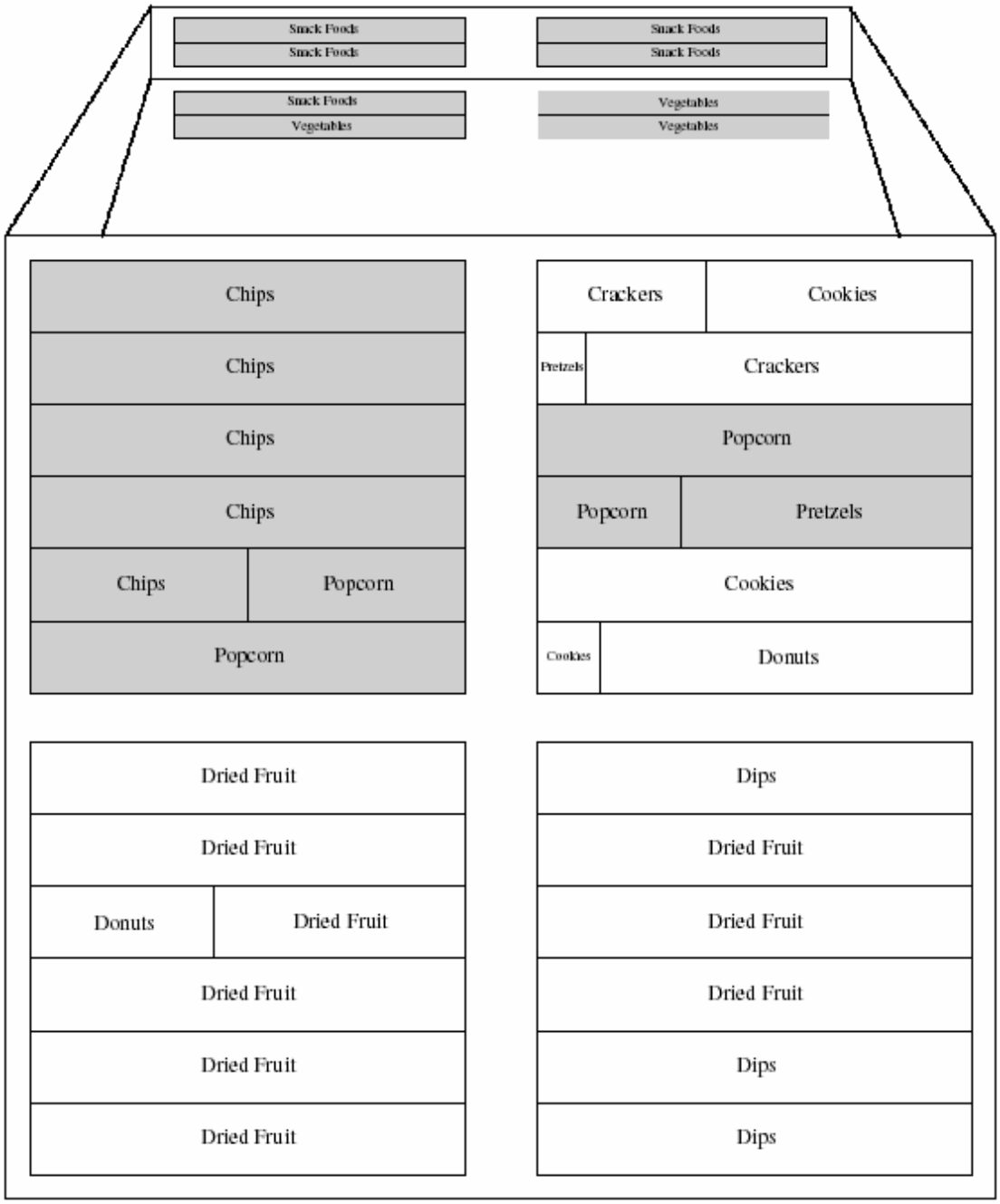


Figura 3.8. Técnica de Distancia: Distribuição Parcial de Subcategorias de Produtos.

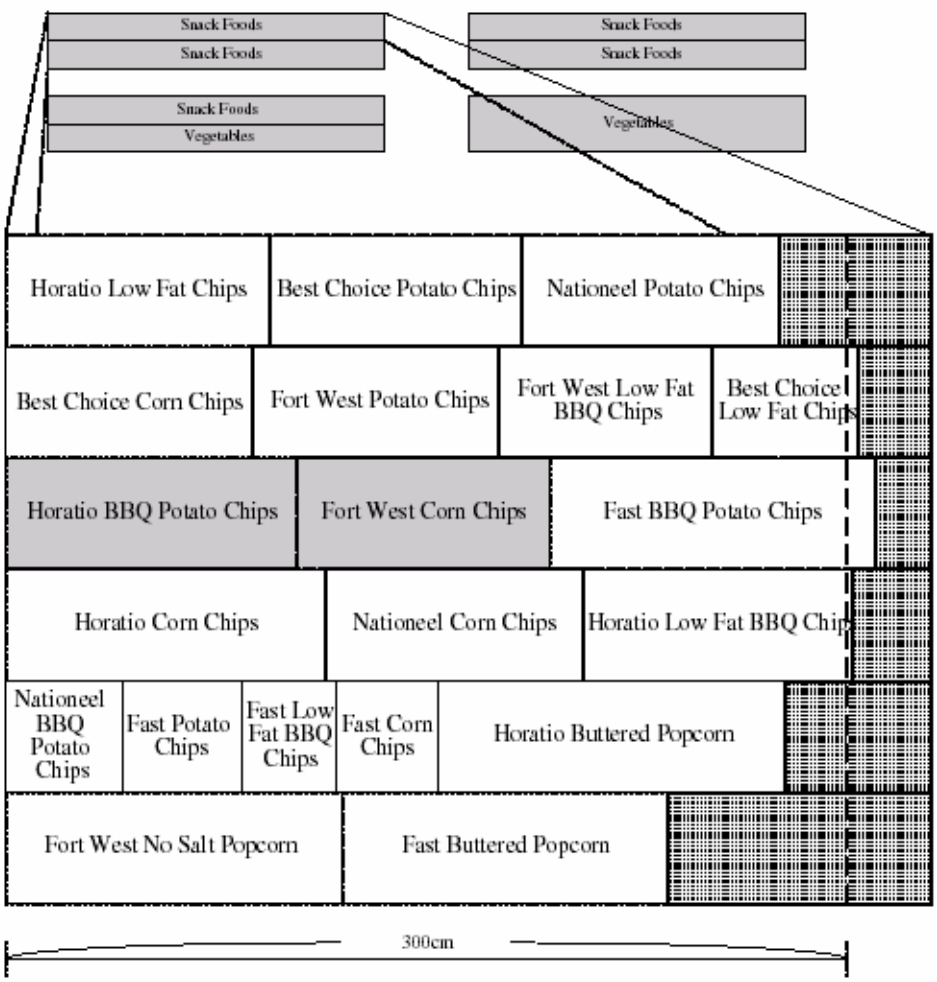


Figura 3.9. Técnica de Distancia: Distribuição Parcial de Itens na Estante.

3.2 Projeção Multidimensional

Uma forma comum de fazer uso de técnica de distância para obtenção de saídas gráficas é através de um método de projeção multidimensional. A seguir, são apresentados dois desses métodos.

O primeiro, chamado de escalonamento multidimensional ou MDS (*Multi-Dimensional Scaling*), é uma subárea da análise multivariada, que visa construir uma configuração de *n* pontos no espaço Euclidiano, usando informações sobre as distâncias entre *n* objetos (MARDIA, KENT & BIBBY, 1979).

Mais especificamente, dado o conjunto de distâncias entre todos os pares de *n* itens, o escalonamento multidimensional procura encontrar uma representação desses itens num espaço com o menor número possível de dimensões, tal que as proximidades

entre os itens sejam quase as mesmas das distâncias originais (JOHNSON & WICHERN, 1998).

Em linhas gerais, o método realiza uma redução de dimensionalidade, escolhendo um número q de dimensões, sendo q menor do que as $(n - 1)$ dimensões originais, e adotando um vetor de coordenadas $(x_1, x_2, \ldots, x_q)$ para cada item.

Calcula-se, então, as distâncias Euclidianas entre cada par de itens na dimensão q e, em seguida, as seqüências de distâncias originais e calculadas são comparadas. O que importa nesta comparação é a correspondência das distâncias dos respectivos pares de itens dentro de cada seqüência e não as suas magnitudes.

Se essa correspondência das seqüências for considerada aceitável, então o conjunto de coordenadas dos itens é adotado como solução para o espaço q-dimensional. Senão, o processo deve ser repetido, alterando levemente as coordenadas de cada item.

O objetivo do método é, portanto, escalonar (colocar em escala) a seqüência de distâncias calculadas, de acordo com a seqüência original. Isso nem sempre é possível, dependendo do valor escolhido para q.

Existem algumas formas de medida da qualidade da correspondência entre as seqüências (*Stress*, *SStress* (JOHNSON & WICHERN, 1998), *Stress1* (MANLY, 2008), *Desajuste* (HAIR *et al*, 2005)), sendo que, em todas elas, quanto menor a medida (mais próximo de zero), melhor o resultado.

É usual executar o método para valores crescentes de q, obtendo os respectivos valores da medida de qualidade escolhida e traçando, em seguida, um gráfico dessa medida em função de q. O número de dimensões ideal a ser utilizado, é o menor valor de q que torna a função menor que um limite pré-estabelecido para a medida de qualidade escolhida.

O gráfico da Figura 3.10 exemplifica a função *Stress* calculada a partir das distâncias entre 22 objetos, isto é, o espaço original possui 21 dimensões (JOHNSON & WICHERN, 1998).

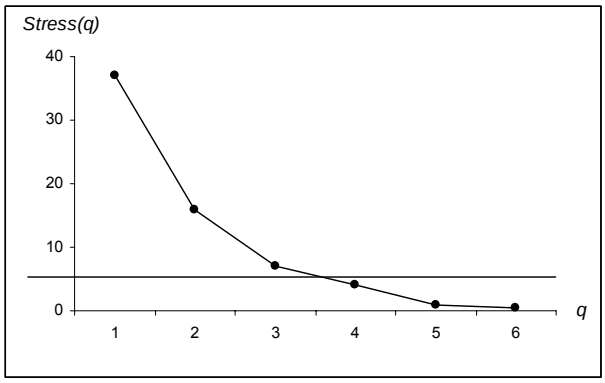


Figura 3.10. Gráfico da Função *Stress*(q).

Para obter uma projeção dos itens com pouca perda de informações, é recomendável que o valor do *Stress* fique abaixo do limite de 5%. Assim, o número mínimo a ser adotado no exemplo é de $q = 4$ dimensões.

Se o resultado final encontrado apontar para um valor de $q \leq 3$, a projeção pode ser direta. Caso contrário, é comum utilizar os eixos das componentes principais para traçar o mapa dos itens.

O segundo método, chamado *FastMap*, também propõe o mapeamento de n objetos como pontos no espaço Euclidiano, reduzindo a dimensionalidade e preservando, tanto quanto possível, as distâncias entre os objetos (FALOUTSOS & LIN, 1995).

Em resumo, seu algoritmo identifica, inicialmente, os dois objetos mais distantes entre si, definindo a reta que passa por eles no espaço original de $(n - 1)$ dimensões. Em seguida, define um hiperplano $(n - 2)$-dimensional, perpendicular à reta e projeta os objetos este hiperplano, obtendo a redução de uma dimensão. O processo é repetido até alcançar um hiperplano k-dimensional, sendo k especificado pelo usuário.

Em sua proposta original (FALOUTSOS & LIN, 1995) são apontadas algumas vantagens do *FastMap* em relação ao MDS, das quais destaca-se o desempenho computacional. Enquanto o *FastMap* é linear ($O(n)$), o MDS é quadrático ($O(n^2)$), o que pode tornar o MDS inviável para grandes massas de dados.

Essa diferença é comprovada em outro trabalho (CANO et al, 2002), onde numa aplicação com 1840 objetos o MDS gastou 894 segundos e o *FastMap* apenas 18,4. Por outro lado, observou-se que o *FastMap* necessita, as vezes, de várias execuções para

alcançar uma boa visualização e que os mapas do MDS apresentam, em geral, melhor qualidade.

3.3 Sistema de Projeção Gráfica

Na busca de um sistema de projeção gráfica que pudesse apoiar as aplicações desse trabalho, foi adotado um sistema chamado PEx (*Projection Explorer*), por atender plenamente todas as necessidades.

O PEx é apresentado como uma ferramenta de visualização multidimensional, disponível publicamente, destinada a exploração interativa dos dados, através de projeções de alta precisão (Paulovich *et al*, 2007).

Trata-se de um sistema bastante completo, oferecendo diversos recursos dos quais se destacam (Paulovich *et al*, 2007):

- Incorpora vários métodos de redução de dimensionalidade e de projeção (inclusive o MDS e o *FastMap*).
- Trabalha com conjuntos de dados estruturados (matrizes) ou não estruturados (textos).
- Manipula conjuntos de dados relativamente grandes (da ordem de poucos milhares a dez mil pontos de dados) com um custo computacional adequado, sem sacrificar a precisão.
- Em sua saída, projeta um conjunto de pontos no plano (R^2), representando um conjunto de dados no R^n, sendo cada ponto exibido como um círculo.
- Rótulos informativos podem ser mostrados acima dos pontos escolhidos.
- Informações adicionais representadas por um campo escalar associado a cada ponto, definem a sua cor e o seu tamanho.

- Várias ferramentas estão disponíveis (ícones no lado direito da janela principal) para explorar o gráfico de visualização, como, por exemplo, para destacar, selecionar, pesquisar, reposicionar e colorir os pontos de dados.

- Possui técnicas exploratórias complementares, como o traçado dos *K* vizinhos mais próximos (*KNN – K nearest neighborhoods*) no R^2 ou no R^n e a identificação de grupos pelo algoritmo *k-means*.

No caso da aplicação em questão, poucos foram os recursos do PEx utilizados, uma vez que o objetivo principal é somente a criação de projeções gráficas de itens no R^2 a partir da matriz de distância desses itens no R^n. Para isso, foi selecionado o *FastMap*, além de recursos simples adicionais para definir tamanho, cor e rótulo (código ou nome) dos itens no mapa.

Para exemplificar a execução do PEx, foi utilizada uma matriz de distância gerada pelo Sistema CalcD (que será detalhado no próximo capítulo) a partir da base de dados retirada do problema *The Charles Book Club*, que será chamada simplesmente de Livro.

Foram utilizados dois arquivos de entrada. O primeiro é de uso obrigatório, chamado Livro.dmat, contendo o número total de itens na linha 1, o código de cada item na linha 2, o nível de cada item (identificando tamanho e cor dos itens) na linha 3 e a matriz de distância (triangular inferior) a partir da linha 4. O conteúdo do arquivo Livro.dmat é mostrado na Listagem 3.1.

O segundo arquivo, chamado Livro.titles, é usado opcionalmente, quando se deseja rotular os itens por código ou por nome. O conteúdo deste arquivo é apresentado na Listagem 3.2.

A Figura 3.11 apresenta a tela do PEx com a projeção gerada a partir do arquivo Livro.dmat e identificando os itens por seus códigos.

Os mapas gerados pelo PEx podem ser exportados como imagens separadas, com é o caso do arquivo Livro.png da Figura 3.12, onde os itens foram identificados pelos nomes.

```
11
1;2;3;4;5;6;7;8;9;10;11
3;3;2;2;2;2;2;1;1;1;1
0.2604
0.1742;0.1725
0.1538;0.1542;0.1321
0.1310;0.1290;0.1133;0.1156
0.1290;0.1310;0.1259;0.1133;0.1109
0.1271;0.1274;0.1194;0.1194;0.1016;0.1016
0.0429;0.0424;0.0382;0.0429;0.0259;0.0382;0.0415
0.0292;0.0146;0.0231;0.0340;0.0335;0.0335;0.0340;0.0208
0.0165;0.0098;0.0137;0.0182;0.0184;0.0169;0.0049;0.0143;0.0137
0.0119;0.0033;0.0033;0.0148;0.0152;0.0033;0.0152;0.0101;0.0083;0.0148
```

Listagem 3.1. Conteúdo do Arquivo Livro.dmat.

```
Cód.;Nome
1;ChildBks1
2;CookBks2
3;DoItYBks3
4;GeogBks4
5;RefBks5
6;YouthBks6
7;ArtBks7
8;Florence8
9;ItalCook9
10;ItalArt10
11;ItalAtlas11
```

Listagem 3.2. Conteúdo do Arquivo Livro.titles.

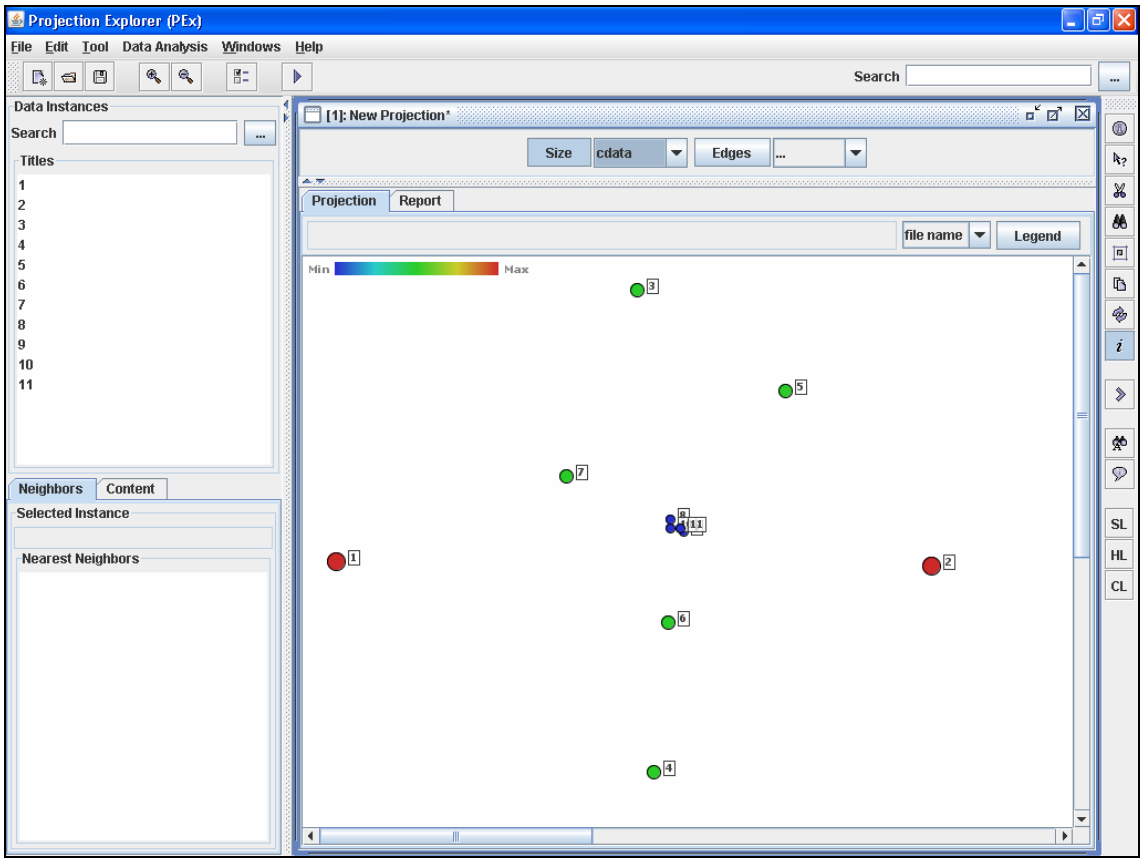


Figura 3.11. Tela do PEx com a Projeção Gerada a partir do Arquivo Livro.dmat.

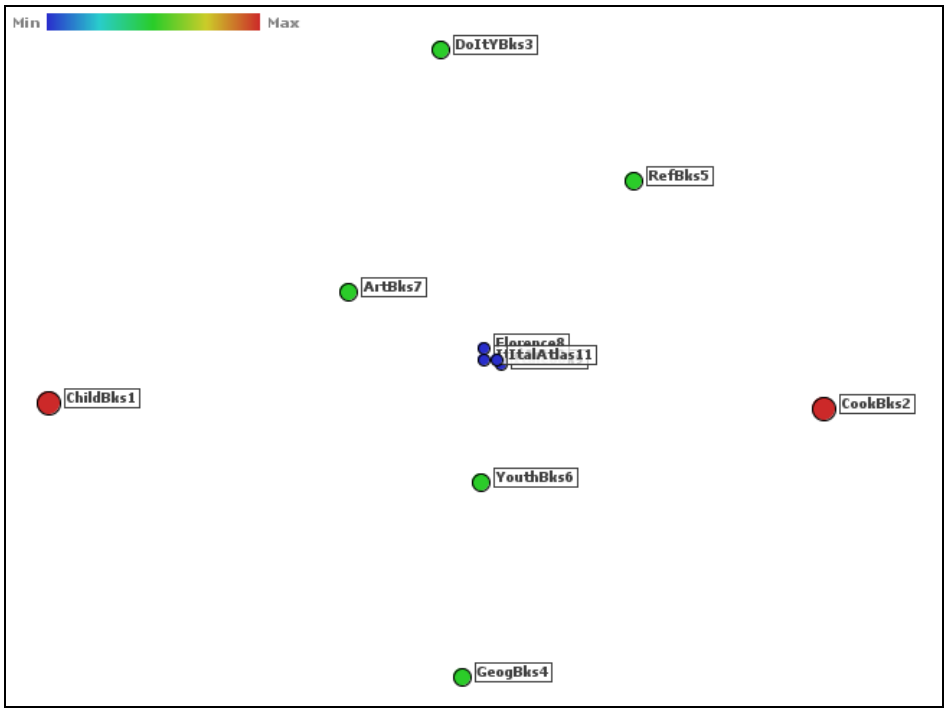


Figura 3.12. Arquivo Livro.png.

CAPÍTULO 4
TRANSFORMAÇÃO DE ASSOCIAÇÃO EM DISTÂNCIA

Este capítulo apresenta o sistema CalcD – um sistema inteligente que utiliza o conhecimento descoberto na mineração das regras de associação mais fortes de todos os 2-*itemsets* existentes em uma base de dados, para calcular as distâncias entre os itens envolvidos e criar estruturas de dados que possibilitam a visualização das associações entre esses itens no espaço bidimensional.

A complementação do CalcD por um sistema de projeção gráfica (PEx), cria um módulo de produção de imagens, configurando uma ferramenta que em muito pode auxiliar a pós-análise. Além disso, esse módulo pode ser incluído como parte integrante do processo iterativo de mineração, gerando imagens que orientam o minerador na busca das regras mais interessantes.

A criação de uma função que, a partir da regra de associação entre dois itens, calcula a distância entre eles, é o primeiro assunto tratado neste capítulo. Em seguida, é detalhado o sistema CalcD com seus algoritmos e estruturas de dados e são realizados experimentos computacionais, que permitem avaliar os mapeamentos das projeções bidimensionais obtidos.

4.1 A Função de Distância

A aplicação dos resultados obtidos na mineração de regras de associação tem levado, muitas vezes, a tomada de decisão baseada em distância. Um exemplo clássico é a regra *fralda* → *cerveja*, descoberta pela Wall-Mart, uma cadeia americana de lojas, cuja decisão foi colocar os dois produtos lado a lado, com conseqüente aumento de vendas (BISPO, 1998).

Nesta decisão intuitiva, que se mostrou acertada, foi adotada uma pequena distância para um caso particular.

Tomando este exemplo como base, a generalização do problema de transformação de associação em distância poderia ser assim enunciado:

"Dada uma regra de associação entre dois itens A e B, isto é, dada a regra $A \rightarrow B$, determinar a melhor distância a ser adotada entre A e B".

Neste caso, a melhor distância é aquela que melhor se enquadra entre as duas estratégias básicas de *marketing* a seguir (HAN & KAMBER, 2006):

(1) dois itens freqüentemente comprados juntos podem ser expostos em locais distantes, de forma a encorajar a venda de outros entre eles.

(2) dois itens freqüentemente comprados juntos podem ser expostos proximamente, de forma a encorajar a venda conjunta.

Numa análise simples dessas estratégias, chega-se à seguinte questão: quando é que "dois itens freqüentemente comprados juntos" devem ser expostos em locais distantes ou próximos?

Na busca da resposta, adotou-se os seguintes princípios:

- Será usada a estrutura suporte-confiança, que a partir do algoritmo Apriori, tornou-se comum na grande maioria dos sistemas de mineração de regras de associação.

- Será usada a regra mais forte para obtenção da distância entre os itens A e B, isto é, aquela que possui a maior confiança, já que o suporte é simétrico. Considerando que as freqüências de ocorrência de A e B na base de dados de n transações são, respectivamente, f_A e f_B e fazendo uso da expressão (2.2), tem-se:

$$C(A \rightarrow B) = \frac{c(A \cup B)}{c(A)} = \frac{c(A \cup B)}{f_A} \tag{4.1}$$

$$C(B \rightarrow A) = \frac{c(A \cup B)}{c(B)} = \frac{c(A \cup B)}{f_B} \tag{4.2}$$

Se $f_A \leq f_B$, então $C(A \rightarrow B) \geq C(B \rightarrow A)$

Concluindo: para obtenção da distância entre os itens A e B será usada a regra de associação $A \rightarrow B$ [S, C], tal que

$$f_A \leq f_B,\ S(A \rightarrow B) = \frac{c(A \cup B)}{n} \text{ e } C(A \rightarrow B) = \frac{c(A \cup B)}{f_A}. \quad (4.3)$$

Uma característica simples da estrutura suporte-confiança válida para qualquer regra é que, em valores absolutos, o suporte é sempre menor ou, no máximo, igual à confiança. Na regra apresentada anteriormente, isso é facilmente verificável, pois como $f_A \leq n$, então:

$$\frac{c(A \cup B)}{n} \leq \frac{c(A \cup B)}{f_A} \text{ ou } S(A \rightarrow B) \leq C(A \rightarrow B).$$

A partir desses princípios e dessa característica, a estratégia (1) pode ser interpretada a da seguinte forma: para que dois itens A e B sejam expostos em locais distantes e motivem a venda de outros entre eles, eles precisam vender bem individualmente e conjuntamente, isto é, f_A, f_B e $c(A \cup B)$ têm que ter valores relativamente altos. Assim, um valor alto para $c(A \cup B)$ faz com que o suporte e a confiança da regra de associação $A \rightarrow B$ também sejam altos. Além disso, como n é fixo para todas as regras, um valor alto para f_A indica que o valor do suporte está relativamente próximo do valor da confiança.

Já para interpretar a estratégia (2), pode-se voltar ao raciocínio da regra *fralda* → *cerveja*, onde a f_{fralda} é relativamente baixa (só compra fraldas quem realmente necessita delas) e $c(fralda \cup cerveja)$ tem que ser um valor próximo ou igual a f_{fralda} para justificar a proximidade de exposição dos produtos. Isso resulta em valor relativamente baixo para o suporte e alto para a confiança.

Com base nesta análise e considerando a regra de associação A → B, as duas estratégias básicas de *marketing* passam a representar a mesma informação, reescritas de formas opostas:

(1) quanto maior o suporte e quanto menor a diferença da confiança para o suporte, maior a distância entre A e B.

ou

(2) quanto menor o suporte e quanto maior a diferença da confiança para o suporte, menor a distância entre A e B.

Finalmente, generalizando a solução do problema colocado anteriormente, pode-se adotar o suporte da regra (S) como distância básica entre A e B e subtrair dessa distância básica um percentual dela própria, correspondente à diferença da confiança (C) menos o suporte, chegando a seguinte função de distância (D):

$$D = S - (C - S) * S \tag{4.4}$$

Diversas simulações variando os valores do suporte e da confiança da regra foram realizados, bem como testes com a base de dados Livro e todos os resultados foram considerados satisfatórios. Alguns exemplos são apresentados a seguir:

- As Figuras 4.1 e 4.2 mostram a influência da diferença ($C - S$) no cálculo de D.

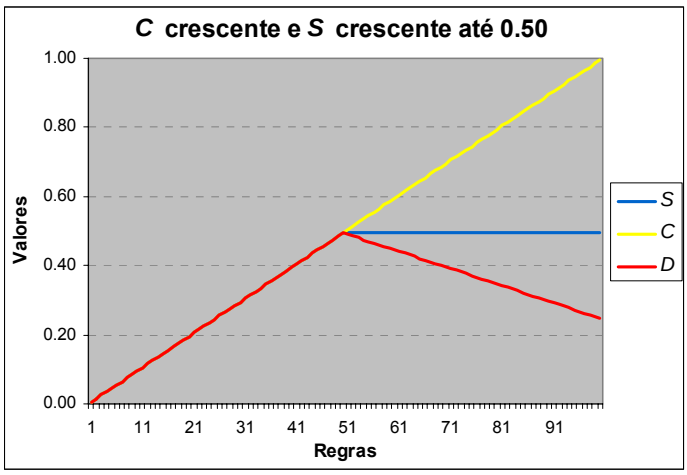


Figura 4.1. Simulação de D com C Crescente e S Crescente até 0.50.

- A Figura 4.3 demonstra como a função é capaz de atender as duas estratégias básicas de *marketing*. A regra 1, por exemplo, possui o maior suporte (S_{max} = 34%) e confiança relativamente alta (59%), isto é, são os dois itens que mais vendem conjuntamente (34% das transações) e em 59% das transações em que o A aparece, B também está presente. A função então, calculou a distância máxima para separá-los (D_{max}). Além da regra 1, as regras de 2 a 21 atendem claramente à estratégia (1), assumindo os maiores valores para as respectivas distâncias (entre 67% e 39% de D_{max}).

Em relação a estratégia (2), as regras 22 e 31 se destacam por possuírem valores razoáveis para S (35% e 20% de S_{max}), mas que, em razão dos valores de C serem muito altos (100% em ambas as regras), causando uma diferença ($C - S$) também muito elevada, receberam da função pequenas distâncias entre seus respectivos itens (6% e 2% de D_{max}), atendendo a estratégia (2). Essas regras caracterizam o chamado problema do item raro, isto é, são regras em que o antecedente A vende relativamente pouco, mas na grande maioria das transações onde ele aparece, o conseqüente B também está presente.

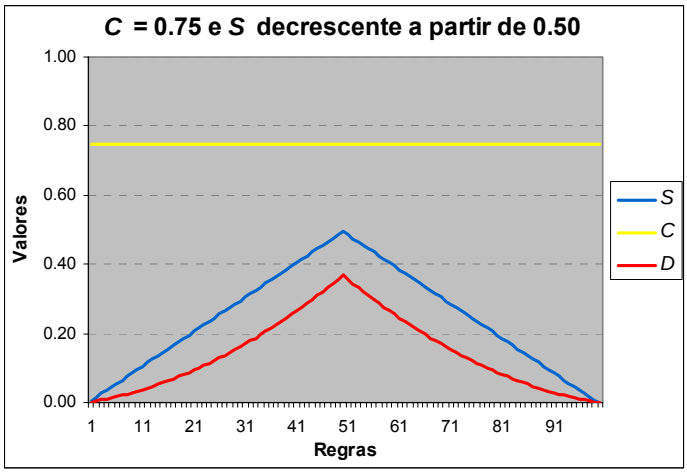


Figura 4.2. Simulação de D com $C = 0.75$ e S Decrescente a partir de 0.50.

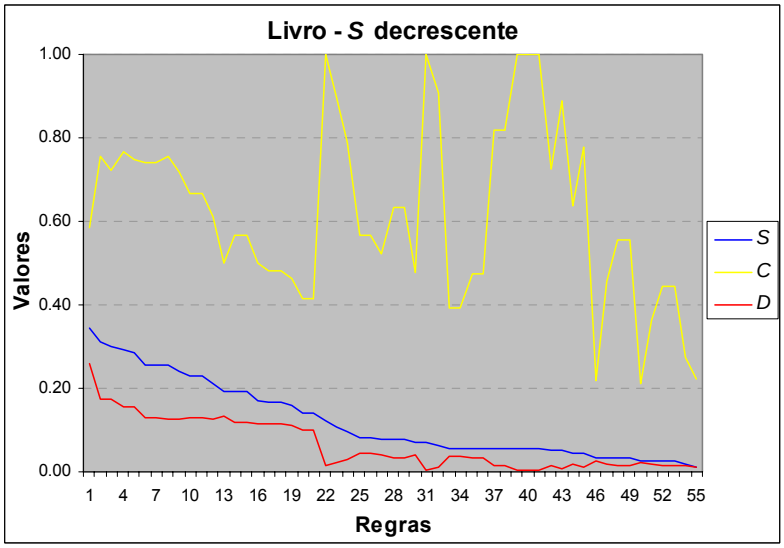


Figura 4.3. Teste de D com S Decrescente - Base de Dados Livro.

- Na Figura 4.4 as regras 44 e 52 são as que foram destacadas anteriormente como problemas do item raro, apresentando picos no gráfico do S, mas mantendo os valores de D bastante reduzidos. Outro fato que pode ser observado é a diferença de variação dos gráficos de S e de D. Na regra 11, por exemplo, S = 74% de Smax, enquanto D é apenas 49% de Dmax.

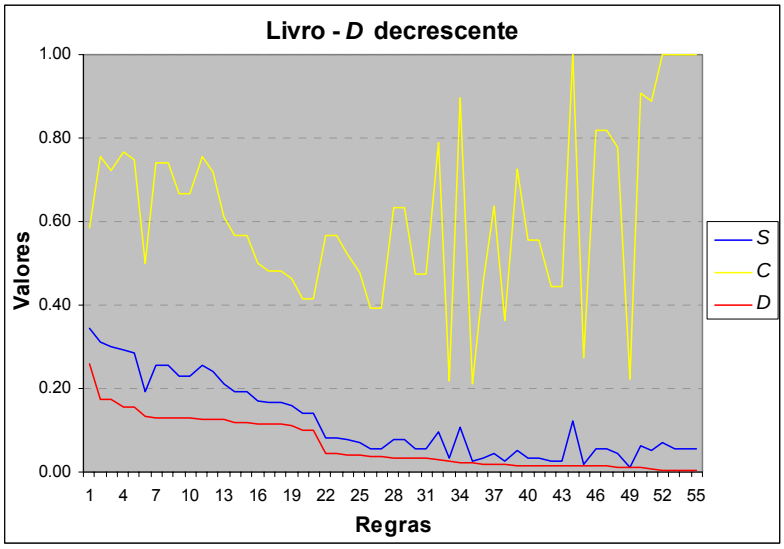


Figura 4.4. Teste de *D* com F Decrescente - Base de Dados Livro.

As tentativas de solução do problema de representar regras de associação através de distâncias têm apresentado algumas deficiências. Os sistemas que usam técnica de distância, descritos no capítulo anterior, por exemplo, resolvem somente parte do problema, isto é:

- O primeiro (DAV) usa o valor do suporte da regra como distância e, desta forma, atende somente a estratégia (1), isto é, itens com alta venda conjunta serão posicionados distantes um do outro, mas não resolve o problema do item raro.

- O segundo valoriza mais a maximização dos lucros, a partir da margem de lucro de cada item e do peso da posição ocupada na prateleira. Em termos de *marketing*, não atende nenhuma das duas estratégias, pois usa o inverso da freqüência dos itens como desempate para determinar a distância entre eles. Isto significa que os dois itens com maiores freqüências individuais tendem a ser os mais próximos e os dois com menores freqüências, os mais distantes.

4.2 O Sistema CalcD

O sistema CalcD calcula as distâncias entre todos os pares de itens presentes numa base de dados de transações, gerando uma matriz de distância. Em seguida, ele altera algumas das distâncias dessa matriz, de forma a organizar os itens em estruturas hierárquicas, para que, quando essa matriz for utilizada por um sistema de projeção multidimensional, as imagens obtidas tenham uma melhor qualidade de posicionamento dos itens para efeito de análise visual.

Outra saída da matriz de distância para projeção gráfica disponibilizada pelo sistema, permite a análise de agrupamentos através da construção da imagem formada por cada estrutura hierárquica. Além disso, o CalcD realiza um tratamento de distâncias nulas na matriz, gerando um relatório com todos os pares de itens com distância nula.

A seguir, são apresentados os programas GeraD, HierarqD e ClustD que constituem o sistema CalcD, realizando um experimento computacional com a base de dados Livro, ilustrado com imagens geradas pelo sistema PEx .

4.2.1 Programa GeraD

O programa GeraD minera as regras de associação mais fortes de uma base de dados de transações, calcula as distâncias entre todos os pares de itens e, finalmente, gera e grava a matriz de distância original.

Usa dois arquivos de entrada, o primeiro contendo a freqüência de todos os itens em ordem decrescente e o segundo é o arquivo de transações. Estes arquivos são gerados pelo sistema SelMultN (descrito no próximo capítulo), que varre o arquivo original de transações, calcula e coloca em ordem decrescente as freqüências de todos itens. Essa ordenação identifica um código numérico inteiro, a partir de 1 (um), para cada item, isto é, o código do item identifica a sua posição dentro da seqüência decrescente de freqüências. Em seguida, o sistema SelMultN gera um novo arquivo de transações, substituindo os itens do arquivo original pelos seus respectivos códigos.

A utilização desses códigos tem dois objetivos em relação a melhoria de desempenho:

- Na leitura do arquivo de freqüências, não é necessário ler a identificação dos itens, pois a primeira freqüência é do item de código 1, a segunda é do 2 e assim por diante.

- O código de um item é usado internamente como índice da estrutura de dados a ser manipulada. Por exemplo, quando o GeraD lê o código de um item em uma transação, ele já está indicando a linha (ou coluna) de uma matriz que deverá ser alterada.

Uma dificuldade encontrada ao longo deste trabalho foi a grande quantidade de arquivos distintos a serem manipulados, tornando necessária a adoção do seguinte padrão de nomeação dos arquivos:

<conteúdo><base de dados><versão> - <parâmetros> . <tipo>

onde:

- <conteúdo> é a abreviatura sugestiva do conteúdo do arquivo;

- <base de dados> é a abreviatura do nome da base de dados utilizada;

- <versão> é um campo opcional que identifica a versão do problema em questão, podendo ser, por exemplo, um período de tempo (mês, ano etc.) ou simplesmente uma identificação numérica;

- <parâmetros> é um campo opcional que identifica os parâmetros usados na execução em questão, e

- <tipo> é o tipo do arquivo.

Assim, fazendo uso da base de dados Livro, os arquivos de entrada do programa GeraD foram chamados de FreqItensL.txt e TransCodL.txt e seus conteúdo são mostrados nas Tabelas 4.1 e 4.2.

Antes de iniciar a execução do programa GeraD, o usuário (minerador ou especialista do domínio) deve definir os níveis de freqüência dos itens, analisando uma planilha, também gerada pelo sistema SelMultN, contendo os códigos, nomes e os

valores absoluto e percentual das freqüências de cada item, além da coluna Nível em branco (Tabela 4.3).

92
92
65
60
54
54
53
23
19
11
9

Tabela 4.1. Arquivo FreqItensL.txt.

1 2 5
1 4 5
2 3 4 9
2
1
3
2
3
1 5
1 2 3 4 5
2 8
1 2 3 4 6
7
1 2 3 4 5 7
2
...

Tabela 4.2. Arquivo TransCodL.txt.

Cód.	Nome	Freqüência		Nível
		Abs.	Per.	
1	ChildBks	92	58.60	
2	CookBks	92	58.60	
3	DoItYBks	65	41.40	
4	GeogBks	60	38.22	
5	YouthBks	54	34.40	
6	RefBks	54	34.40	
7	ArtBks	53	33.76	
8	Florence	23	14.65	
9	ItalCook	19	12.10	
10	ItalArt	11	7.01	
11	ItalAtlas	9	5.73	

Tabela 4.3. Planilha para Definição dos Níveis de Freqüência.

O usuário deve definir pelo menos dois níveis, numerando-os em ordem decrescente até 1 (um). A coluna Nível serve para auxiliar nesta tarefa. Uma alternativa que se mostrou interessante para escolha dos níveis é mudar o seu valor nas quebras

mais bruscas da seqüência de freqüências. A Tabela 4.4 mostra a escolha de três níveis baseada nesta alternativa.

Cód.	Nome	Freqüência		Nível
		Abs.	Per.	
1	ChildBks	92	58.60	3
2	CookBks	92	58.60	3
3	DoItYBks	65	41.40	2
4	GeogBks	60	38.22	2
5	YouthBks	54	34.40	2
6	RefBks	54	34.40	2
7	ArtBks	53	33.76	2
8	Florence	23	14.65	1
9	ItalCook	19	12.10	1
10	ItalArt	11	7.01	1
11	ItalAtlas	9	5.73	1

Tabela 4.4. Planilha com os Níveis de Freqüência Definidos.

Essa definição dos níveis de freqüência tem dois objetivos importantes:

- O sistema CalcD utiliza essa definição para organizar as estruturas hierárquicas, onde cada item do nível de freqüência mais alto dá origem a uma estrutura.

- O nível de freqüência de um item identifica o tamanho e a cor do círculo que vai representá-lo no mapa de projeção do sistema PEx.

Com a planilha preenchida em mãos e fornecidos os dois arquivos de entrada, inicia-se a execução do programa GeraD, solicitando as informações da Tabela 4.5 via vídeo/teclado:

Solicitação do Programa	**Informação do Usuário**
Número de Níveis de Freqüência:	3
Número de Itens por Nível de Freqüência:	
Nível 3:	2
Nível 2:	5
Nível 1:	4

Tabela 4.5. Informações Solicitadas pelo Programa GeraD.

O processamento é, então, efetivamente iniciado, realizando a seguinte seqüência de tarefas:

- Leitura do arquivo FreqItensL.txt e criação do vetor *FREQ* com as freqüências dos itens, cujos índices são os seus respectivos códigos.

- Leitura do arquivo TransCodL.txt e criação da matriz *c* com a contagem de suporte das regras de associação mais fortes de todos os 2-itemsets, isto é, as regras envolvendo dois itens A e B, com o sentido $A \rightarrow B$, tal que $f_A \leq f_B$ (conforme 4.3).

  A Tabela 4.6 representa a matriz *c* resultante.

→	1	2	3	4	5	6	7	8	9	10	11
1											
2	54										
3	49	47									
4	45	46	30								
5	40	36	26	27							
6	36	40	33	26	25						
7	38	40	30	30	22	22					
8	13	12	9	13	5	9	11				
9	15	19	17	12	9	9	12	4			
10	8	10	9	7	5	4	11	3	9		
11	7	9	9	4	5	9	5	2	8	4	

Tabela 4.6. Matriz de Contagem de Suporte *c* da Base de Dados Livro.

A primeira linha e a primeira coluna da Tabela 4.6 estão representando os índices de coluna e linha, respectivamente, que, por sua vez, são os códigos dos itens. Assim, o valor do elemento situado na linha 6, coluna 4, por exemplo, é o valor de $c(6 \rightarrow 4) = 26$.

Embora a matriz seja simétrica, só interessam os valores abaixo da diagonal principal, para que seja atendida a condição $f_A \leq f_B$, tratando-se, portanto, de uma matriz triangular inferior, sem a diagonal principal. Isso possibilita armazená-la em uma representação vetorial com mais de 50% de economia de memória. Neste caso, a representação vetorial pode armazenar todos os elementos situados abaixo da diagonal principal, percorrendo-os linha por linha, de cima para baixo e da esquerda para a direita. O vetor resultante é apresentado na Tabela 4.7.

Neste exemplo com somente 11 itens, o armazenamento do vetor no lugar da matriz causa uma economia de (11 x 11 = 121 – 55 =) 66 elementos (54.55%).

[*I*] =	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
c =	54	49	47	45	46	30	40	36	26	27	36	40	33	26	25	38	40	30	30	22

21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
22	13	12	9	13	5	9	11	15	19	17	12	9	9	12	4	8	10	9	7

41	42	43	44	45	46	47	48	49	50	51	52	53	54	55
5	4	11	3	9	7	9	9	4	5	9	5	2	8	4

Tabela 4.7. Representação Vetorial da Matriz de Contagem de Suporte *c*.

Para generalizar o problema de transformação da matriz *c* em sua representação vetorial, considerando um total de *m* itens, é necessário:

1°) Dimensionar o vetor com *p* elementos (quantidade de elementos situados abaixo da diagonal principal), sendo $p = \sum_{i=1}^{m-1} i = m\,(m-1)$ div 2.

2°) Relacionar o índice *I* do vetor com os índices *L* (de linha) e *C* (de coluna) da matriz para efeito de consulta e atribuição, isto é, dados os valores válidos de *L* e *C*, o elemento *c*[*L*, *C*] está na posição *I* do vetor. Como *I* é crescente ao percorrer *c* linha por linha, de cima para baixo e da esquerda para a direita, pode-se determinar:

$$I = \sum_{i=1}^{L-2} i + C = (L-2)(L-1)\underline{div}2 + C$$

A utilização da representação vetorial no lugar da matriz tem duas conseqüências. A primeira (positiva) é a economia de mais de 50% do espaço de memória e a segunda (negativa) é a perda de desempenho pela necessidade de usar a transformação de índices a cada acesso no vetor. Mas, como os problemas práticos de mineração de regras de associação envolvem muitos itens, a questão de economia de memória se torna primordial, fazendo com que a solução do uso da representação vetorial seja, com certeza, vantajosa.

Todas as matrizes usadas pelo sistema CalcD possuem o mesmo aspecto da matriz *c*, isto é, são triangulares inferiores, e todas elas são tratadas internamente (na memória principal) por suas representações vetoriais. Por

razão puramente didática, elas serão apresentadas neste texto em seus formatos originais.

- A próxima tarefa executada pelo GeraD é a criação da matriz *S* contendo os suportes das regras de associação mais fortes de todos os 2-itemsets. Para obter *S*, basta dividi *c* pelo total de transações *n*. A matriz *S* criada para a base de dados Livro, mostrada na Tabela 4.8, é o resultado da divisão de *c* (Tabela 4.6) por *n* = 157 transações.

→	1	2	3	4	5	6	7	8	9	10	11
1											
2	0.3439										
3	0.3121	0.2994									
4	0.2866	0.2930	0.1911								
5	0.2548	0.2293	0.1656	0.1720							
6	0.2293	0.2548	0.2102	0.1656	0.1592						
7	0.2420	0.2548	0.1911	0.1911	0.1401	0.1401					
8	0.0828	0.0764	0.0573	0.0828	0.0318	0.0573	0.0701				
9	0.0955	0.1210	0.1083	0.0764	0.0573	0.0573	0.0764	0.0255			
10	0.0510	0.0637	0.0573	0.0446	0.0318	0.0255	0.0701	0.0191	0.0573		
11	0.0446	0.0573	0.0573	0.0255	0.0318	0.0573	0.0318	0.0127	0.0510	0.0255	

Tabela 4.8. Matriz de Suporte *S* da Base de Dados Livro.

- Calcula-se, então, a matriz *C* com as confianças das regras, dividindo cada linha *L* da matriz *c* pela freqüência do item *L* (f_L) do vetor *FREQ*. A Tabela 4.9 apresenta a matriz *C* do exemplo.

→	1	2	3	4	5	6	7	8	9	10	11
1											
2	0.5870										
3	0.7538	0.7231									
4	0.7500	0.7667	0.5000								
5	0.7407	0.6667	0.4815	0.5000							
6	0.6667	0.7407	0.6111	0.4815	0.4630						
7	0.7170	0.7547	0.5660	0.5660	0.4151	0.4151					
8	0.5652	0.5217	0.3913	0.5652	0.2174	0.3913	0.4783				
9	0.7895	1.0000	0.8947	0.6316	0.4737	0.4737	0.6316	0.2105			
10	0.7273	0.9091	0.8182	0.6364	0.4545	0.3636	1.0000	0.2727	0.8182		
11	0.7778	1.0000	1.0000	0.4444	0.5556	1.0000	0.5556	0.2222	0.8889	0.4444	

Tabela 4.9. Matriz de Confiança *C* da Base de Dados Livro.

- Na seqüência, é calculada a matriz *D* contendo as distâncias entre cada par de itens, fazendo uso da função (4.4). A matriz *D* do exemplo é mostrada na Tabela 4.10.

- Finalizando, o programa grava o arquivo com a matriz de distância *D* original já adaptado para as especificações do arquivo principal de entrada

do sistema PEx, possibilitando a sua visualização gráfica. O arquivo DOrigL.txt, resultante do processamento do programa GeraD para a base de dados Livro, após substituição de seu tipo de txt para dmat, foi apresentado na Listagem 3.1.

→	1	2	3	4	5	6	7	8	9	10	11
1											
2	0.2604										
3	0.1742	0.1725									
4	0.1538	0.1542	0.1321								
5	0.1310	0.1290	0.1133	0.1156							
6	0.1290	0.1310	0.1259	0.1133	0.1109						
7	0.1271	0.1274	0.1194	0.1194	0.1016	0.1016					
8	0.0429	0.0424	0.0382	0.0429	0.0259	0.0382	0.0415				
9	0.0292	0.0146	0.0231	0.0340	0.0335	0.0335	0.0340	0.0208			
10	0.0165	0.0098	0.0137	0.0182	0.0184	0.0169	0.0049	0.0143	0.0137		
11	0.0119	0.0033	0.0033	0.0148	0.0152	0.0033	0.0152	0.0101	0.0083	0.0148	

Tabela 4.10. Matriz de Distância *D* da Base de Dados Livro.

Considerando que a matriz de distância original *D* do arquivo DOrigL.dmat contem as distâncias entre todos os pares de itens no R^n (no caso R^{10}) e que o método *FastMap*, usado através do sistema PEx, realizou a projeção dessas distâncias no R^2, mantendo tanto quanto possível a relação entre elas, a análise visual dos mapas gerados (Figuras 3.11 e 3.12) permite destacar:

- O posicionamento dos itens 1 e 2 do nível de freqüência 3 é ideal, pois são os que mais vendem conjuntamente e devem ocupar as posições mais distantes.

- O posicionamento dos itens 3, 4, 5, 6 e 7 de níveis 2 é satisfatório, pois vendem razoavelmente bem em relação aos itens de nível 3 e entre eles próprios, devendo ocupar distâncias intermediárias. Em relação aos itens de nível 3, pode-se observar que o item 5 está mais próximo do item 2 do que do 1, isto é, ele vende melhor com o 1 do que com o 2 e o item 7 tem comportamento contrário. Entretanto, não é possível distinguir visualmente com qual dos dois itens de nível 3, os itens 3, 4 e 6 vendem melhor (estão mais distantes).

- O posicionamento dos itens 8, 9, 10 e 11 de níveis 1 é ruim. Como esses itens têm freqüências relativas muito baixas, eles estão próximos de todos os

outros itens no R^n e suas projeções no R^2 fica concentrada, não representando quais distâncias são importantes e quais não são.

Esses problemas de posicionamento mostraram que a matriz de distância original é um ponto de partida que precisa ser melhorada para a obtenção de uma projeção realmente representativa das associações importantes.

Neste aspecto, a associação mais importante de um item de baixa freqüência será com o item de maior freqüência possível, que esteja mais próximo. Esta é, na verdade, a melhor solução do problema de item raro.

A generalização dessa solução resultou em um método que organiza os itens em ordem decrescente de seus níveis de freqüência, criando uma estrutura hierárquica distinta para cada item do nível mais alto e mantendo as distâncias originais entre eles. Os demais itens são alocados nessas estruturas, valorizando e mantendo a menor distância e ajustando as suas outras distâncias, se necessário.

4.2.2 Programa HierarqD

O programa HierarqD usa como entrada o arquivo da matriz de distância *D* original criado pelo programa GeraD e executa a seqüência das três seguinte tarefas: trata as distâncias nulas, hierarquiza *D* e otimiza *D*. Para cada tarefa é gravado um arquivo de saída.

Como o programa GeraD minera as regras de associação de todos os 2-*itensets* da base de dados de transações, não havendo poda de regras por suporte mínimo ou confiança mínima, pode ocorrer situações de distâncias nulas na matriz *D*. Isso ocorre quando dois itens não se associam, pois, neste caso, $c = 0$ e, consequentemente, $S = 0$, $C = 0$ e $D = 0$.

Na realidade, sob o ponto de vista das estratégias de *marketing*, a condição de não associação significa que não existe distância entre os itens. Por outro lado, é necessário adotar um valor de distância para efeito da projeção de todo o conjunto de itens analisados no R^2.

Como não há uma distância ideal para este caso, diversos experimentos foram realizados e a melhor solução encontrada foi a adoção da distância máxima existente entre todos os itens pertencentes ao maior dos níveis de freqüência envolvidos na regra. Isso fez com que, na maioria dos experimentos, o item de menor freqüência e seus descendentes hierárquicos ficassem mais isolados dos demais pertencentes à mesma estrutura.

Além disso, a rotina emite um relatório dos itens que não se associam, para acompanhar o mapa de projeção correspondente, com o objetivo de elucidar a análise do mapa em relação a esses itens.

A hierarquização da matriz *D* é segunda tarefa executada pelo programa e busca valorizar as distâncias mais importantes para efeito da projeção dos itens do R^n no R^2, atendendo as estratégias de *marketing* da melhor forma, isto é, identificando para cada item projetado, quais outros devem estar distantes e quais devem estar próximos dele no R^2.

A seguir são apresentados alguns conceitos e uma idéia geral do método de hierarquização:

De um conjunto no R^n, sejam *A* e *B* dois itens que possuem uma distância relativamente grande ($D^n(A, B)$). Um sistema de projeção com uma resolução razoável, manterá *A* e *B* a uma distância relativamente grande também no R^2 ($D^2(A, B)$).

Convencionou-se chamar de "área de influência" o espaço ao redor de um item no R^n definido, no mínimo, pela metade da distância entre esse item e outro com nível de freqüência igual ou superior ao dele e que lhe seja o mais próximo.

Assim, supondo que *A* e *B* são os mais próximos e do mesmo nível de freqüência, a área de influência mínima é o espaço em torno de cada um, distante até $D^n(A, B) / 2$.

Seja agora um terceiro item *C* com nível de freqüência inferior ao de *A* e *B* e que, no R^n, está mais próximo de *A* do que de qualquer outro item com nível de freqüência superior, inclusive *B*, isto é, $D^n(A, C) < D^n(B, C)$.

Diz-se, então, que *C* pertence a área de influência de *A*, que passa a ser chamado de pai ou pivô de *C*. Se $D^n(A, C) \leq D^n(A, B) / 2$, então *C* pertence a área de influência mínima de *A*, senão *C* pertence a área de influência expandida de *A*.

Como um determinado item (*C*), além da sua própria área de influência, pode pertencer a outra área de influência de um e somente um outro item (*A*), entre os de mesmo nível de freqüência que seja superior ao dele (*A* e *B*) e, também, para que seja garantida uma boa visualização no R^2, a menor distância $D^n(A, C)$ é sempre mantida e é verificado se a maior não está invadindo outra área de influência, isto é, no exemplo, se $D^n(B, C) \leq D^n(A, B) / 2$. Se a invasão não se confirmar, a maior distância também será mantida, caso contrário ela será substituída por um percentual de $D^n(A, B)$, definido pelo usuário, chamado ajuste de hierarquização (AH).

Desta forma, a menor distância é valorizada e, se necessário, *C* é empurrado para fora da área de influência de *B*, passando a pertencer exclusivamente a área de influência de *A*.

O resultado da projeção no R^2 retratará claramente que *A* está a uma distância relativamente grande de *B* e que *C* está mais próximo de *A* do que de *B*. A interpretação da projeção para efeito da organização da prateleira é direta: *A* e *B* devem estar distantes e *C* deve estar em um local próximo da *A*.

Generalizando, *A* é pai ou pivô de todos os itens que, como *C*, forem alocados em sua área de influência e todo item possui uma área de influência onde podem ser alocados outros itens.

Para que, por exemplo, um item *C* possa ser alocado na área de influência do item *A*, têm que ser atendidas as duas seguintes condições:

- O nível de freqüência de *C* tem que ser menor que o de *A*.
- *C* tem que estar mais próximo de *A* do que de qualquer outro item com nível de freqüência igual ou superior ao de *A* e que pertença à mesma área de influência em que *A* está alocado.

O percentual de ajuste da hierarquização (AH) é de fundamental importância para obtenção de uma boa projeção no R2. Em linhas gerais, ele determina o quanto um

item de baixo nível de freqüência deve ser deslocado para fora da área de influência de um outro que não é o seu pivô e possui nível superior ao dele. Na idéia geral anterior, foi afirmado que o teste de invasão de área de influência deveria ser feita com a metade da distância entre o pivô e o proprietário da área de influência em questão ($D^n(A, B) / 2$). Ocorre que, após diversos experimentos, verificou-se que muitas vezes a projeção de melhor qualidade era obtida com a definição de uma área de influência mínima com um raio menor ou maior do que a metade da distância. Assim, foi substituído o teste de invasão no programa HierarqD de $D^n(B, C) \leq D^n(A, B)/2$ para $D^n(B, C) \leq \text{AH}*D^n(A, B)$.

Para esclarecer melhor o funcionamento do algoritmo de hierarquização, é apresentado a seguir, um exemplo de sua execução sobre a matriz de distância *D* original da Tabela 4.10, supondo que o usuário tenha definido um ajuste de hierarquização de 100% (tratado pelo algoritmo como AH = 1).

A Tabela 4.11 mostra a mesma matriz *D*, identificando por cores os três níveis de freqüência dos itens de cada linha.

→	1	2	3	4	5	6	7	8	9	10	11
1											
2	0.2604										
3	0.1742	0.1725									
4	0.1538	0.1542	0.1321								
5	0.1310	0.1290	0.1133	0.1156							
6	0.1290	0.1310	0.1259	0.1133	0.1109						
7	0.1271	0.1274	0.1194	0.1194	0.1016	0.1016					
8	0.0429	0.0424	0.0382	0.0429	0.0259	0.0382	0.0415				
9	0.0292	0.0146	0.0231	0.0340	0.0335	0.0335	0.0340	0.0208			
10	0.0165	0.0098	0.0137	0.0182	0.0184	0.0169	0.0049	0.0143	0.0137		
11	0.0119	0.0033	0.0033	0.0148	0.0152	0.0033	0.0152	0.0101	0.0083	0.0148	

Nível 3
Nível 2
Nível 1

Tabela 4.11. Identificação dos Níveis de Freqüência na Matriz *D*.

O algoritmo de hierarquização usa as abreviaturas NF para nível de freqüência e AH para ajuste de hierarquização e executa os seguintes passos:

1. NF mais alto (NF = 3): manter as distâncias inalteradas.

2. Segundo NF (NF = 2) em relação ao NF mais alto (NF = 3) (retângulo definido entre as linhas de 3 a 7 e colunas de 1 a 2):

Para cada linha (item de NF = 2), fazer:

- Manter inalterada a menor distância. O item da coluna (de NF = 3) correspondente a essa menor distância passa a ser chamado de pivô (ou pai) do item da linha. O valor do pivô de cada linha passa a ser armazenado no vetor *PIVO*.

- Verificar se as demais distâncias da linha são maiores que o produto de AH pela distância entre o pivô e o item (de NF = 3) da coluna em questão (teste de invasão). Se for, manter a distância, senão, substituí-la por AH vezes a distância entre o pivô e o item da coluna em questão.

  Seja a análise do item 7: o item 7 está mais próximo do item 1 do que do 2, logo $D(7, 1)$ é mantida e 1 passa a ser o pivô da linha 7. As outras distâncias entre o item 7 e os demais itens do NF = 3 (no caso, somente o item 2) são verificadas se são maiores que a metade da distância entre o pivô e o item de NF = 3 da coluna em questão (no caso, o item 2), isto é, verifica-se: $D(7, 2) > AH * D(2, 1)$ ou $0.1274 > 1 * 0.2604 = 0.2604$.

  Neste caso, houve invasão do item 7 na área de influência do item 2 e, portanto, deve-se atribuir:

  $D(7, 2) \leftarrow AH * D(2, 1)$ ou $D(7, 2) \leftarrow 1 * 0.2604$.

  Em termos de hierarquia, como 7 está mais próximo de 1 do que de 2, ele pertence à área de influência de (ou é filho de) 1 e não de 2.

  Executando para todas as linhas do retângulo, a matriz *D* e o vetor *PIVO* ficam de acordo com a Tabela 4.12 abaixo.

3. Segundo NF (NF = 2) em relação a ele mesmo (triângulo definido entre as linhas de 4 a 7 e colunas de 3 a 6):

   Para cada linha (item de NF = 2), fazer:

   - Se o item da linha tiver o mesmo pivô do da coluna (forem irmãos), manter a distância, senão verificar se é maior que o produto de AH pela

distância entre os pivôs (pais) (teste de invasão). Se for, manter a distância, senão, substituí-la por AH * a distância entre os pivôs.

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.1321									1
5	0.2604	0.1290	0.1133	0.1156								2
6	0.1290	0.2604	0.1259	0.1133	0.1109							1
7	0.1271	0.2604	0.1194	0.1194	0.1016	0.1016						1
8	0.0429	0.0424	0.0382	0.0429	0.0259	0.0382	0.0415					
9	0.0292	0.0146	0.0231	0.0340	0.0335	0.0335	0.0340	0.0208				
10	0.0165	0.0098	0.0137	0.0182	0.0184	0.0169	0.0049	0.0143	0.0137			
11	0.0119	0.0033	0.0033	0.0148	0.0152	0.0033	0.0152	0.0101	0.0083	0.0148		

Tabela 4.12. Matriz *D* com NF = 2 Hierarquizado em relação ao NF = 3.

No exemplo, {3, 5} são filhos de 2 e {4, 6, 7} são filhos de 1. Então, as distâncias entre 3 e 5 e entre 4, 6 e 7, respectivamente, são mantidas, pois eles pertencem às mesmas áreas de influência. Neste caso, nenhuma das demais distâncias ({3, 5} ↔ {4, 6, 7}) passaram no teste de invasão e foram todas substituídas por AH * *D*(2, 1). A matriz *D* com essas alterações fica (Tabela 4.13):

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.2604									1
5	0.2604	0.1290	0.1133	0.2604								2
6	0.1290	0.2604	0.2604	0.1133	0.2604							1
7	0.1271	0.2604	0.2604	0.1194	0.2604	0.1016						1
8	0.0429	0.0424	0.0382	0.0429	0.0259	0.0382	0.0415					
9	0.0292	0.0146	0.0231	0.0340	0.0335	0.0335	0.0340	0.0208				
10	0.0165	0.0098	0.0137	0.0182	0.0184	0.0169	0.0049	0.0143	0.0137			
11	0.0119	0.0033	0.0033	0.0148	0.0152	0.0033	0.0152	0.0101	0.0083	0.0148		

Tabela 4.13. Matriz *D* com NF = 2 Hierarquizado.

4. Para todos os próximos níveis, repetir:

   Passo 4.1: NF atual (NF = 1) em relação ao NF mais alto (NF = 3): procedimento idêntico ao passo 2 do algoritmo: definindo a que área de influência pertence cada item (linha) do NF atual e o seu pivô (Tabela 4.14).

Passo 4.2: NF atual (NF = 1) em relação a todos os seus NF's antecessores (NF = 3 e NF = 2) (retângulo definido entre as linhas de 8 a 11 e colunas de 1 a 7): Para cada linha (item de NF = 1), fazer:

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.2604									1
5	0.2604	0.1290	0.1133	0.2604								2
6	0.1290	0.2604	0.2604	0.1133	0.2604							1
7	0.1271	0.2604	0.2604	0.1194	0.2604	0.1016						1
8	0.2604	0.0424	0.0382	0.0429	0.0259	0.0382	0.0415					2
9	0.2604	0.0146	0.0231	0.0340	0.0335	0.0335	0.0340	0.0208				2
10	0.2604	0.0098	0.0137	0.0182	0.0184	0.0169	0.0049	0.0143	0.0137			2
11	0.2604	0.0033	0.0033	0.0148	0.0152	0.0033	0.0152	0.0101	0.0083	0.0148		2

Tabela 4.14. Matriz *D* com NF = 1 Hierarquizado em relação ao NF = 3.

- Se o item da linha tiver o mesmo pivô do da coluna, manter a distância. Senão, verificar se há invasão e se houver, substituir por AH vezes a distância entre os pivôs (Tabela 4.15).

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.2604									1
5	0.2604	0.1290	0.1133	0.2604								2
6	0.1290	0.2604	0.2604	0.1133	0.2604							1
7	0.1271	0.2604	0.2604	0.1194	0.2604	0.1016						1
8	0.2604	0.0424	0.0382	0.2604	0.0259	0.2604	0.2604					2
9	0.2604	0.0146	0.0231	0.2604	0.0335	0.2604	0.2604	0.0208				2
10	0.2604	0.0098	0.0137	0.2604	0.0184	0.2604	0.2604	0.0143	0.0137			2
11	0.2604	0.0033	0.0033	0.2604	0.0152	0.2604	0.2604	0.0101	0.0083	0.0148		2

Tabela 4.15. *D* com NF = 1 Parcialmente Hierarquizado em relação aos NF's = 2 e 3.

- Percorre as linhas novamente e, entre as distâncias onde o item da linha tiver o mesmo pivô que o da coluna, identificar a menor distância, definindo um novo pivô para a linha. Na linha 8, por exemplo, serão comparadas as distâncias das colunas 2, 3 e 5, pois todas tem pivô = 2. A menor delas define para a linha 8 o seu *NOVOPIVO* = 5.

- Percorre as linhas novamente e, entre as distâncias onde o item da linha tiver o mesmo pivô do da coluna, manter a menor distância e verificar as demais com AH vezes a distância entre o *NOVOPIVO* e o item da coluna em questão e, se necessário, substituir por AH vezes essa distância.

Finalmente, atualizar o *PIVO* da linha com o valor do *NOVOPIVO*. No processo da linha 8, por exemplo, tem-se:

D(8, 5) será mantida (é a menor), *D*(8, 2) ← AH * *D*(5, 2), *D*(8, 3) ← AH * *D*(5, 3) e *PIVO*(8) ← *NOVOPIVO*.

Obs.: Na linha 11 há um empate no menor elemento (colunas 2 e 3). Neste caso, o programa adota o menor item entre essas colunas (item 2) como *NOVOPIVO*, por possuir maior força gravitacional (freqüência).

Executando para todas as linhas do NF atual (1), tem-se (Tabela 4.16):

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.2604									1
5	0.2604	0.1290	0.1133	0.2604								2
6	0.1290	0.2604	0.2604	0.1133	0.2604							1
7	0.1271	0.2604	0.2604	0.1194	0.2604	0.1016						1
8	0.2604	0.1290	0.1133	0.2604	0.0259	0.2604	0.2604					5
9	0.2604	0.0146	0.1725	0.2604	0.1290	0.2604	0.2604	0.0208				2
10	0.2604	0.0098	0.1725	0.2604	0.1290	0.2604	0.2604	0.0143	0.0137			2
11	0.2604	0.0033	0.1725	0.2604	0.1290	0.2604	0.2604	0.0101	0.0083	0.0148		2

Tabela 4.16. Matriz *D* com NF = 1 Hierarquizado em relação aos NF's = 2 e 3.

Passo 4.3: NF atual (NF = 1) em relação a ele mesmo (triângulo definido entre as linhas de 9 a 11 e colunas de 8 a 10): procedimento idêntico ao passo 3 do algoritmo.

A versão final da matriz *D* hierarquizada é apresentada na Tabela 4.17.

→	1	2	3	4	5	6	7	8	9	10	11	PIVO
1												1
2	0.2604											2
3	0.2604	0.1725										2
4	0.1538	0.2604	0.2604									1
5	0.2604	0.1290	0.1133	0.2604								2
6	0.1290	0.2604	0.2604	0.1133	0.2604							1
7	0.1271	0.2604	0.2604	0.1194	0.2604	0.1016						1
8	0.2604	0.1290	0.1133	0.2604	0.0259	0.2604	0.2604					5
9	0.2604	0.0146	0.1725	0.2604	0.1290	0.2604	0.2604	0.1290				2
10	0.2604	0.0098	0.1725	0.2604	0.1290	0.2604	0.2604	0.1290	0.0137			2
11	0.2604	0.0033	0.1725	0.2604	0.1290	0.2604	0.2604	0.1290	0.0083	0.0148		2

Tabela 4.17. Matriz *D* Hierarquizada.

Após a execução do processo de hierarquização, o programa grava um arquivo no formato do arquivo principal de entrada do sistema PEx para imediata projeção. No exemplo anterior, o programa gerou o arquivo DHierarqL-100.dmat, cujo mapa resultante é apresentado na Figura 4.5.

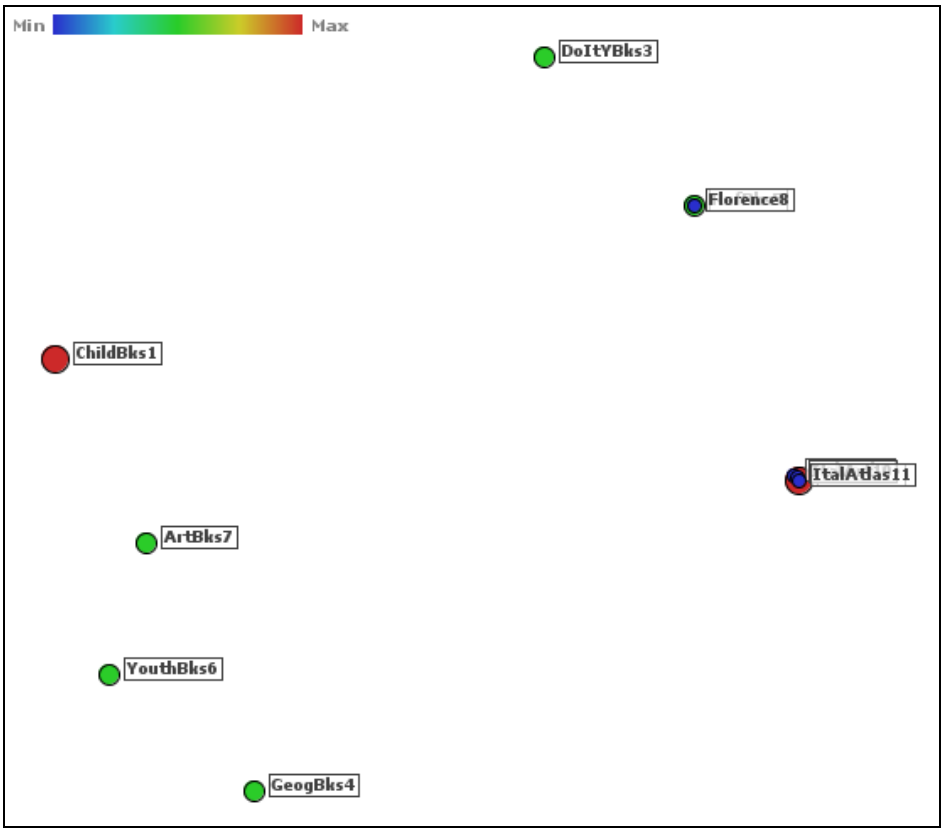


Figura 4.5. Matriz *D* com Ajuste de Hierarquização de 100%.

Com um percentual de 100% de ajuste de hierarquização, a imagem destaca claramente todas as estruturas. O problema agora é que os itens do nível de freqüência mais baixo foram sobrepostos aos seus pais, mantendo a dificuldade de análise visual das distâncias.

Evidentemente que, quanto maior o percentual de ajuste de hierarquização (AH) adotado, mais distâncias serão substituídas na matriz *D*. No exemplo anterior, a adoção de AH igual a 100% provocou a hierarquização mais radical. Se for adotado AH = 0 (zero), o processo não altera nenhuma distância, devolvendo como resultado a prórpia matriz *D* original, cujo mapa (Figura 3.12) apresentou dificuldades inversas de interpretação.

Para obter uma imagem com boa qualidade é necessário executar o programa repetidas vezes, alterando os valores do percentual de hierarquização e analisando a evolução dos mapas correspondentes, até alcançar o melhore ajuste.

De acordo com várias experiências realizadas, observou-se que as melhores imagens ocorrem para valores altos do percentual. Portanto, uma estratégia é diminuir a hierarquização a partir de 100, de 10 em 10 ou de 20 em 20, por exemplo, para chegar numa primeira aproximação e, em seguida, realizar uma sintonia mais fina, se necessário.

Usando a base de dados Livro, foram gerados onze mapas seguindo a estratégia e decrescendo a hierarquização de 10 em 10 (Figuras de I.1 a I.11 do ANEXO I). Na análise seqüencial dessas imagens, pode ser observada a desconstituição das estruturas hierárquicas. Entre elas, as que representam com mais clareza a distribuição dos itens e suas estruturas são as das Figuras I.3, I.4 e I.5 e, por isso foram geradas mais duas imagens, respectivamente, com 65 e 75% de hierarquização (Figuras de I.12 e I.13 do ANEXO I).

Finalmente, o Mapa da Figura I.12, com percentual de hierarquização ajustado em 65% pode ser considerado satisfatório. O único problema que persiste são os itens 9, 10 e 11 do nível de freqüência 1, que ainda estão muito próximos, dificultando as suas identificações. Este problema é resolvido pela terceira tarefa realizada pelo programa HierarqD, através de um método complementar de otimização das distâncias em cada área de influência mínima, que será descrito mais adiante.

O método de hierarquização ainda pode ser considerado sob outros aspectos.

Em termos estruturais, o processo constrói uma floresta, onde cada item do nível mais alto é a raiz de uma árvore e uma área de influência é representada por uma árvore ou sub-árvore.

Na construção da floresta são mantidas as distâncias entre pais e filhos e entre irmãos que tenham o mesmo nível de freqüência (inclusive as raízes). As demais distâncias podem, se necessário, ser substituídas, de acordo com o percentual de ajuste de hierarquização adotado, afastando neste caso árvores ou sub-árvores do item em análise.

A floresta gerada na hierarquização da matriz *D* do exemplo anterior é apresentada na Figura 4.6 a seguir.

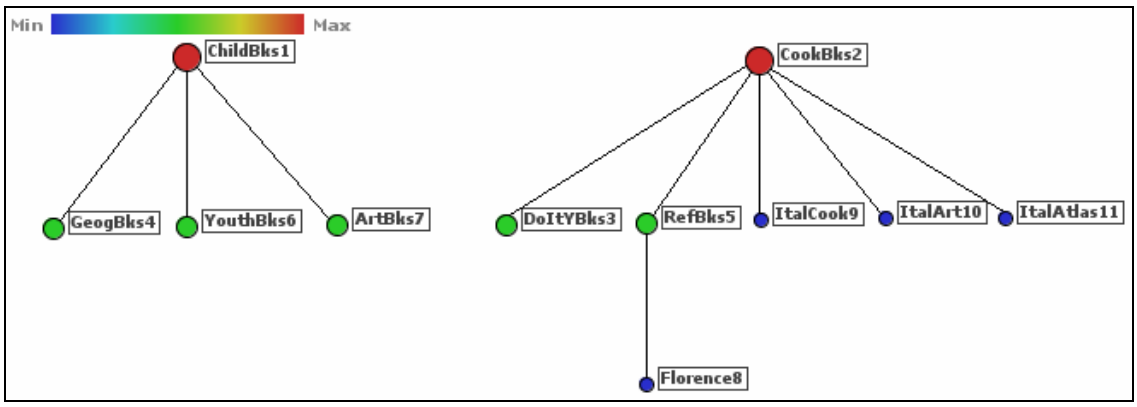


Figura 4.6. Matriz *D* Hierarquizada Representada na Forma de Floresta.

Sob um outro ponto de vista, o processo traduz um agrupamento hierárquico, onde cada área de influência constitui um grupo. Este agrupamento também pode ser uma ferramenta importante para a análise de associações.

A representação do agrupamento formado pela hierarquização da matriz *D* em 65%, é mostrado na Figura 4.7 a seguir.

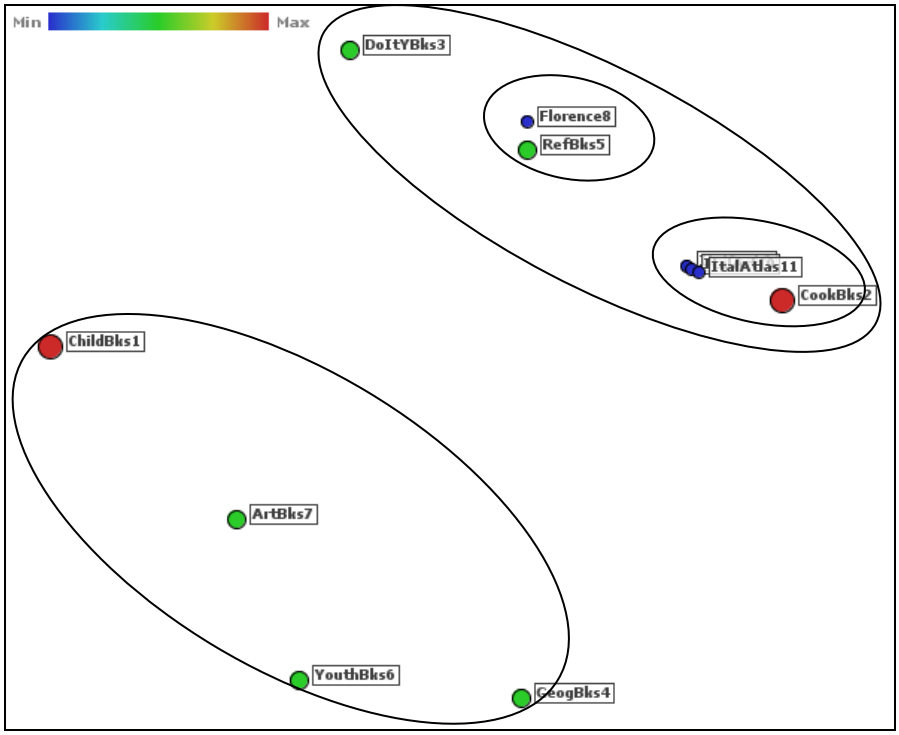


Figura 4.7. Representação do Agrupamento Hierárquico da matriz *D*.

Terminada a hierarquização, o programa parte para a execução da otimização das distâncias dentro de cada área de influência mínima.

Em resumo, trata-se de um processo de maximização linear das distâncias, onde, a partir do segundo nível de freqüência, é determinado, inicialmente, se todos os irmãos estão dentro da mesma área de influência mínima. Neste caso, a maior das distâncias entre os irmãos ou entre cada um deles e o pai é maximizada para sua área de influência e é calculado o fator de acréscimo sofrido por esta distância. Em seguida, todas as distâncias entre o pai e os filhos e entre os irmãos são multiplicadas por este fator, mantendo, desta forma, uma proporcionalidade entre essas distâncias e otimizando a ocupação dessa área de influência mínima por uma melhor distribuição de seus itens.

Na execução do método, o usuário também define um percentual de ajuste de otimização (OT) que é multiplicado ao fator de acréscimo com o objetivo de ajustar a otimização para obtenção de uma imagem de boa qualidade.

Para exemplificar, seja otimizar a área de influência mínima ocupada pelos itens 9, 10 e 11, de nível de freqüência 1 na Figura 4.7, considerando que o usuário forneceu um percentual de ajuste igual a 100% (OT = 1).

Os itens {9, 10, 11} são filhos diretos do item 2 e formam com ele uma área de influência restrita a ser otimizada.

A distância que será analisada quanto a otimização é a maior distância (DMAX) entre 2, 9, 10, e 11, que, no caso, é $D(11, 10) = 0.0148$ (Tabela 4.17). Essa análise deve verificar se é possível maximizar DMAX = $D(11, 10)$ dentro da área de influência restrita mínima do item 2 (do pai).

Por ser de nível 3, o item 2 possui ainda, uma área de influência ampla com os filhos {3, 5} de nível 2.

Como dentro dessa área de influência ampla do item 2, os itens {2, 3, 5} possuem as suas próprias áreas de influência (Figura 4.7) e para que não haja invasão de outras áreas na maximização de DMAX, determina-se a distância ótima (DOTM) para a área de influência restrita de 2, como sendo a metade da menor das distâncias entre 2 e 3 ou entre 2 e 5. Portanto, DOTM = $D(5, 2)/2 = 0.1290/2 = 0.0645$ (Tabela 4.17).

Analisando, como DOTM = 0.0645 é maior do que DMAX = 0.0148, é possível otimizar as distâncias entre os itens 2, 9, 10 e 11 dentro da área de influência restrita mínima do item 2.

A distância máxima é substituída pelo produto do ajuste de otimização (OT) vezes a distância ótima ($D(11, 10) \leftarrow 1 * 0.0645$) e, para que haja proporcionalidade na substituição das demais distâncias, é calculado o fator de correção F = OT * DMAX / DOTM (= 1 * 0.0645 / 0.0148 = 4.3585).

Finalmente, multiplica-se F por todas as distâncias entre o pai (2) e os filhos (9, 10 e 11) e entre os filhos.

A Figura 4.8 mostra a projeção da matriz *D* com 65% de hierarquização e 100% de otimização, onde a área de influência mínima do item 5 também foi otimizada e as demais mantidas.

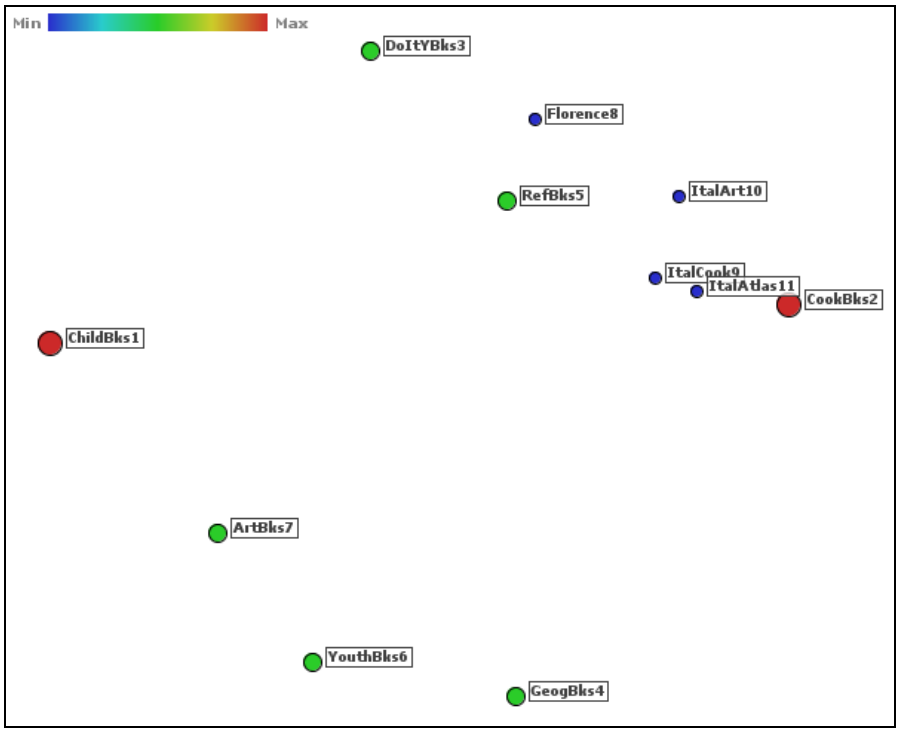


Figura 4.8. Matriz *D* com 65% de hierarquização e 100% de otimização.

Nota-se uma sensível melhora na qualidade da projeção, embora o ajuste da otimização em 100%, aproximou demasiadamente os itens 9 e 10 do item 5, a ponto de causar dúvida se eles pertencem a área de influência 2 ou 5.

Por isso, é também necessário sintonizar o ajuste de otimização, a exemplo do que foi feito com a hierarquização.

As Figuras de II.1 a II.11 do ANEXO II apresentam os mapas gerados com a base de dados Livro, mantendo o ajuste de hierarquização em 65% e variando o de otimização, a partir de 0%, de 10 em 10, até 100%.

Analisando visualmente essas imagens, foi escolhida a da figura II.8, ajustada com 65% de hierarquização e 70% de otimização como a que melhor representa a distribuição dos itens da base de dados no R^2, inclusive os de nível de freqüência 1, sem a necessidade de sintonia fina no ajuste de otimização. Uma cópia dessa imagem é apresentada a seguir (Figura 4.9).

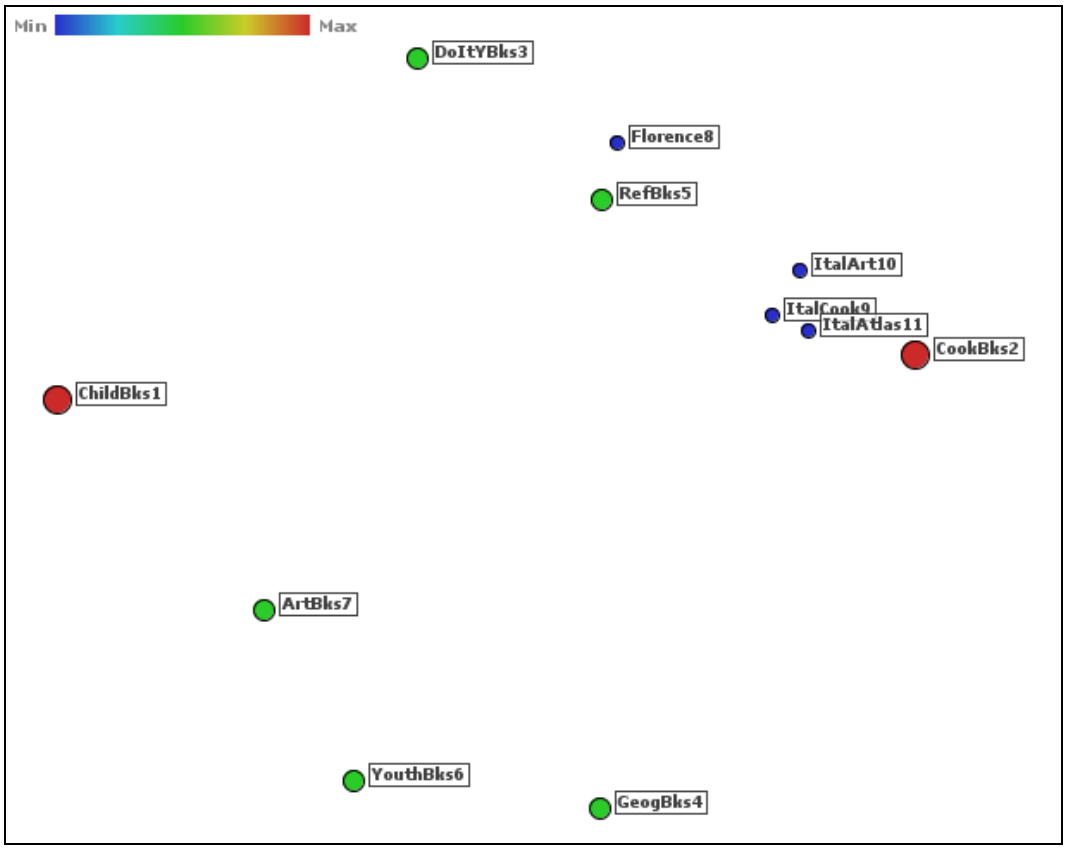


Figura 4.9. Matriz *D* com 65% de Hierarquização e 70% de Otimização.

A análise visual deste mapa permite diversas formas de interpretações.

Em um aspecto mais geral e considerando que os itens neste exemplo estão em alto nível de abstração (tipos de livros), tem-se, de imediato, uma alternativa inteligente

para a planta baixa da livraria, baseada nas compras de seus clientes em um determinado período de tempo.

Olhando mais de perto, pode-se identificar as forças das associações através das distâncias entre os itens. Por exemplo: os itens 1 e 2 são os mais fortemente associados e são os que melhores atendem a estratégia de *marketing* (1); a associação entre o item 4 e o item 1 é mais forte do que entre o item 7 e o item 1, embora esta também seja importante para efeito de uma ação baseada na estratégia (2); o problema do item raro é facilmente identificado entre o item 5 e o item 8 e entre os itens 9, 10 e 11 e o item 1, etc.

É evidente que o mapa, por si só, não encerra o problema de identificação de associações. Ele é, na verdade, uma ferramenta poderosa que deve funcionar iterativamente com outros programas, especialmente um para mineração de regras de associação.

Desta forma, em uma análise mais detalhada, pode-se selecionar subconjuntos de itens para mineração de regras, tornando essa tarefa mais objetiva, com maior facilidade de interpretação dos resultados e com melhor desempenho. Um exemplo seria a mineração das regras de associação envolvendo somente os itens que pertencem à área de influência do item 1, isto é, os itens 1, 4, 6 e 7. Outro, seria minerar as regras com os filhos do item 1 (4, 6 e 7) e o item 2, que também são regras bastante fortes.

Buscando ainda mais detalhes, se a base de dados dispõe dos subitens que compõem cada item apresentado, pode-se gerar outros mapas e/ou minerar regras multiníveis para identificar, por exemplo, quais títulos de livros (baixo nível de abstração) dos que constituem o item 2 estão associados ao tipo de livro (alto nível) representado pelo item 9 e, a partir deste conhecimento, criar promoções que possam impulsionar as vendas do item 9. Outro exemplo prático neste sentido é a elaboração de um catálogo inteligente de livros, podendo constar tanto títulos, quanto tipos de livros.

Muitas são as opções de trabalho, tornadas conscientes agora, por uma única imagem.

Na finalização da execução do programa HierarqD, ele ainda grava um outro arquivo, que é usado como arquivo de entrada do programa ClustD, com todo o

conteúdo do arquivo de hierarquização acrescido do número total de níveis de freqüência e do vetor PIVO.

4.2.3 Programa ClustD

O vetor PIVO, criado por necessidade de funcionamento do método de hierarquização, contem a representação do agrupamento hierárquico dos itens da base de dados. Cada um de seus elementos indica o código do pai do item, que, por sua vez é representado pelo índice da posição.

A sua existência motivou o desenvolvimento do programa ClustD que grava um arquivo no formato do arquivo principal de entrada do sistema PEx, contendo um novo vetor com informações sobre os grupos e a versão final da matriz *D* (hierarquizada e otimizada), que, quando projetada no plano, distingue cada agrupamento formado pela cor dos seus itens componentes.

Como cada item do nível de freqüência mais alto dá origem a uma árvore, isto é, a um grande grupo, a idéia inicial de funcionamento do ClustD é representar a divisão da escala de cores do sistema PEx em tantas sub-escalas quanto o número de itens pertencentes a esse nível de freqüência mais alto. Em seguida, o vetor PIVO é transformado, de forma a atribuir para cada item o valor da cor que ele ocupa dentro da sua sub-escala.

O mapa do agrupamento correspondente à matriz *D* da base de dados Livro com 65% de hierarquização e 70% de otimização é apresentado na Figura 4.10.

A projeção da matriz *D* representando o agrupamento hierárquico, agora apoiado pelo novo vetor gerado pelo programa ClustD, identifica claramente as estruturas hierárquicas criadas a partir da definição dos níveis de freqüência pelo usuário, reforçando ainda mais as possibilidades de análise do minerador, até mesmo para auxiliá-lo na obtenção da melhor imagem de distribuição dos itens no R^2.

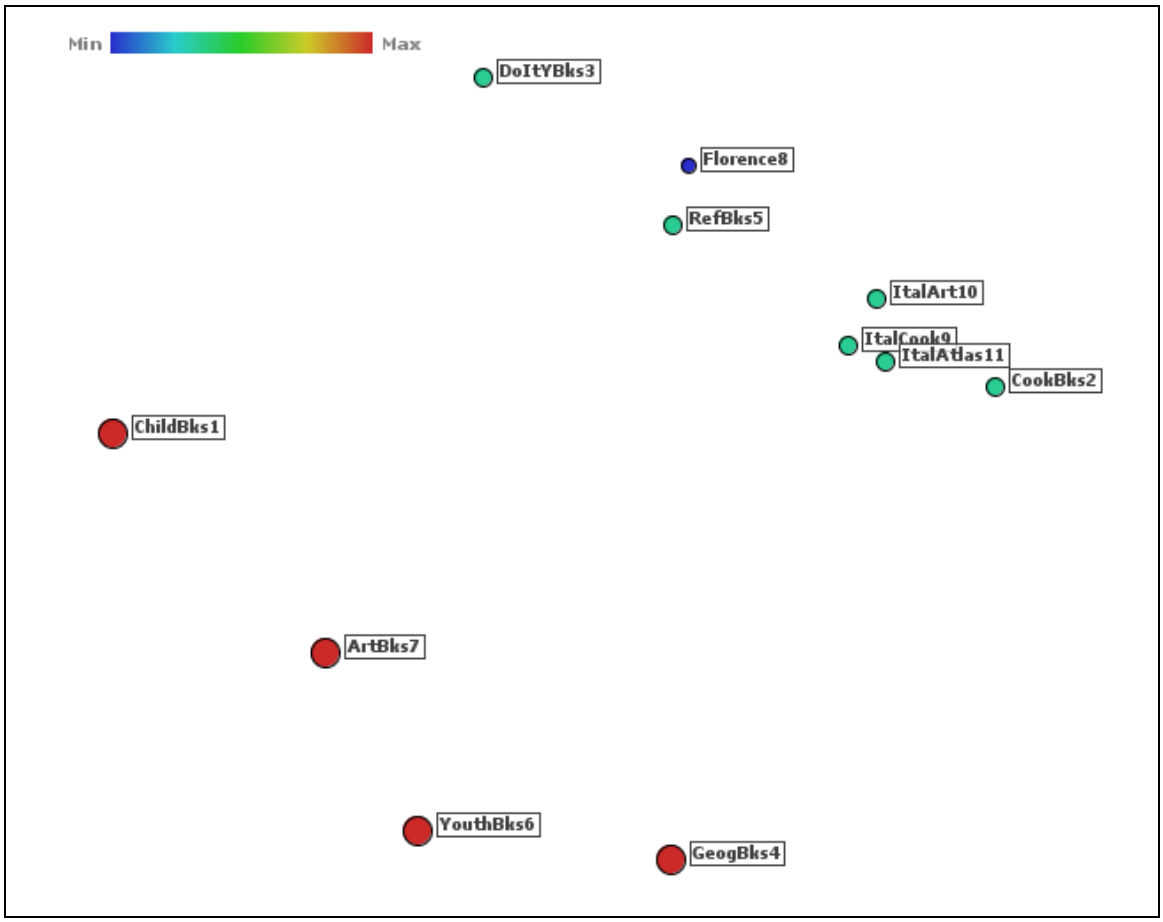


Figura 4.10. Agrupamento da Matriz *D* com 65% de Hierarquização e 70% de Otimização.

Em resumo, o sistema CalcD realiza as seguintes tarefas:

- Programa GeraD: minera uma base de dados de transações, obtendo as regras de associações mais fortes de todos os 2-*itemsets*. Usa o conhecimento descoberto para aplicar uma função de distância baseada na estrutura suporte-confiança, calculando a matriz *D* original com as distâncias entre todos os pares de itens.

- Programa HierarqD: trata distâncias nulas. Em seguida, usa como base a definição de níveis de freqüência realizada pelo usuário para organizar os itens em estruturas hierarquizadas por nível de freqüência, permitindo um ajuste das distâncias entre essas estruturas e dentro de cada uma delas, de forma a valorizar as distâncias mais importantes na matriz *D* e possibilitar uma projeção dos itens no R^2 com boa qualidade para efeito de análise

visual. O programa grava ainda um arquivo com um vetor que identifica as estruturas hierárquicas.

- Programa ClustD: lê o arquivo gravado anteriormente, altera o vetor para que os itens pertencentes a cada estrutura passem a ocupar uma sub-escala determinada de cores, viabilizando uma nova projeção dos itens, onde cada estrutura passa a ser representada por um grupo de itens bem definido.

As aplicações realizadas neste capítulo utilizaram a base de dados Livro, que é bastante simples, envolvendo somente onze itens em alto nível de abstração (tipos de livros). Mesmo assim, pode-se observar o grande potencial dos dois mapas gerados (itens por nível de freqüência e itens agrupados por estrutura hierárquica) no que diz respeito à quantidade de conhecimento que eles carregam para efeito de facilidade e diversidade de análise.

Deve-se destacar, entretanto, que a hierarquia entre os itens aqui utilizada é identificada, tão somente, pelos níveis de freqüência definidos pelo usuário, isto é, quanto maior o nível de freqüência de um item, mais alta é a sua posição em sua estrutura. Da mesma forma, cada estrutura hierárquica assim definida é constituída por um conjunto específico de itens, que passou a ser chamado de um grupo.

O capítulo a seguir apresenta a experiência da aplicação do sistema CalcD sobre uma base de dados real, bem como as dificuldades encontradas e os resultados obtidos.

CAPÍTULO 5
REGRAS DE ASSOCIAÇÃO MULTINÍVEIS

Na busca por uma aplicação prática que pudesse ilustrar as funcionalidades do sistema CalcD como ferramenta de análise, utilizou-se uma base de dados de transações mensais de um mercado, constituída de itens em baixo nível de abstração.

Numa primeira experiência e após a preparação dos dados de um mês, a mineração de regras de associação forneceu resultados muito pouco significativos. A realização de uma análise nos dados apontou duas causas para o problema: o baixo nível de abstração dos itens e uma razoável ocorrência de transações com poucos itens envolvidos.

Tornou-se então necessário o desenvolvimento de dois sistemas, sendo um para preparação dos dados, que, entre outras tarefas, possibilita criar níveis de abstração mais elevados e outro conseqüente, para seleção multinível de um subconjunto de itens para análise.

A integração desses dois sistemas à possibilidade de análise gráfica, através dos sistemas CalcD e PEx, bem como da utilização de sistemas de mineração de regras de associação e de pós-análise subjetiva, permitiram a proposição de uma metodologia iterativa de mineração de regras de associação multiníveis.

A proposta dessa metodologia e a descrição dos sistemas de preparação de dados e de seleção multinível são apresentadas a seguir.

5.1 Visão Geral da Metodologia

A necessidade comum de executar repetidas vezes o processo de mineração na busca de bons resultados e a incorporação da análise gráfica neste processo, motivaram a elaboração de uma metodologia iterativa para mineração de regras de associação multiníveis.

A visão geral dessa metodologia e a ligação entre seus sistemas são apresentadas na Figura 5.1 a seguir.

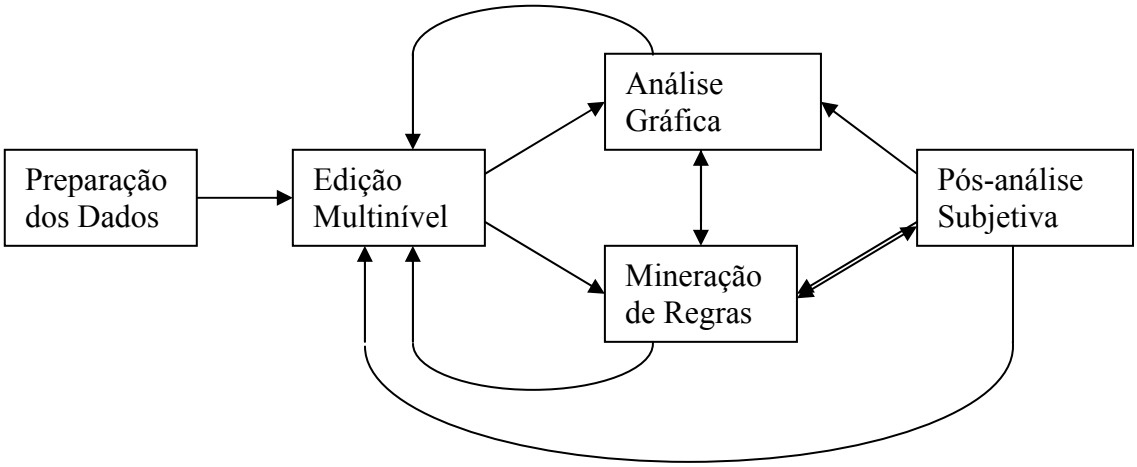


Figura 5.1. Visão Geral da Metodologia.

5.2 Sistema de Preparação dos Dados

Os objetivos gerais deste sistema são: realização do tratamento inicial dos dados, eliminando possíveis inconsistências, permitir a criação de dois níveis de abstração mais elevados para cada item (níveis 1 e 2), a partir do seu nome (nível 0) e possibilitar a realização de uma pré-análise nos dados.

A base de dados é composta de um arquivo contendo o Cadastro de Itens por Departamento e arquivos mensais de transações.

Os objetivos específicos do sistema são:

- Realizar a limpeza dos dados, eliminar algumas possíveis inconsistências e prepará-los para a mineração de regras de associação.

- Criar dois níveis de abstração mais elevados para os itens (nível 0, isto é, o nível de abstração mais baixo):

  nível 1 – Produto Básico.

  nível 2 – Produto Generalizado.

Exemplo: fornecido o item ABACAXI CALDA D CERRADO 400G (nível 0) foram criados os níveis produto básico ABACAXI CALDA (nível 1) e produto generalizado DOCE CALDA (nível 2).

Desta forma, pretende-se gerar regras de associação mais significativas, alem de possibilitar a mineração multinível.

- Produzir estatísticas (contagens, tabelas, histogramas e gráficos) para subsidiar o minerador de dados na geração das regras de associação mais interessantes e fornecer maior apoio ao especialista do domínio (proprietário da loja) na tomada de decisões.

Para permitir a criação dos níveis de abstração 1 e 2, o sistema monta uma tabela contendo os níveis de abstração 0, 1, 2 e 3 de cada item cadastrado, correspondendo a Itens Cadastrados, Produto Básico, Produto Generalizado e Departamento, respectivamente. Um exemplo é mostrado na Tabela 5.1.

Níveis de Abstração - Itens Cadastrados				
Nível 0	**Nível 1**	**Nível 2**	**Nível 3**	
Itens Cadastrados*	**Produto Básico**	**Produto Generalizado**	**Departamento**	
			Nome	**Cód.**
ABACAXI CRISTALIZADO			BALANCA	1
ABACAXI CALDA D CERRADO 400G	ABACAXI CALDA	DOCE CALDA	SOBREMESAS	12
...	...	...	...	...

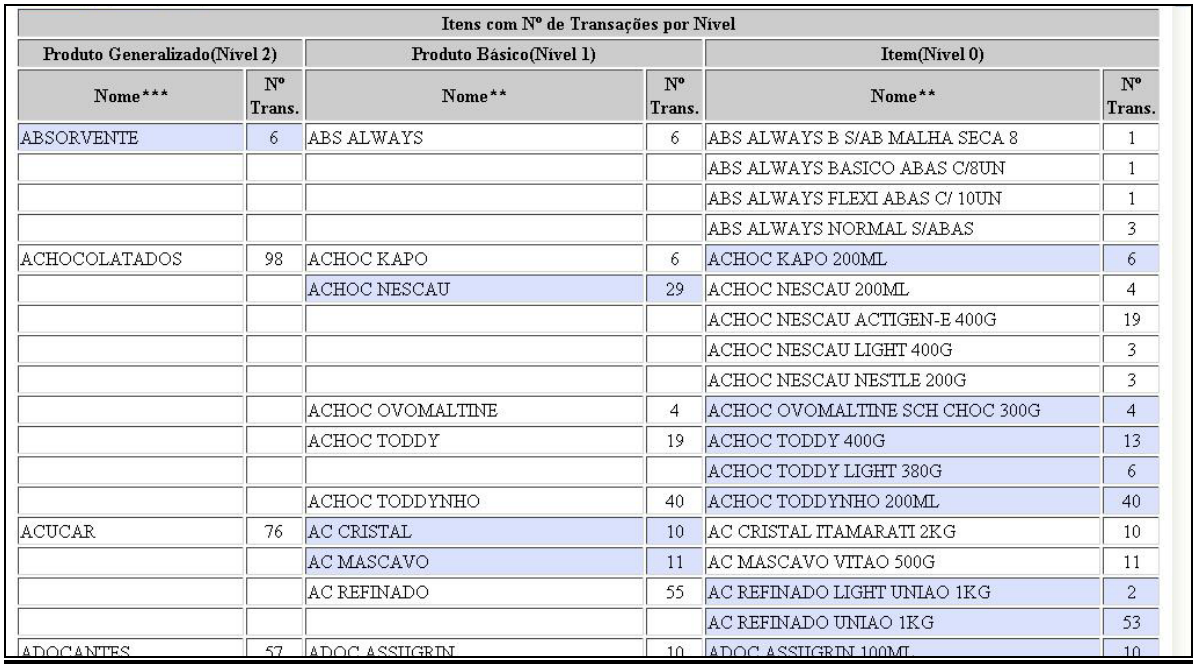
Tabela 5.1. Níveis de Abstração

A maior dificuldade encontrada neste sistema foi a criação manual dos níveis de abstração 1 e 2 para uma quantidade muito elevada de itens. Além disso, inicialmente pensava-se em usar o departamento como nível de abstração mais elevado, o que se mostrou inviável, uma vez que diversos itens de mesma natureza estavam cadastrados em departamentos distintos por questão organizacional.

5.3 Sistema de Edição Multinível

Este sistema calcula a freqüência de ocorrência de cada item em cada nível de abstração (0, 1 e 2) e cria uma interface para seleção multinível de um subconjunto de

itens. Realizada a seleção pelo usuário, o sistema grava um arquivo de transações, envolvendo somente os itens selecionados. A Figura 5.2 apresenta a tela dessa interface com alguns itens selecionados.



Itens com Nº de Transações por Nível					
Produto Generalizado(Nível 2)		Produto Básico(Nível 1)		Item(Nível 0)	
Nome***	Nº Trans.	Nome**	Nº Trans.	Nome**	Nº Trans.
ABSORVENTE	6	ABS ALWAYS	6	ABS ALWAYS B S/AB MALHA SECA 8	1
				ABS ALWAYS BASICO ABAS C/8UN	1
				ABS ALWAYS FLEXI ABAS C/ 10UN	1
				ABS ALWAYS NORMAL S/ABAS	3
ACHOCOLATADOS	98	ACHOC KAPO	6	ACHOC KAPO 200ML	6
		ACHOC NESCAU	29	ACHOC NESCAU 200ML	4
				ACHOC NESCAU ACTIGEN-E 400G	19
				ACHOC NESCAU LIGHT 400G	3
				ACHOC NESCAU NESTLE 200G	3
		ACHOC OVOMALTINE	4	ACHOC OVOMALTINE SCH CHOC 300G	4
		ACHOC TODDY	19	ACHOC TODDY 400G	13
				ACHOC TODDY LIGHT 380G	6
		ACHOC TODDYNHO	40	ACHOC TODDYNHO 200ML	40
ACUCAR	76	AC CRISTAL	10	AC CRISTAL ITAMARATI 2KG	10
		AC MASCAVO	11	AC MASCAVO VITAO 500G	11
		AC REFINADO	55	AC REFINADO LIGHT UNIAO 1KG	2
				AC REFINADO UNIAO 1KG	53
ADOCANTES	57	ADOC ASSUGRIN	10	ADOC ASSUGRIN 100ML	10

Figura 5.2. Tela de Edição Multinível.

Evidentemente, este sistema é de fundamental importância na geração de transações multiníveis. No momento da seleção de um determinado item, os seus correspondentes dos outros níveis de abstração são automaticamente desabilitados.

Por outro lado, foi observada uma deficiência em relação ao tempo de execução na geração do arquivo de transações multiníveis, quando eram selecionados muitos itens em diferentes níveis. Esse fato se deve à enorme quantidade de comparações e substituições de itens no arquivo original de transações, onde os itens estão no nível mais baixo (nível 0).

5.4 Análise Gráfica

Os arquivos de saída do Editor Multinível foram gerados para serem usados diretamente pelo sistema GeraD com o objetivo de produzir projeções gráficas ou podem ser usados por um sistema de mineração de regras de associação.

Para ilustrar, as Figuras 5.3 e 5.4 apresentam as projeções por nível de freqüência e por grupos, respectivamente, geradas com todos os itens do nível de abstração 2 com a base de dados de transações de um determinado mês. Foram usados cinco níveis de freqüência e a matriz foi ajustada com 95% de hierarquização e 5.7% de otimização. No mapa foram nomeados somente os seis itens com nível de freqüência mais alto (5).

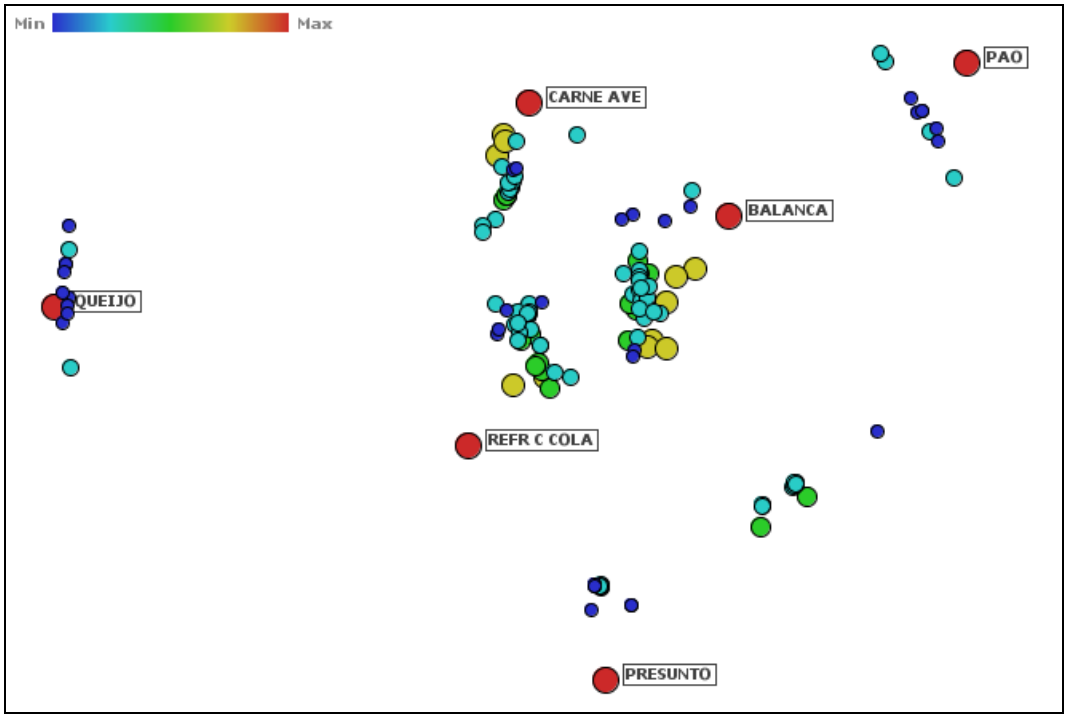


Figura 5.3. Distribuição dos Itens por Nível de Freqüência.

Essas projeções mostram claramente a identificação dos seis grupos, permitindo uma ampla variedade de análises, tanto gráficas quanto pela utilização de sistema de mineração de regras.

Seja, por exemplo, verificar graficamente os possíveis itens raros que se associam com o item Vinho, que pertence ao grupo do item Carne de Ave. Para isso, pode-se gerar uma projeção com somente os itens desse grupo, excluindo Carne de Ave. O resultado é apresentado na Figura 5.4.

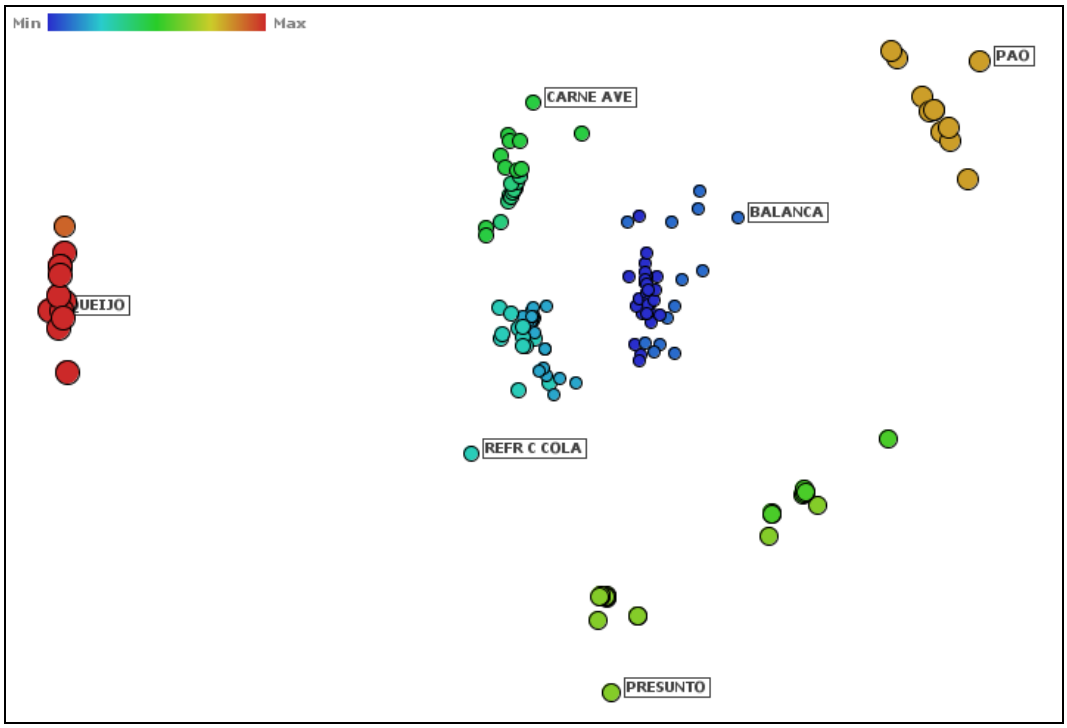


Figura 5.4. Distribuição dos Itens por Grupos.

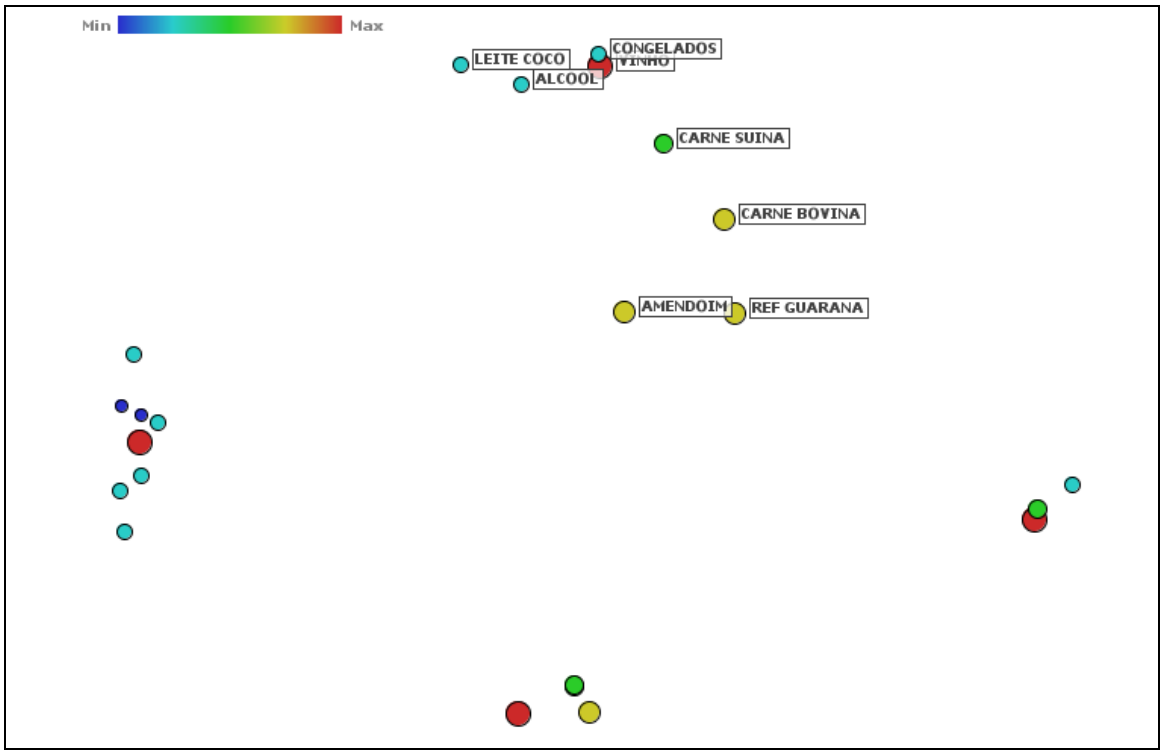


Figura 5.5. Possíveis Itens Raros Associados ao Item Vinho.

CAPÍTULO 6
CONCLUSÃO

A presente tese apresentou uma função baseada na estrutura suporte-confiança, para calcular a distância entre dois itens envolvidos em uma regra de associação, a partir das duas estratégias de *marketing* definidas pela análise de cestas de compras: dois itens que vendem bem individualmente e conjuntamente, devem estar distantes e dois itens que vendem bem conjuntamente e um deles vende bem individualmente, devem estar próximos. A função atendeu bem a todos os diversos testes realizados.

Ela foi, então, incorporada a um sistema inteligente que usa o conhecimento descoberto na mineração das regras de associação de todos os 2-*itemsets* existentes em uma base de dados, montando uma matriz com as distâncias entre os itens no R^n.

Usando um sistema de projeção multidimensional, a imagem da matriz produzida no R^2 não se mostrou satisfatória. Retomando o raciocínio das estratégias de *marketing*, foi elaborado um método para valorizar as distâncias realmente importantes, criando uma hierarquização por níveis de freqüência dos itens e guiada pelo usuário. Em complementação, um outro método, também guiado pelo usuário, otimiza as distâncias dos itens dentro de cada estrutura hierárquica.

Um último método disponível no sistema trata cada estrutura hierárquica para efeito de análise de agrupamento.

Ajustadas a hierarquização e a otimização, as saídas do sistema possibilitam a geração de duas projeções da matriz de distância, destacando os níveis de freqüência e os agrupamentos hierárquicos. Essas imagens carregam uma grande quantidade de informações, tanto de uso prático mais imediato, quanto indicando caminhos mais técnicos a percorrer.

A tentativa inicial de usar o sistema sobre uma base de dados real, não apresentou bons resultados. A principal razão foi que o baixo nível de abstração dos dados causava uma enorme quantidade de regras muito fracas. Uma análise estatística dos dados mostrou o perfil de vendas mensais da loja, enquanto um sistema de

preparação dos dados foi desenvolvido com o principal objetivo de criar mais dois níveis de abstração para os produtos.

Agora, cada produto pode ser representado em três níveis de abstração e os testes com o sistema de projeção gráfica apontavam para a possibilidade de mineração de regras sobre subconjuntos selecionados de itens. Restava o desenvolvimento de um terceiro sistema de seleção dos itens a serem analisados em múltiplos níveis.

O funcionamento desses três sistemas acompanhados por um outro específico de mineração de regras constituiu numa metodologia iterativa de mineração de regras de associação multiníveis, que faz uso de diversas formas de restrições (de tempo, de *itemsets*, de níveis de freqüência, de níveis de abstração, etc.), tornando a tarefa mais objetiva na obtenção de soluções de problemas.

Considerando que ainda há muito por fazer, são sugeridos os seguintes trabalhos futuros:

- A inclusão de um método mais específico de projeção gráfica ao próprio sistema de geração da matriz de distância visando maior desempenho e facilidade de ajuste da hierarquia e da otimização.

- Estudo de um método que possa automatizar a escolha da quantidade dos níveis de freqüência, bem como de quais itens devem pertencer a cada nível.

- Estudo sobre o impacto da utilização de outras medidas de interessabilidade na composição da função de distância.

Além da contribuição da metodologia iterativa de mineração de regras de associação multiníveis, o uso da função de distância para geração de imagens que possibilitam resolver diversos tipos de problemas, foram os pontos mais fortes apresentados nessa tese.

Objetividade é talvez a melhor palavra para caracterizar a funcionalidade de boas projeções de itens no R^2. O uso irrestrito dos mapas direciona o minerador, desde antes da obtenção da primeira regra de associação, sugerindo caminhos promissores, até a fase de pós-análise, auxiliando na avaliação dos conhecimentos encontrados, de forma a suportar tomadas de decisões inteligentes.

REFERÊNCIAS BIBLIOGRÁFICAS

AGRAWAL R., IMIELINSKI T. & SWAMI A. N., "Mining Association Rules between Sets of Items in Large Databases". In: *Proc. of the ACM SIGMOD Intl. Conf. on Management of Data (SIGMOD'93)*, pp. 207–216, Washington, USA, 1993.

AGRAWAL R. & SRIKANT R., "Fast algorithms for mining association rules". In: *Int. Conf. Very Large Data Bases (VLDB'94)*, pp. 487-499, Santiago, Chile, 1994.

ALVES A. S., "Regras de Associação e Classificação em Ambiente de Computação Paralela Aplicadas a Sistemas Militares". *Tese de Doutorado*, COPPE/UFRJ, 2007.

BISPO C. A, F., "Uma Análise da Nova Geração de Sistemas de Apoio a Decisão". *Dissertação de Mestrado*, Escola de Engenharia de São Carlos/Universidade de São Paulo, São Carlos, SP, Brasil, 1998.

CANO P., KALTENBRUNNER M., GOUYON F. & BATTLE E., "On the Use of FastMap for Audio Retrieval and Browsing". In: *Proceedings of 3rd Conference on Music Information Retrieval*, Paris, França, 2002.

CHEN M. & LIN C., "A Data Mining Approach to Product Assortment and Shelf Space Allocation", Expert Systems with Applications, Vol. 32, No. 4, pp. 976-986, Pergamon Press, 2007.

CUROTTO C. L., "Integração de Recursos de Data Mining com Gerenciadores de Bancos de Dados Relacionais", *Tese de Doutorado*, COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2003.

FALOUTSOS C. & LIN K., "FastMap: A Fast Algorithm for Indexing Data Mining and Visualization of Traditional and Multimedia Datasets". In: *Proc. of the Intl. Conf. on Management of Data (SIGMOD'95)*, pp. 163-174, San Jose, USA, 1995.

FAYYAD U. M., PIATETSKY-SHAPIRO, G. & SMITH, P., "From Data Mining to Knowledge Discovery: An Overview". In: *Advances in Knowledge Discovery and Data Mining*, FAYYAD, U.M. *et alii* (eds.), AAAI/MIT Press, pp. 1-36, 1995.

FAYYAD U.M., PIATETSKY-SHAPIRO, G. & SMITH, P., "From Data Mining to Knowledge Discovery in Databases". *Al Magazine*, Vol. 17, No. 3, pp. 37-54, 1996[a].

FAYYAD U.M., PIATETSKY-SHAPIRO, G. & SMITH, P. "Knowledge Discovery and Data Mining: Towards a Unifying Framework". In: *Proc. Of the KDD'96, 2nd Int'l Conf. on Knowledge Discovery and Data Mining*, Portland, Oregon, USA, pp. 82-88, 1996[b].

HAIR J. F., BLACK B., BABIN B., ANDERSON R. E., TATHAM R. L., "Multivariate Data Analysis". 6th ed., Prentice Hall, 2005.

HAN J. & KAMBER, M., "Data Mining: Concepts and Techniques". 2nd ed., Morgan Kaufmann Publishers, 2006.

HAN J. & KAMBER, M., "Data Mining: Concepts and Techniques, Slides for Textbook, Chapter 5", http://www.cs.sfu.ca. Último acesso em: Março de 2010.

HAO M. C., DAYAL U., HSU M., SPRENGER T. & GROSS M., "Visualization of Directed Association in e-Commerce Transaction Data". In: *Joint Eurographics - IEEE TCVG Symposium on Visualization*, pp. 185-192, 2001.

HASTIE T., TIBSHIRANI R. & FRIEDMAN J., "The Elements of Statistical Learning: Data Mining, Inference and Prediction". Springer-Verlag New York, 2001.

JOHNSON R. A., WICHERN D. W., "Applied Multivariate Statistical Analysis". 4th ed., Prentice Hall, 1998.

LIU B., "Web Data Mining", Springer-Verlag Berlin Heidelberg, 2008.

MANLY B. F. J., "Multivariate Statistical Methods: A Primer". 4th ed., Chapman and Hall, 2004.

MARDIA V. K., KENT J. T., BIBBY J. M., “Multivariate Analysis”. Academic Press, 1979.

NICHOLAS P. D. M. & ZHAO Y., “Association Rules: An Overview”. In: *Post-Mining of Association Rules: Techniques for Effective Knowledge Extraction*, Hershey, PA, USA, Zhao Y., Zhang C., Cao L. (coord.), IGI Global, 2009.

PAULOVICH F. V., OLIVEIRA M. C. F., MINGHIM R., “The Projection Explorer: A Flexible Tool for Projection-Based Multidimensional Visualization”. In: *Proceedings of XX Brazilian Symposium on Computer Graphics and Image Processing (SIBIGRAPI 2007)*, IEEE Computer Society Press, 2007.

REZENDE S. O., “Introdução”. In: *Sistemas Inteligentes: Fundamentos e Aplicações*, Barueri, SP, Brasil, Rezende S. O. (coord.), Editora Manole Ltda., Cap. 1, pp. 3-11, 2003.

REZENDE S. O., PUGLIESI, J. B., MELANDA E. A. & DE PAULA M. F., “Mineração de Dados”. In: *Sistemas Inteligentes: Fundamentos e Aplicações*, Barueri, SP, Brasil, Rezende S. O. (coord.), Editora Manole Ltda., Cap. 12, pp. 307-336, 2003.

SILVER D. L., “Knowledge Discovery and Data Mining”. MBA Course Notes of Dalhousie University, Nova Scotia, Canada, 1998.

ANEXO I
Imagens para o Ajuste da Hierarquização

São apresentadas a seguir as imagens das projeções multidimensionais geradas pelo sistema PEx, para os arquivos calculados pelo programa HierarqD contendo a matriz de distância *D* da base de dados Livro, variando o percentual de ajuste de hierarquização de forma decrescente, a partir de 100%, de 10 em 10.

A criação dessas imagens tem por objetivo identificar visualmente o melhor ajuste a ser usado na representação hierárquica dos itens, para efeito de análise das associações entre eles.

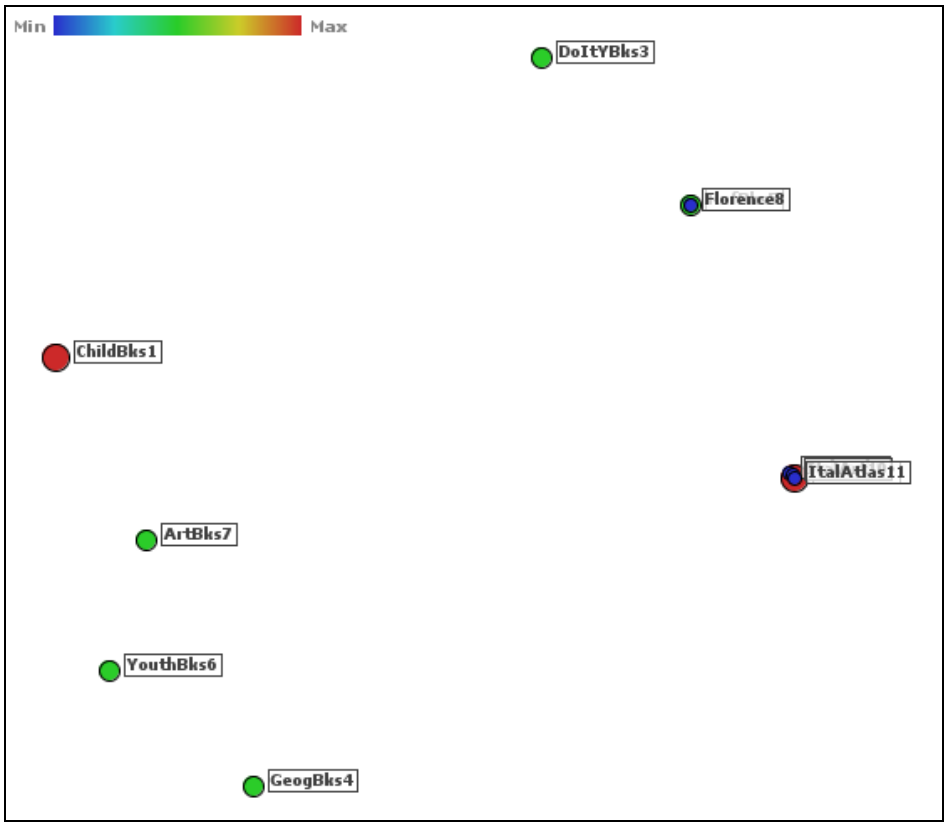


Figura I.1. Matriz *D* com Ajuste de Hierarquização de 100%.

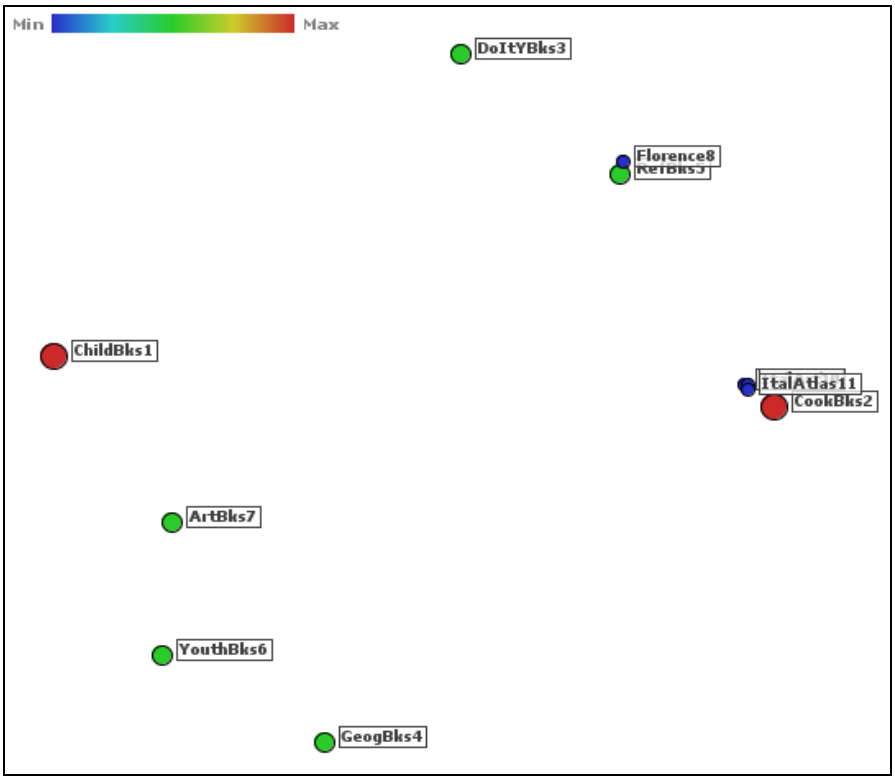


Figura I.2. Matriz *D* com Ajuste de Hierarquização de 90%.

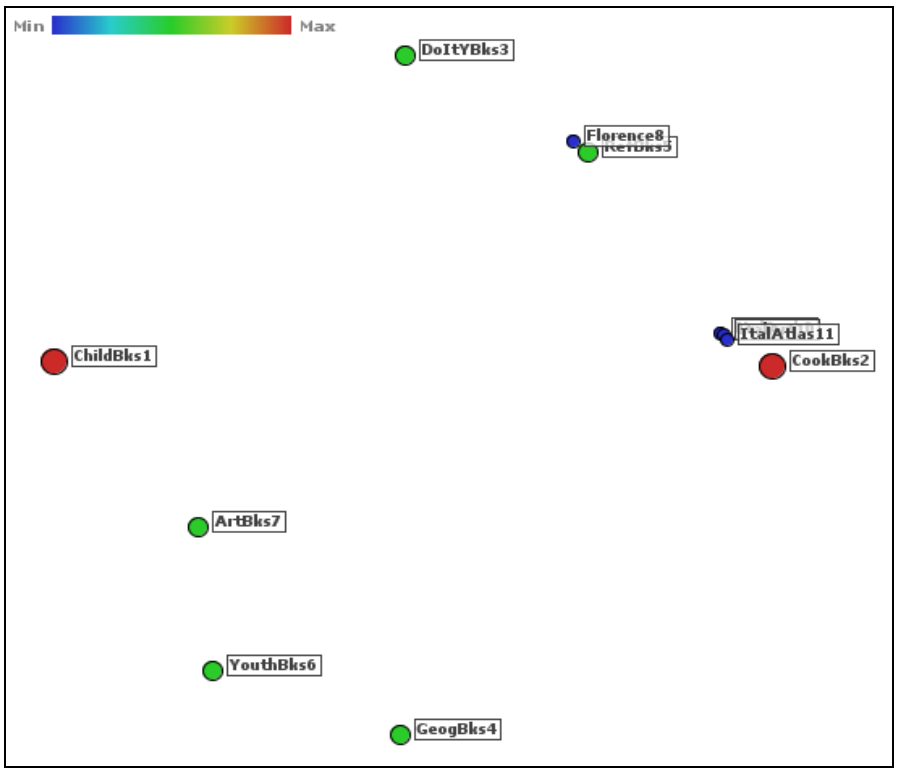


Figura I.3. Matriz *D* com Ajuste de Hierarquização de 80%.

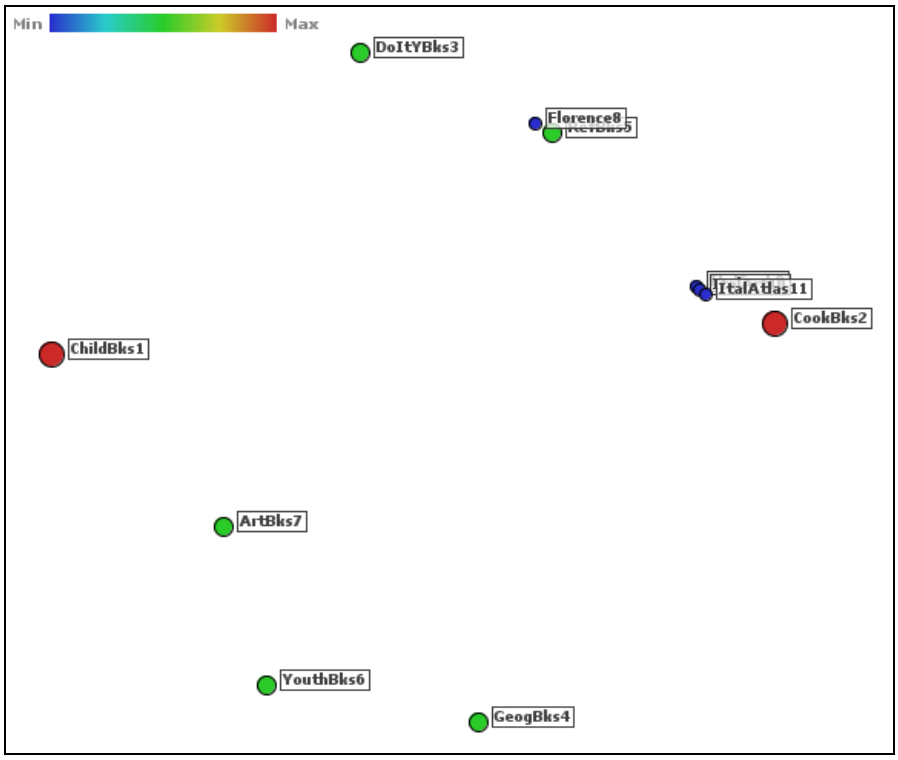


Figura I.4. Matriz *D* com Ajuste de Hierarquização de 70%.

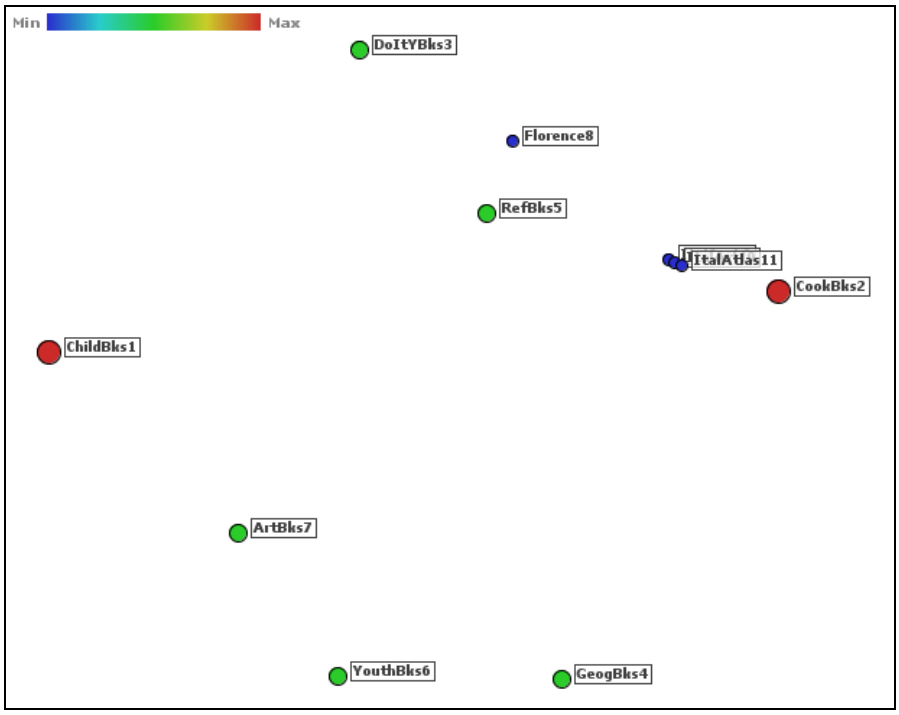


Figura I.5. Matriz *D* com Ajuste de Hierarquização de 60%.

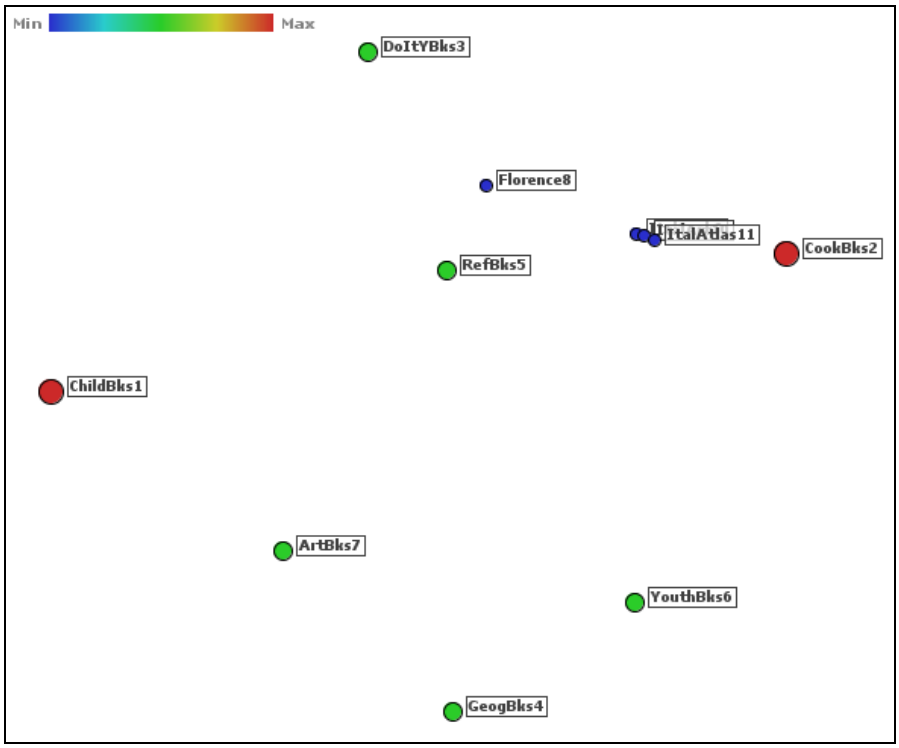


Figura I.6. Matriz *D* com Ajuste de Hierarquização de 50%.

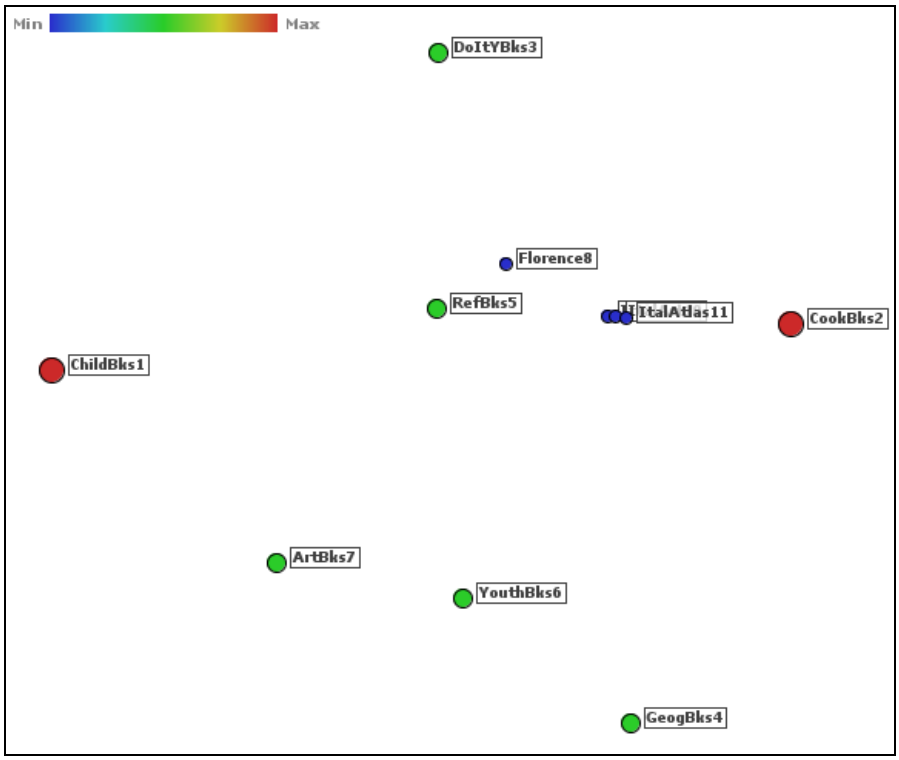


Figura I.7. Matriz *D* com Ajuste de Hierarquização de 40%.

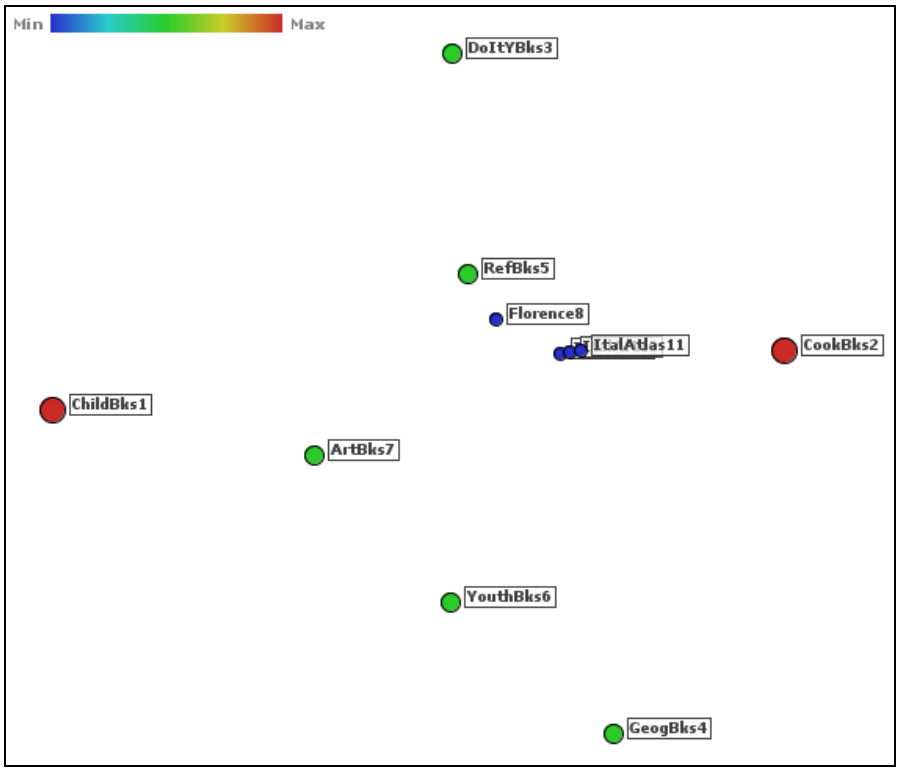


Figura I.8. Matriz *D* com Ajuste de Hierarquização de 30%.

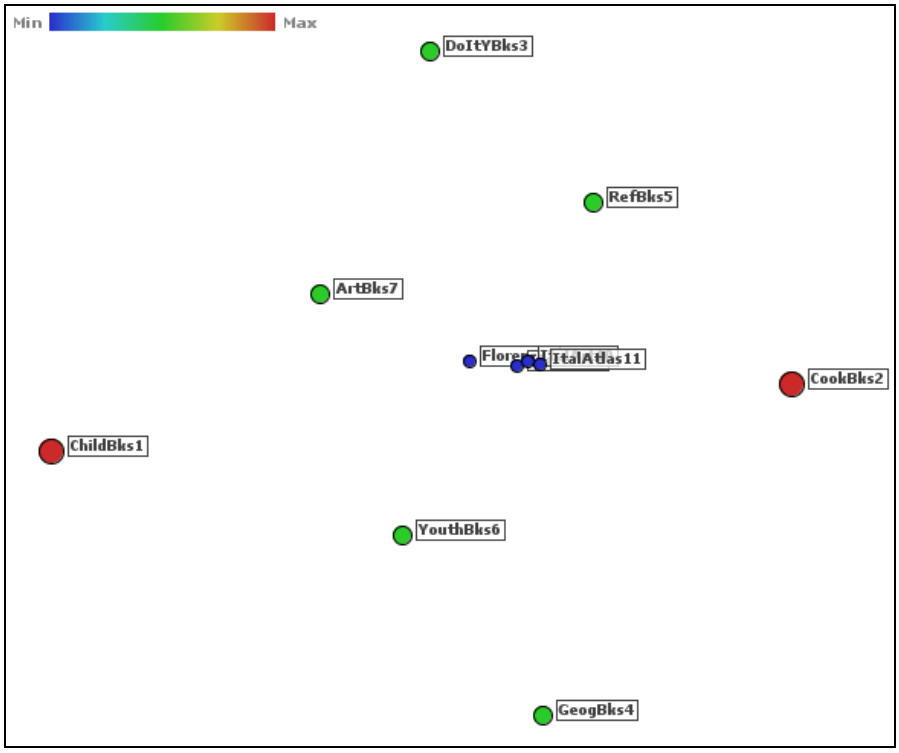


Figura I.9. Matriz *D* com Ajuste de Hierarquização de 20%.

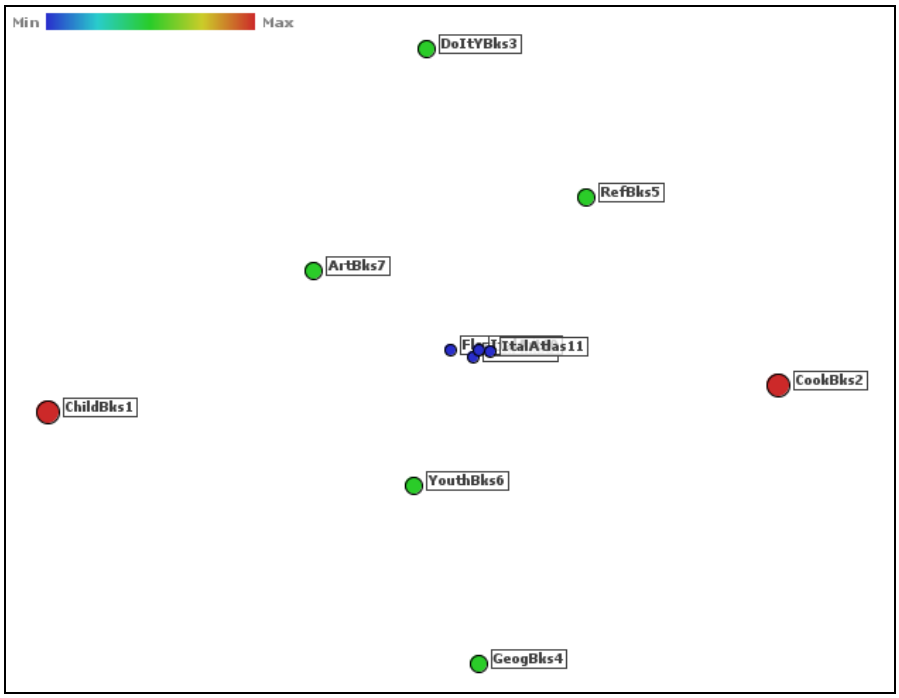


Figura I.10. Matriz *D* com Ajuste de Hierarquização de 10%.

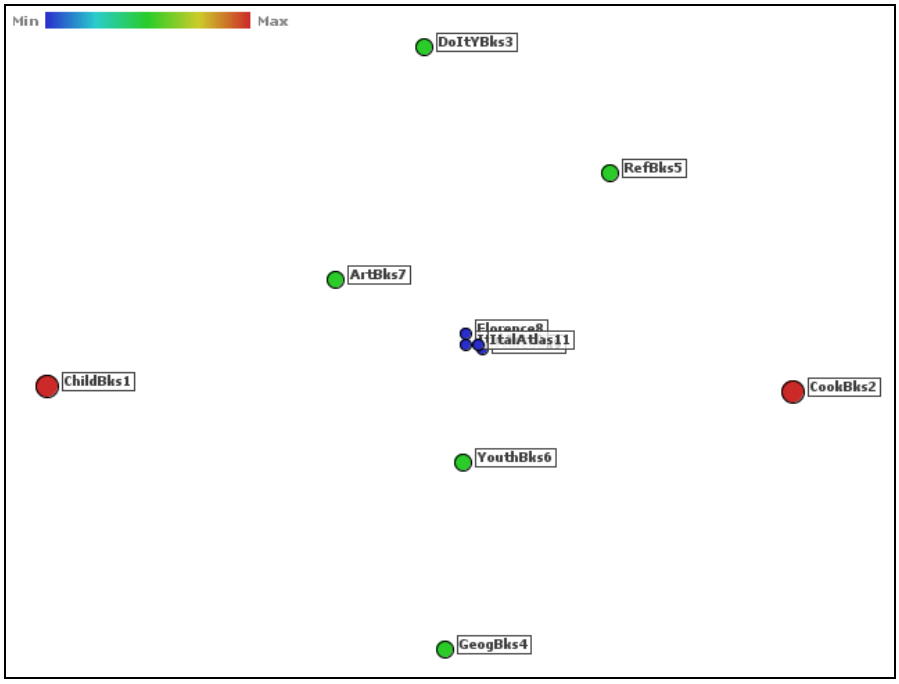


Figura I.11. Matriz *D* com Ajuste de Hierarquização de 0%.

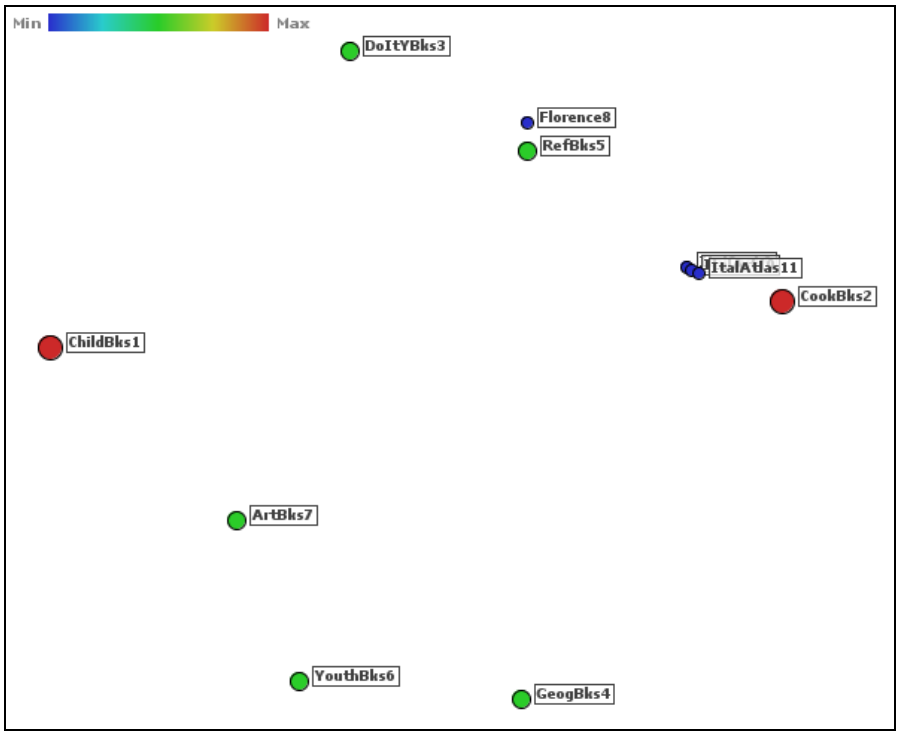


Figura I.12. Matriz *D* com Ajuste de Hierarquização de 65%.

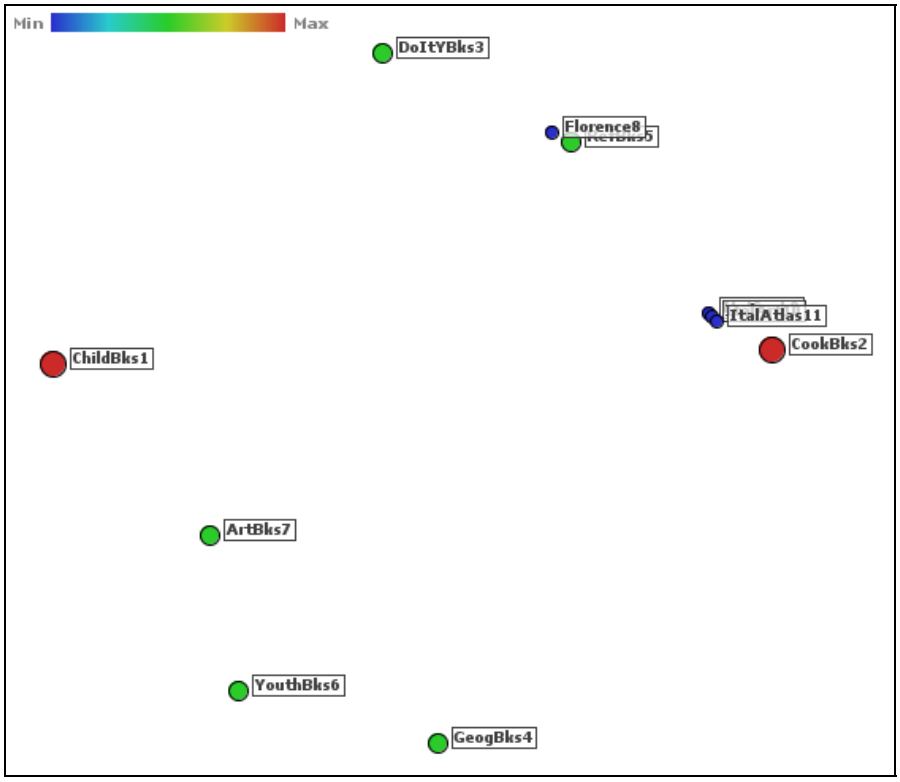


Figura I.13. Matriz *D* com Ajuste de Hierarquização de 75%.

ANEXO II
Imagens para o Ajuste da Otimização

As imagens a seguir, mostram os mapas gerados pelo sistema PEx, para arquivos calculados pelo programa HierarqD, contendo a matriz de distância *D* da base de dados Livro, mantendo a hierarquização em 65% e variando o percentual de ajuste de otimização em ordem crescente, de 10 em 10, a partir de 0%.

A criação dessas imagens tem por objetivo identificar visualmente o melhor ajuste para a otimização dos itens dentro de suas respectivas áreas de influência, para efeito de análise das associações entre eles.

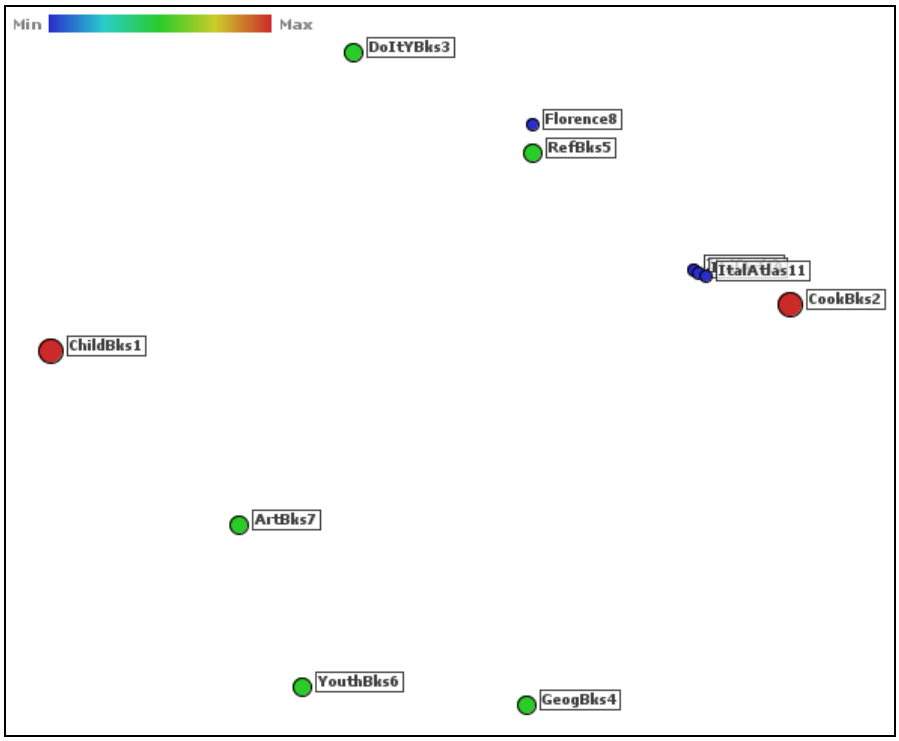


Figura II.1. Matriz *D* com 65% de Hierarquização e 0% de Otimização.

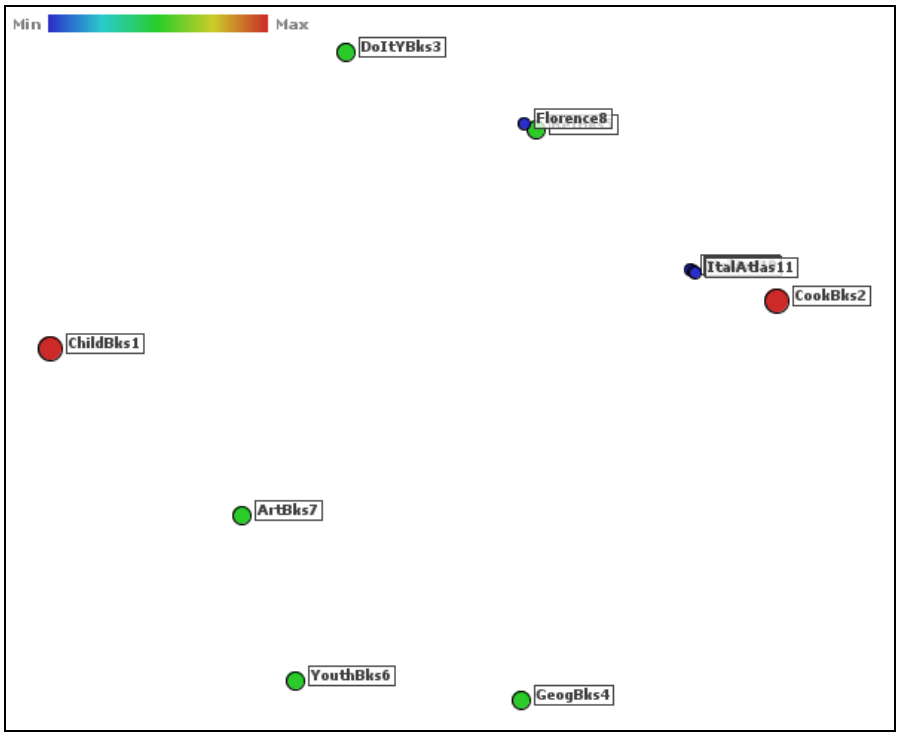


Figura II.2. Matriz *D* com 65% de Hierarquização e 10% de Otimização.

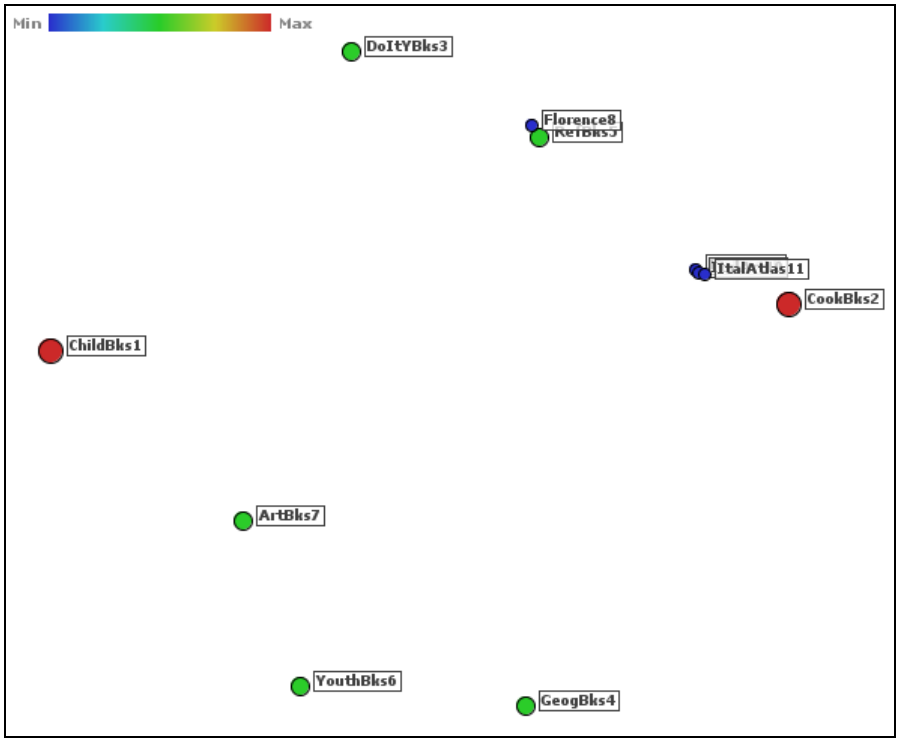


Figura II.3. Matriz *D* com 65% de Hierarquização e 20% de Otimização.

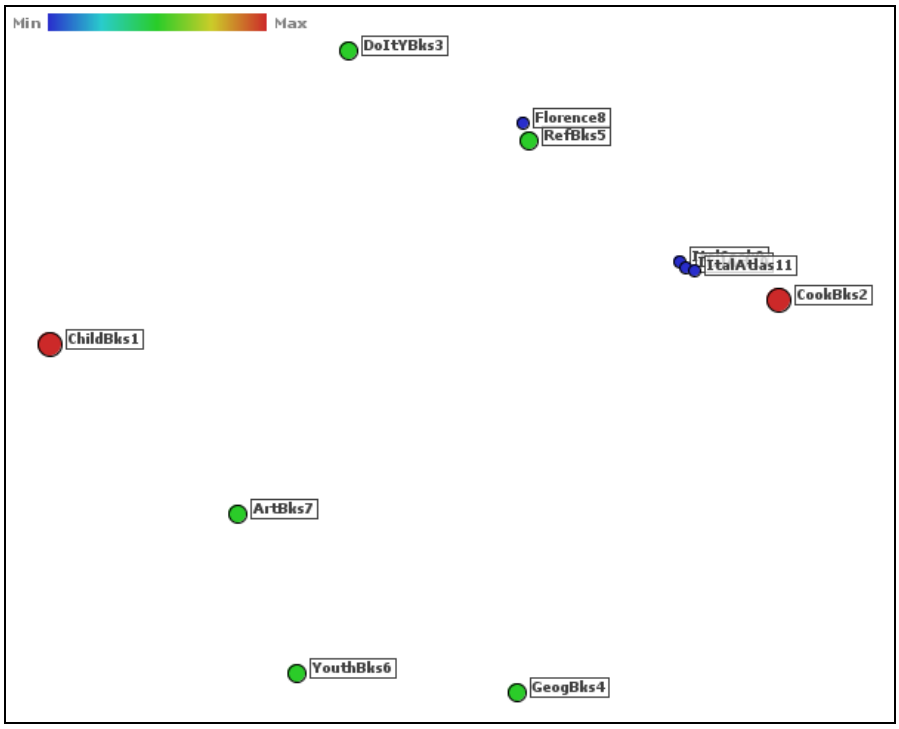


Figura II.4. Matriz *D* com 65% de Hierarquização e 30% de Otimização.

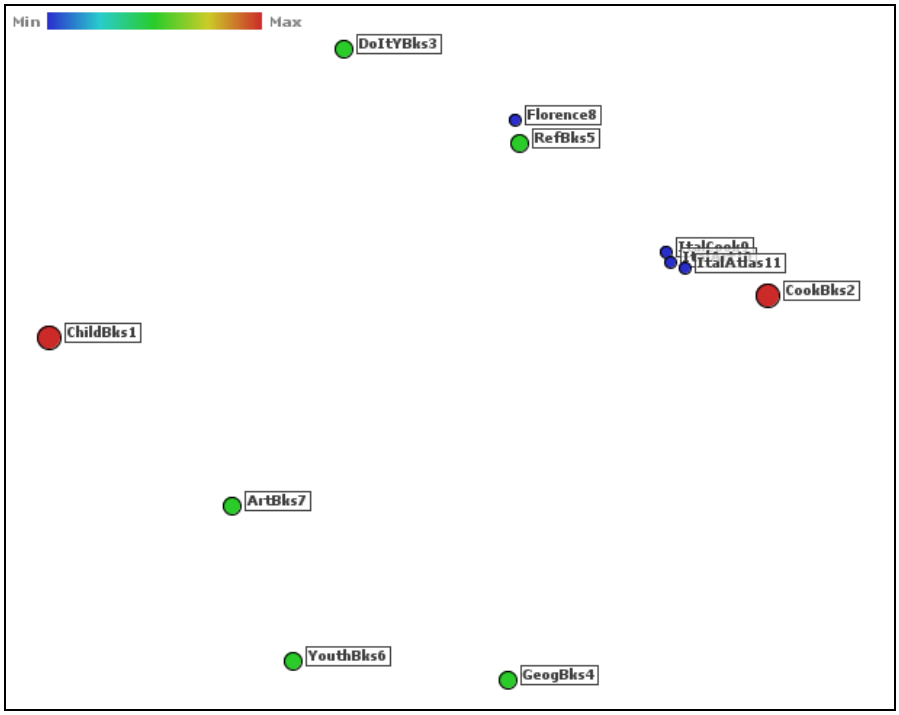


Figura II.5. Matriz *D* com 65% de Hierarquização e 40% de Otimização.

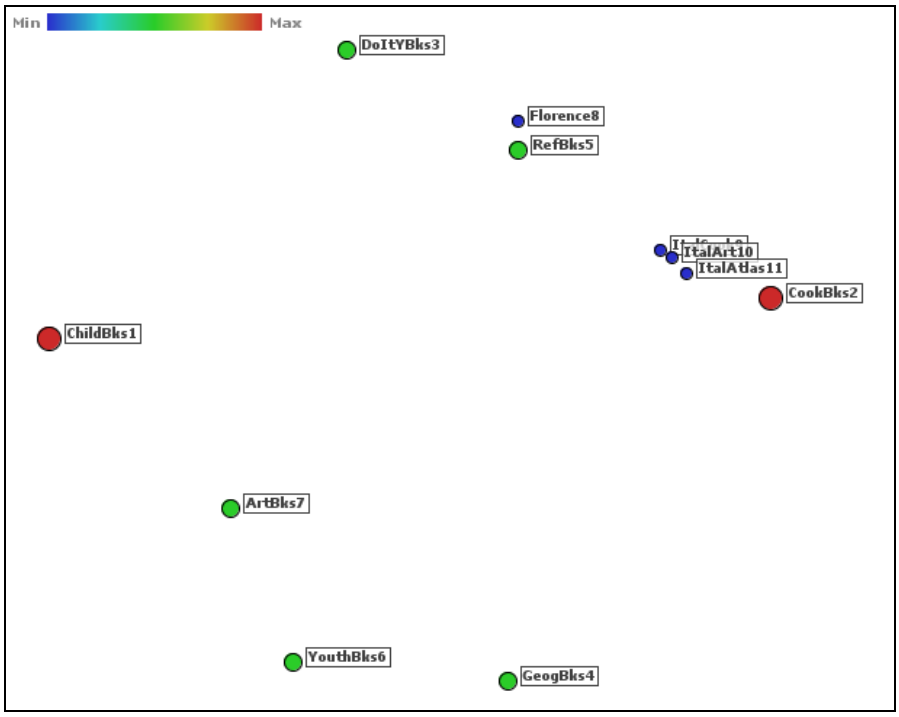


Figura II.6. Matriz *D* com 65% de Hierarquização e 50% de Otimização.

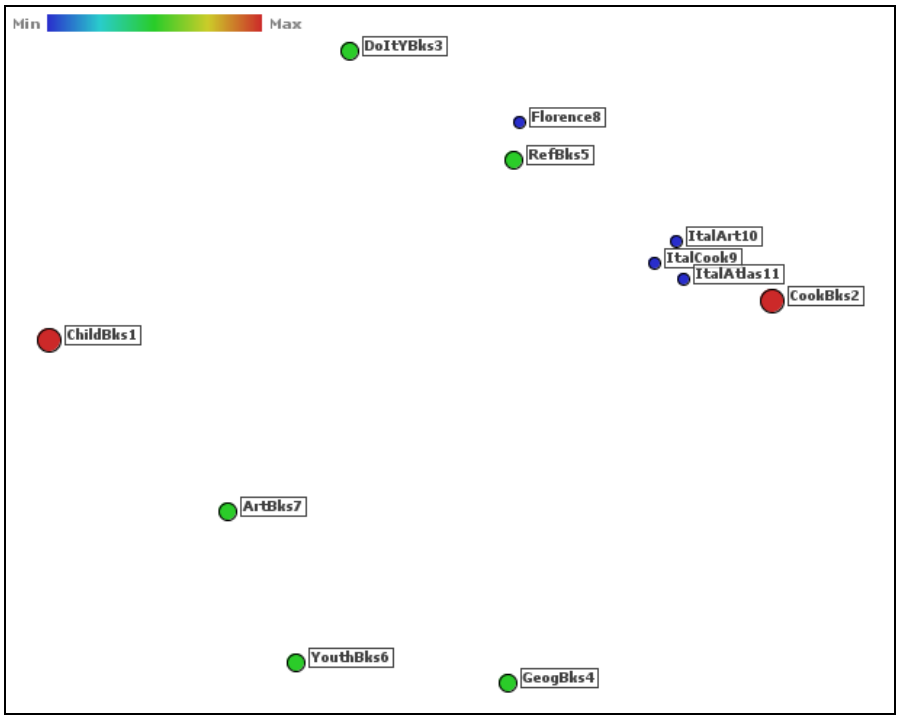


Figura II.7. Matriz *D* com 65% de Hierarquização e 60% de Otimização.

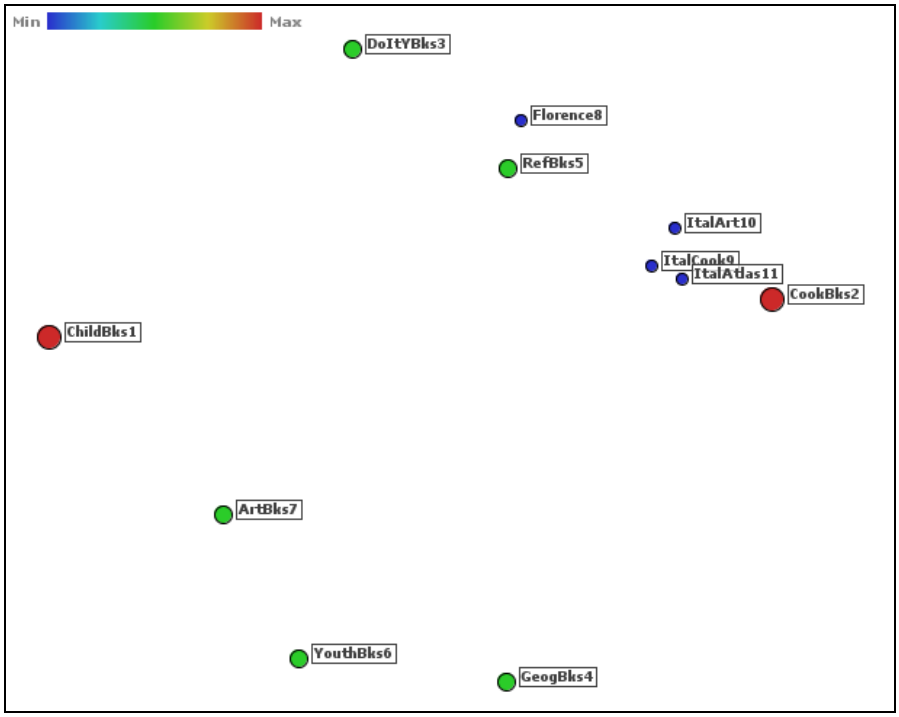


Figura II.8. Matriz *D* com 65% de Hierarquização e 70% de Otimização.

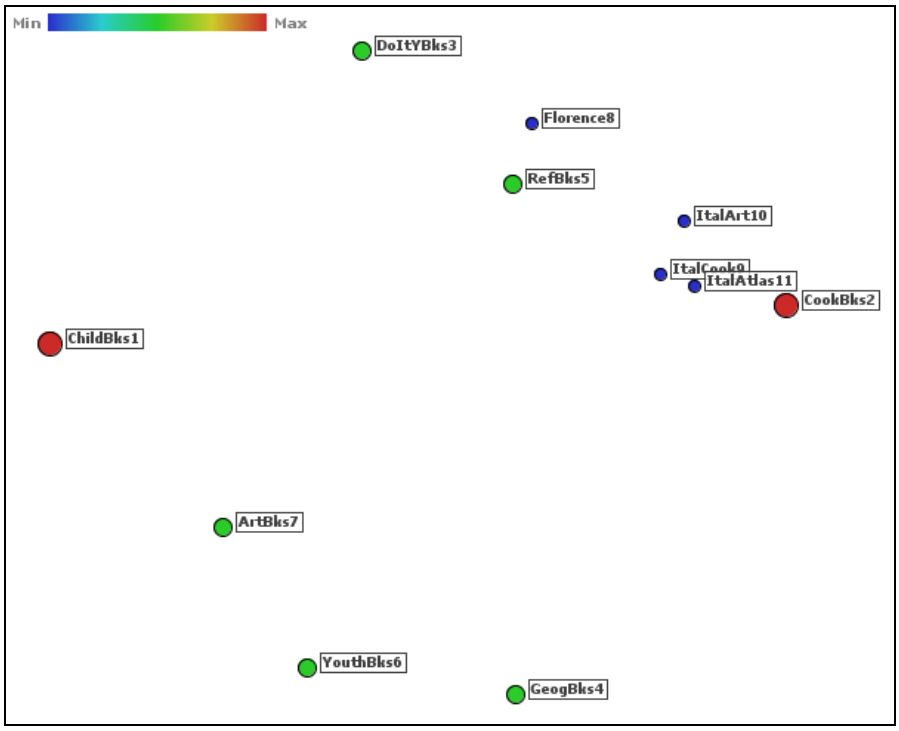


Figura II.9. Matriz *D* com 65% de Hierarquização e 80% de Otimização.

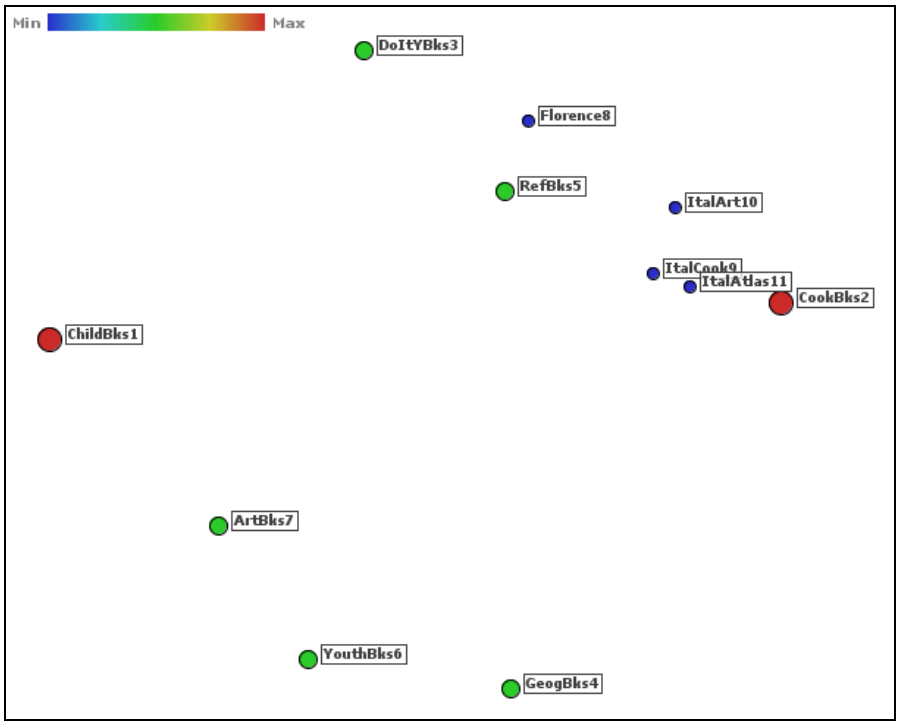


Figura II.10. Matriz *D* com 65% de Hierarquização e 90% de Otimização.

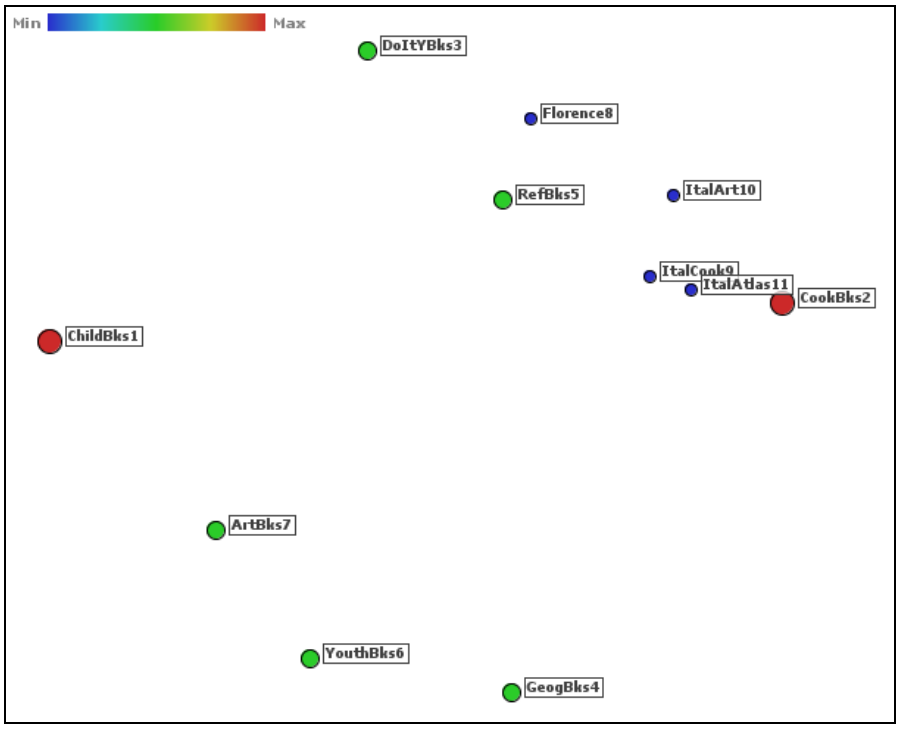


Figura II.11. Matriz *D* com 65% de Hierarquização e 100% de Otimização.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo