PROCEDIMENTO INTERATIVO DE AGRUPAMENTO DE DADOS

Lúcio Pereira de Andrade

TESE SUBMETIDA AO CORPO DOCENTE DA COORDENAÇÃO DOS PROGRAMAS DE PÓS-GRADUAÇÃO DE ENGENHARIA DA UNIVERSIDADE FEDERAL DO RIO DE JANEIRO COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE EM CIÊNCIAS EM ENGENHARIA CIVIL.

Aprovada por:

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

_____
Prof. Alexandre Gonçalves Evsukoff, D. Sc.

_____
Prof. Beatriz de Souza Leite Pires de Lima, D. Sc.

_____
Prof. Antônio Cesar Ferreira Guimarães, D. Sc.

RIO DE JANEIRO, RJ - BRASIL
DEZEMBRO DE 2004

Livros Grátis

http://www.livrosgratis.com.br

Milhares de livros grátis para download.

ANDRADE, LÚCIO PEREIRA DE

Procedimento Interativo de Agrupamento de dados [Rio de Janeiro] 2004

VIII, 193 p. 29,7 cm (COPPE/UFRJ, M.Sc., Engenharia Civil, 2004)

Tese - Universidade Federal do Rio de Janeiro, COPPE

1. Agrupamento de Dados

2. Visualização

I. COPPE/UFRJ II. Título (série)

DEDICATÓRIA

Ao meu filho, João Paulo.

Ao meu pai, Joaquim

AGRADECIMENTOS

A Deus, por ter me guiado e dado as oportunidades para que eu atingisse e completasse tão significativo passo.

Ao professor Nelson Ebecken, pela confiança, paciência e incentivo durante toda esta longa jornada.

Ao João Paulo, Ana Paula, Rosa e todos os meus familiares e amigos por terem entendido minha ausência durante o desenvolvimento deste trabalho.

À Maria Inês, José Flávio, Millan, Badin, Fábio Zappa, Silvana, Rogério e outros bons amigos que incentivaram, recomendaram e colaboraram para o início e conclusão do meu mestrado.

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

PROCEDIMENTO INTERATIVO DE AGRUPAMENTO DE DADOS

Lúcio Pereira de Andrade

Dezembro/2004

Orientador: Nelson Francisco Favilla Ebecken

Programa: Engenharia Civil

Este trabalho realiza a implementação de um procedimento interativo para agrupamento de dados amparado em estudos sobre métodos de segmentação, critérios de avaliação da qualidade da formação dos agrupamentos e técnicas de visualização.

A segmentação das bases de dados pode ser obtida pela aplicação de um método hierárquico aglomerativo, sendo visualizada na forma de dendrograma, ou através do método particional *"K-Means"*, cuja qualidade dos resultados, avaliada pelo critério estatístico de *Calinski* e *Harabasz* leva à busca interativa do número ideal de grupos.

A técnica de visualização "Coordenadas Estrela" torna possível a representação de objetos com n dimensões em um plano bidimensional, permitindo a observação de características espaciais das bases de dados.

Alguns estudos de caso, sobre três bases de dados, foram realizados para análise e comparação com outras metodologias e softwares comerciais existentes.

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

INTERACTIVE PROCEDURES FOR DATA CLUSTERING

Lúcio Pereira de Andrade

December/2004

Advisor: Nelson Francisco Favilla Ebecken

Department: Civil Engineering

The object ive of this work is the implementation of an interactive procedure for data clustering including the analysis of the usual methods and techniques of visualization.

The clustering analysis is performed by an aglomerative hierarchical method, being visualized in the form of dendrogram, or through the partitional "K-Means" method, using the Calinski and Harabasz statistical criterium to obtain ideal number of groups.

The visualization "Star Coordinates" technique is used to the object representation with n dimensions in a two-dimensional domain.

Several cases are studed to compare with another strategies available in commercial tools.

ÍNDICE

RESUMO v
ABSTRACT vi
CAPÍTULO 1 INTRODUÇÃO 1
1.1Objetivo 1
1.2 Importância 2
1.3 Aplicações 3
1.3.1 Ciências Sociais 3
1.3.2 Marketing 4
1.3.3 Ciências Médicas e Biológicas 4
1.3.4 Ciências Exatas e da Terra 5
1.3.5 Empresas e Governo 5
1.4 Outros capítulos 7
CAPÍTULO 2 AGRUPAMENTOS DE DADOS 8
2.1 Métodos de Agrupamento 8
2.1.1 Métodos particionais 12
2.1.2 Métodos hierárquicos 13
2.1.3 Métodos baseados em densidade 15
2.1.4 Métodos baseados em malhas 18
2.1.5 Métodos de modelos 21
2.2 O método *"K-Means"* 25
2.2.1 Medidas de distância 25
2.2.2 Descrição do processamento 27
2.3 O problema da definição do número de grupos 29
2.3.1 Método *Calinski* e *Harabasz* 30
2.3.2 Critério *Condorcet* 32
2.3.3 CCC – *Cubic Clustering Criterion* 34
2.3.4 Índice PBM 40
CAPÍTULO 3 MÉTODOS DE VISUALIZAÇÃO DE RESULTADOS 43
3.1 – Gráficos padronizados 44
3.1.1 – Gráfico de Linhas 45
3.1.2 – Gráfico de Dispersão 46
3.1.3 – Gráfico de Radar 47
3.1.4 – Gráfico de Colunas 48
3.2 – IBM Intelligent Miner for Data 49
3.3 – SAS *Enterprise Miner* 54
3.4 – *Mineset Enterprise Edition* 56
3.5 – *gCLUTO - Graphical CLUstering TOolkit* 60
3.6 – "Coordenadas Estrela" 63

3.7 – Dendrograma 66
CAPÍTULO 4 IMPLEMENTAÇÃO DE UM PROCEDIMENTO 69
4.1 Introdução 69
4.2 Arquitetura 70
4.3 Utilização 71
4.4 Código 81
CAPÍTULO 5 ESTUDOS DE CASOS 82
5.1 Primeiro caso – base de dados do DNA de cana de açúcar 82
5.1.1 Resultados do *SearchCluster* sobre a base de cana de açúcar 83
5.1.2 Resultados do *gCLUTO* sobre a base de cana de açúcar 86
5.1.3 Resultados do *Enterprise Miner* sobre a base de cana de açúcar 87
5.1.4 Resultados do *Mineset* sobre a base de cana de açúcar 89
5.1.5 Resultados do *Intelligent Miner* sobre a base de cana de açúcar 90
5.2 Segundo caso – base de dados de automóveis 93
5.2.1 Resultados do *SearchCluster* sobre a base de automóveis 94
5.2.2 Resultados do *gCLUTO* sobre a base de automóveis 102
5.2.3 Resultados do *Enterprise Miner* sobre a base de automóveis 106
5.2.4 Resultados do *Mineset* sobre a base de automóveis 111
5.2.5 Resultados do *Intelligent Miner* sobre a base de automóveis 113
5.3 Terceiro caso – base de dados de clientes "*churn*" 119
5.3.1 Resultados do *SearchCluster* sobre a base de clientes "*churn*" 120
5.3.2 Resultados do *gCLUTO* sobre a base de clientes "*churn*" 127
5.3.3 Resultados do *Enterprise Miner* sobre a base de clientes "*churn*" 129
5.3.4 Resultados do *Mineset* sobre a base de clientes "*churn*" 134
5.3.5 Resultados do *Intelligent Miner* sobre a base de clientes "*churn*" 135
CAPÍTULO 6 CONCLUSÃO 140
REFERÊNCIAS BIBLIOGRÁFICAS 144
APÊNDICE I LISTAGEM DO PROGRAMA SEARCHCLUSTER 147

CAPÍTULO 1
INTRODUÇÃO

1.1Objetivo

O objetivo deste trabalho foi desenvolver uma ferramenta computacional e mostrar resultados práticos da implementação de técnicas para agrupamento de dados e sua visualização.

O conjunto inicial de n objetos é dividido em um número previamente desconhecido e ideal de grupos, determinado através de sucessivos arranjos desses objetos em grupos, de maneira que o centro de cada grupo guarde a máxima distância possível em relação aos demais e os seus objetos tenham a mínima distância entre si, utilizando-se para formação dos arranjos a técnica conhecida como *"K-Means"* [1] e para o cálculo da distância entre os objetos a medida Euclidiana.

A visualização de objetos com diversos atributos, n-dimensões, em um plano de duas dimensões, obtida através do emprego da técnica "Coordenadas Estrela" [2], possibilita a percepção virtual da distribuição espacial relativa dos objetos, a forma dos grupos e a posição de seus centros. A representação da classificação hierárquica é feita através de dendrogramas, método onde os objetos ou grupos são representados graficamente com uma estrutura em árvore, semelhantemente a uma classificação taxonômica. No desenvolvimento desta representação, foi utilizado o método conhecido como Vizinho Mais Próximo (ou Ligação Simples) [3].

As técnicas exploradas neste trabalho foram implementadas em um aplicativo ao qual foi dado o nome de *SearchCluster*. Desenvolvido com a linguagem de programação *Microsoft® Visual Basic 6.0*, o aplicativo possui uma interface amigável, que permite pesquisar e analisar bases de dados com diversas configurações em formato texto, de maneira interativa.

1.2 Importância

Desde os primórdios de sua organização social, o homem cultiva uma tendência natural de agrupar objetos segundo alguma combinação de atributos que representem interesse no momento. Nas primitivas tribos de homens das cavernas, onde os grupos eram separados segundo padrões físicos, mulheres coletavam e homens caçavam. Crianças separam brinquedos em grupos, inicialmente pela cor e tamanho, depois de acordo com características com um pouco mais de refinamento, como em um grupo os carrinhos, noutro os bonecos de animais. Pessoas adultas colecionam diversos objetos: chaveiros, bonés, latas de cerveja, selos, automóveis. Grupos de objetos muito similares entre si, mas completamente diversos, se comparados com os representantes de outras coleções.

Os objetos do mundo real normalmente possuem uma grande quantidade de atributos que, por suas semelhanças, formam classes distintas. Porém, mesmo dentro dessas classes pode ser crucial a formação de subgrupos, o que pode ser de extrema dificuldade, face a quantidade de objetos, o número de seus atributos, a sutileza das diferenças apresentadas e da escassez de tempo.

Um especialista pode ter o sentimento à cerca do número de grupos em que se dividem os objetos de uma coleção e até ser capaz de identificar muitos objetos que, dentro de um grupo, tenham mais ou menos proximidade entre si para formar aqueles grupos. O problema é dispor do tempo desse especialista na faina exaustiva de separação dos objetos, ao invés de utilizá-lo mais nobremente na tarefa de analisar os grupos formados, interpretar o conhecimento descoberto durante o processamento.

E, após a separação dos objetos em grupos, como perceber rapidamente padrões, linhas gerais, áreas mais promissoras, em meio aos ainda muitos algarismos que representam a informação? Novamente, torna-se possível poupar tempo ao analista apresentando uma visualização gráfica dos dados, sua forma de agrupamento e a hierarquia formada por eles.

Com o advento e a popularização da tecnologia de bancos de dados nas últimas décadas, o volume de informações disponíveis é cada vez maior, atingindo volumes que inviabilizam a sua análise direta por falta de recursos técnicos e tempo para observação de cada unidade da informação. A separação de um elevado número de objetos em grupos que guardem significativa semelhança facilita a percepção de

padrões, descoberta de conhecimento e manipulação desses e outros objetos que venham juntar-se à população inicialmente observada.

As técnicas de agrupamento, que não são comprobatórias e, sim, exploratórias por natureza, objetivam justamente realizar uma separação ótima entre os objetos de uma coleção, permitindo a descoberta de novos padrões, previamente desconhecidos e insuspeitos. O resultado da segmentação, independentemente da ferramenta utilizada, somente pode ser interpretado eficientemente por um especialista na área de origem dos dados sob análise.

Apesar dessas técnicas não atingirem a determinação dos grupos com total precisão, é de consenso, com diversas demonstrações bem sucedidas, que grupos bem formados são aqueles que apresentam a mínima distância entre os seus objetos e a máxima distância para os demais grupos.

1.3 Aplicações

As técnicas de segmentação e visualização de dados são aplicáveis, e muito úteis, em diversas áreas do conhecimento humano onde a quantidade dos dados a serem tratados seja elevada, o número de seus atributos seja considerável, haja necessidade de proceder a vários e sucessivos arranjos para melhor análise das possibilidades ou até diversas combinações entre tais aspectos.

Podem ser citados vários casos que explicitam claramente a grande diversidade de áreas onde há o real emprego e sempre novas possibilidades de aplicação das técnicas de análise de agrupamento. Alguns exemplos interessantes, relacionados a seguir, são encontrados na página oficial da *Clustan Ltd* [4], citados como áreas de aplicações bem sucedidas do *software clustan.*

1.3.1 Ciências Sociais

Demografia: através do estudo do agrupamento de registros de nascimentos, é possível deduzir as conseqüências sociais e genéticas de padrões migratórios ancestrais.

Arqueologia: a divisão dos utensílios e inscrições achados em escavações arqueológicas em grupos distintos pode explicar as suas origens culturais.

Psicologia educacional: a modelagem das realizações acadêmicas do universo de estudantes auxilia no melhoramento de grupos de estudantes com maiores dificuldades.

Geografia: os grupos de fazendas permitem traçar a distribuição de atividades agrícolas diferentes e mudanças de medida de produtividade.

Lingüística: agrupamento de idiomas e variedades de fala, auxilia no entendimento da sua evolução e no desenvolvimento da taxonomia da semântica.

Sociologia: com atitudes sociais, comportamentos e papéis dos indivíduos auxiliam no desenvolvimento da tipificação de classes sociais.

1.3.2 Marketing

A análise de agrupamento é de importante utilidade no ramo do marketing, podendo ter diversas aplicações, tais como: segmentação de clientes e mercados, *design* de produtos e marcas, especificação de alvos de mala-direta e *telemarketing*, conversão e retenção de clientes, diminuição dos custos de *marketing*.

1.3.3 Ciências Médicas e Biológicas

Antropologia: o agrupamento de dados antropométricos revela diferenças genéticas entre gêneros humanos ancestrais e ajudam na reconstrução da evolução da espécie.

Bioquímica: o estudo da segmentação de dados da composição de aminoácidos e genes ajuda a explicar as sucessões evolutivas de mutações que originam uma nova espécie.

Microbiologia: os estudos de grupos de bactérias propiciaram a melhoria da taxonomia existente, a identificação do aparecimento de novos vírus e desenvolvimento de vacinas.

Psicologia clínica: o agrupamento de padrões de comportamentos sociais extremos tornou possível identificar síndromes psicológicas associadas ao autismo.

Botânica: mostras de vegetação devidamente agrupadas facilitam a descrição da ecologia de comunidades de plantas nativas e indicam áreas para desenvolvimento agrícola ou conservação e métodos para prevenção de erosão de terrenos.

Psiquiatria: o estudo de sintomas psicopatológicos, separados em grupos similares, ajuda na identificação de síndromes de desordens psiquiátricas e no desenvolvimento dos tratamentos.

1.3.4 Ciências Exatas e da Terra

Astronomia: as técnicas de análise de agrupamento aplicadas em estrelas e galáxias ajudam no desenvolvimento de teorias sobre as origens e evolução de sistemas estelares.

Química: o agrupamento de estruturas químicas auxilia no entendimento de suas propriedades biológicas, físicas e químicas.

Informática: projetistas de bancos de dados têm realizado agrupamento de dados em elementos de sistema de informações para aperfeiçoar o desempenho das operações de recuperação da informação.

Engenharia: diversas aplicações de reconhecimento de padrão baseadas em lógica de análise de agrupamento para automação na identificação de características em imagens de satélite, impressões digitais, sinais de radar e outras aplicações.

Física: as técnicas de agrupamento de dados aplicadas em fotografias de câmara de bolha agilizam a classificação das trilhas feitas por diferentes partículas carregadas.

Meteorologia: o agrupamento de padrões de tempo permite descobrir tipos comuns de seqüências para previsões de longo alcance.

1.3.5 Empresas e Governo

Contabilidade e Administração: a segmentação de bases de dados de contas de empresas permite identificar tipos de negócios com capitalização comum,

desempenho e aproveitamento de perfis para uso em reestruturação e troca de programas de administração.

Bancos e administradoras de cartões de crédito: agrupar os dados das contas de clientes de suas gigantescas bases de dados, de acordo com perfis socio-econômicos e financeiros, permite melhorar a rentabilidade das instituições e, analisando as características de uso, descobrir transações incomuns, indicadores de prováveis ações delituosas.

Além destes casos, existem diversos relatos sobre as mais diversas aplicações de técnicas de agrupamento em casos concretos. Entre eles podemos citar:

- prospecção dos dados de acesso a um servidor de notícias na Web, utilizada na construção de um sistema de publicação personalizada, baseada nos perfis dos leitores [5];

- técnicas de c*luster analysis* na construção de tipologias de curva de carga, empregadas com a finalidade de identificar a estrutura natural de agrupamentos em um conjunto de medições de curvas de carga, associando cada grupo formado a uma tipologia [6];

- inteligência competitiva para o ensino superior privado: uma abordagem através da mineração de textos. As técnicas de agrupamento permitem identificar e agrupar documentos que possuam informações em comum, proporcionando uma certa organização a uma série de documentos dispostos de maneira desorganizada [7];

- regulação da qualidade de suprimento por comparação e impacto na receita permitida no setor elétrico, através da formação de conjuntos com características de atendimento semelhantes, tornando mais justas as metas estabelecidas após comparação com outras empresas com características similares [8];

- potencial de desenvolvimento dos municípios fluminenses: uma metodologia alternativa ao Índice de Qualidade dos Municípios - IQM, tornando possível a obtenção de uma fórmula mais equilibrada, de interpretação mais fácil e menos sensível a erros [9];

- padrões espaciais de mortalidade e diferenciais sócio-econômicos na cidade do Rio de Janeiro, através da constituição de grupos sócio-economicamente

homogêneos para análise da distribuição dos indicadores de mortalidade nos diferentes estratos [10];

- aplicação de dados de detecção remota em ecologia: com a classificação de imagens de satélite foi possível otimizar a aplicação de recursos humanos e logísticos para fundamentar a escolha dos locais de amostragem, através da identificação de zonas homogêneas [10].

Um caso de particular relevância do presente trabalho é a implementação de um procedimento com capacidade de agrupar os objetos de forma pertinente, mesmo perante a ausência do conhecimento antecipado do número de grupos que compõem toda a coleção de dados em estudo. A necessidade do prévio conhecimento do número ideal de grupos onde devem ser divididos e alocados os objetos de uma base é um fator bastante limitador para os diversos algoritmos de segmentação que dependem desse parâmetro para sua execução, como é o caso da aplicação clássica da técnica “K-Means”.

1.4 Outros capítulos

A presente tese contém 6 capítulos e estão referenciados como se segue: Capítulo 1 – Introdução; Capítulo 2 – Agrupamento de dados: discorre sobre trabalhos desenvolvidos e aplicações das técnicas de agrupamento de dados; Capítulo 3 – Métodos de Visualização de Resultados: apresenta as técnicas utilizadas no desenvolvimento de um módulo para visualização em duas dimensões de objetos n-dimensionais, incluindo a demonstração dos resultados obtidos; Capítulo 4 – Implementação de um Procedimento: apresenta as fases de desenvolvimento de um software e os métodos empregados para descoberta e agrupamento de dados em número previamente desconhecido de grupos; 5 – Estudos de Casos, apresenta uma análise sobre diversos casos de implementação e emprego prático de técnicas de agrupamento e visualização de dados; 6 – Conclusão, encerra o presente trabalho com algumas considerações finais sobre o estudo e propostas para continuação da implementação.

CAPÍTULO 2
AGRUPAMENTOS DE DADOS

2.1 Métodos de Agrupamento

A descoberta do conhecimento em base de dados (KDD, sigla em inglês) é o processo não trivial de identificação de padrões que sejam válidos, novos, potencialmente úteis e compreensíveis. O processo de KDD foi proposto em 1989 para referir-se às etapas que produzem conhecimentos a partir dos dados e, principalmente, à etapa de mineração dos dados (*data mining*), que é a fase que transforma dados em informações [12].

De acordo com esta definição de KDD, o termo “processo não trivial” indica que existem vários passos, desde a seleção, a preparação dos dados e a procura de conhecimento, até à sua avaliação (tudo repetido em várias iterações cada vez mais refinadas e interativamente com o usuário). Os padrões implícitos nos dados devem ser válidos para novos dados com algum grau de certeza, devem constituir algum fato novo e prestar-se à tomada de decisões úteis. Finalmente, o conhecimento descoberto deve ser representado de forma compreensível e adequada ao tipo de usuário a que se destina [5].

O processo de KDD envolve vários passos distintos e complementares, como pode ser visto na Figura 2.1. Dentro da prospecção de dados as tarefas mais comuns são a classificação, o agrupamento (*clustering*), a modelagem de dependências, a análise de ligações e a análise de seqüências.

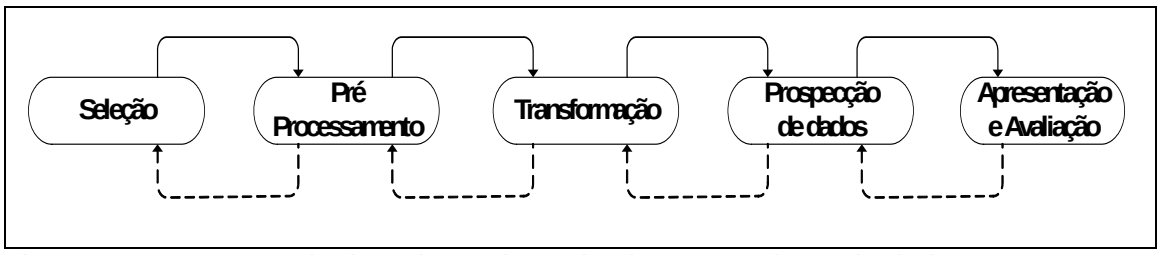


Figura 2.1: Os passos da descoberta do conhecimento em base de dados.

De forma instintiva os seres humanos tendem a visualizar conjuntos em grupos discretos. É uma das formas mais naturais e rápidas de impor alguma ordem,

algum padrão inteligível, a um grande volume de objetos apresentado sem maiores informações que levem a um entendimento direto por parte do observador. Porém, perceber padrões e agrupar objetos tendo por origem uma grande coleção não é tarefa simples, ou que possa ser feita manualmente em tempo hábil. São necessários algoritmos de segmentação, especializados em descoberta e formação de grupos de dados.

Segmentação é o processo de agrupamento dos dados em classes ou grupos, de maneira que objetos dentro de um mesmo grupo tenham alta similaridade, mas objetos pertencentes a grupos diferentes sejam muito distintos. Para a mensuração de tais similaridades (ou dissimilaridades) entre os objetos da coleção analisada são utilizadas diversas técnicas de medida, normalmente baseadas em medidas de distância que, inclusive, devem ser adaptadas para dados especiais, como por exemplo variáveis binárias, nominais e ordinais [1].

É extremamente útil quando não se tem conhecimento prévio sobre o conjunto de dados em questão é necessário separá-lo em grupos. A segmentação geralmente encontra-se associada com alguma técnica de descrição de conceitos a fim de identificar os atributos de cada classe.

As ferramentas de agrupamento de dados podem ser empregadas nas mais diversas áreas do conhecimento humano e, até mesmo por essa diversidade, de acordo com HAN e KAMBER [1], devem atender alguns requisitos específicos, tais como:

Escalabilidade: geralmente os algoritmos de segmentação trabalham bem em grupos de dados pequenos, cerca de 200 objetos. Entretanto, uma grande base de dados pode conter milhões de objetos. Assim, tais algoritmos necessitam ser altamente escaláveis.

Habilidade em lidar com tipos diferentes de atributos: normalmente, os algoritmos de segmentação desenvolvidos para agrupar dados numéricos. Contudo, algumas aplicações podem requer a segmentação de outros tipos de dados, tais como binários, categóricos (nominais), ordinais ou misturas destes tipos de dados.

Descoberta de grupos com formas arbitrárias: os algoritmos de segmentação são muitas vezes baseados em distâncias Euclidianas ou *Manhattan*. Tais algoritmos tendem a achar grupos “esféricos”, com tamanho e densidade similares. Contudo, um

grupo pode assumir qualquer forma. É importante desenvolver algoritmos que possam descobrir grupos que possuam formas arbitrárias.

Necessidade mínima de conhecimento do domínio para determinação dos parâmetros de entrada: em sua maioria, os algoritmos de segmentação requerem dos usuários certos parâmetros para análise de agrupamento, tais como o número desejado de grupos. Os algoritmos, assim, ficam muito sensíveis à influência de parâmetros que muitas vezes são difíceis de determinar, principalmente, em grupos de objetos com um elevado número de atributos.

Habilidade em lidar com dados “ruidosos”: muitos bancos de dados do mundo real contém dados fora do padrão ou perdidos, desconhecidos ou errôneos. Alguns algoritmos de segmentação são sensíveis a tais dados e podem conduzir a agrupamentos de baixa qualidade.

Insensibilidade para a ordem de entrada dos registros: alguns algoritmos de segmentação podem gerar dramáticas diferenças entre os grupos formados, dependendo da ordem que os dados lhes são apresentados.

Alta dimensionalidade: um banco de dados ou *data warehouse* pode conter várias dimensões - atributos. Muitos algoritmos de segmentação são bons para manuseio de objetos com baixa dimensionalidade, envolvendo somente dois a três dimensões. A capacidade humana permite julgar a qualidade de segmentos de até três dimensões. O desafio está em agrupar objetos de dados representados por um elevado número de atributos – alta dimensionalidade, principalmente considerando-se que tais dados podem ser muito esparsos e altamente assimétricos.

Segmentação baseada em restrições: aplicações do mundo real podem necessitar segmentações que obedeçam a algumas restrições. Um bom algoritmo de segmentação deve ser capaz de descobrir os grupos ideais que atendam a essas restrições.

Interpretabilidade e usabilidade: deve ser esperado que o resultado da segmentação seja explicável, compreensível e utilizável.

A segmentação, ou *clustering*, é a tarefa central para a qual muitos algoritmos têm sido propostos. Regularmente variantes novas de métodos mais velhos, ou novas abordagens emergem ao mesmo tempo em que as atividades humanas demandam o estabelecimento de padrões para comparação de tais algoritmos [13]. Existe uma

grande diversidade de algoritmos de segmentação devido à grande variedade de princípios de indução e modelos. E os princípios indutivos existem em tão grande número porque “segmentação é em parte no olho do observador”, sendo formalizações matemáticas do que os pesquisadores acreditam ser uma definição de agrupamento. O conhecimento do domínio é que forma a crença de que existem subgrupos em meio a um grande volume de objetos e tal convicção é que molda as estruturas utilizadas para representar os grupos. Assim, diferenças nos princípios indutivos são de ordem “filosófica”, mas fundamentais. Contudo, de acordo com ESTIVILL-CASTRO [13], alguns mecanismos podem ser usados na comparação de princípios indutivos e, conseqüentemente, de algoritmos de segmentação, resumidos a seguir:

- os métodos de segmentação são oriundos da percepção do observador;

- diferentes pesquisadores podem determinar boas segmentações por diferentes fórmulas matemáticas;

- a segmentação traduz em otimização problemas cuja complexidade computacional é tipicamente intratável e são solucionados por algoritmos de aproximação;

- o primeiro nível de comparação entre dois algoritmos de segmentação dá-se em termos da qualidade da solução, como o valor obtido pela mesma função objetivo e recursos de computação iguais (por exemplo, número de avaliações da função objetivo);

- existem objetivos funcionais (princípios indutivos) que apresentam custo menor que outros. Naturalmente, busca-se aperfeiçoar aqueles na esperança de apresentarem boas soluções com custo menor. Contudo, isto implica uma investigação sobre o relacionamento entre os princípios indutivos e, num segundo estágio, entre os próprios algoritmos;

- pesquisas e compilações de categorias de algoritmos de segmentação são naturalmente baseadas mais em modelos do que em princípios de indução. A mais forte distinção entre tais algoritmos está na adoção de modelos matemáticos (contínuos), comuns em inferências estatísticas, ou modelos de estruturas (discreto), comuns em aprendizado de máquina;

- os pesquisadores que propõem modelos e princípio indutivo novos ou algoritmos de segmentação melhorados deveriam explicitar seus métodos matemáticos, o que facilitaria investigações e comparações prévias com métodos emergentes;

- os índices de validade da segmentação são formulações matemáticas diretas de princípios de indução. A comparação entre os algoritmos pode fornecer alguns *insights* sobre o contexto no qual um algoritmo processa melhor que outro. Contudo, isto não implica que um algoritmo produz resultados mais válidos que outro. A validade depende da existência de uma estrutura do conjunto de dados;

- a qualidade da segmentação pode ser demonstrada por critérios externos de validade e medidas associadas. Métodos de aprendizado supervisionado podem ser usados para avaliação de algoritmos, como medidas de discrepância entre os resultados da segmentação e os rótulos previamente conhecidos;

- um algoritmo adaptado a um determinado universo de modelos não tem chance de bom desempenho se os grupos de dados têm uma estrutura que é, na verdade, representada por uma família radicalmente diferente (por exemplo, “K-Means” não pode determinar grupos não-convexos).

São inúmeras as citações sobre algoritmos de segmentação que podem ser encontradas na literatura. A escolha do algoritmo mais adequado depende tanto do tipo dos dados disponíveis quanto do propósito específico da aplicação.

De maneira geral, a maioria dos métodos de agrupamento podem ser classificados em cinco grandes categorias: métodos particionais, métodos hierárquicos, métodos baseados em densidade, métodos baseados em malhas e métodos baseados em modelos. A seguir, cada uma das categorias citadas será apresentada com mais detalhes, conforme apresentado por HAN e CAMBER [1].

2.1.1 Métodos particionais

Considerando-se que n é o número de objetos de uma base de dados e k é o número de grupos desejado, os algoritmos particionais irão gerar k partições utilizando medidas de distância, de forma que uma determinada função objetivo seja otimizada, sendo que objetos comparados com os demais objetos dentro de sua própria partição

guardem máxima similaridade e objetos comparados com os objetos de outras partições apresentem mínima similaridade.

Os métodos particionais clássicos mais conhecidos, e comumente utilizados, são *"K-Means"* e "*k-Medoids"*. O método *"K-Means"* é baseado no conceito estatístico do centróide de forma que a semelhança dos objetos dentro do mesmo grupo, "intragrupo", seja alta e a semelhança entre objetos de grupos diferentes, "intergrupo", seja baixa, sendo, porém, muito sensível a dados fora do padrão, ou seja, objetos que tenham um valor extremamente alto ou baixo, que podem causar substanciais distorções nos resultados. O método *"K-Means"* será visto em mais detalhes mais adiante, neste capítulo. Em contrapartida, o método "*K-Medoids"* busca diminuir a sensibilidade a objetos isolados, deixando de levar em conta o centro do grupo como ponto de referência para utilizar o medóide, que é o objeto localizado mais próximo ao centro do grupo. Assim, este método pode abrandar a influência de objetos isolados, baseando-se no princípio de minimização da dessemelhança entre os objetos de um grupo e o seu ponto de referência.

A estratégia básica do algoritmo de segmentação "*K-Medoids"* é achar k grupos em n objetos. Primeiramente, arbitrando para cada grupo um objeto como seu representante (o medóide). Cada objeto remanescente é agrupado com o medóide ao qual possui a maior semelhança. Então, iterativamente, é substituído um medóide por um não-medóide, desde que a qualidade da segmentação resultante seja melhor. Esta qualidade é calculada usando uma função de custo que calcula a medida de dessemelhança média entre um objeto e o medóide de seu grupo.

Numa comparação entre os dois métodos particionais clássicos, o método "*K-Medoids"* pode ser apontado como mais robusto que *"K-Means"* por trabalhar com mais eficiência na presença de dados isolados e outros ruídos, em decorrência do medóide sofrer menos influência de dados desse tipo. Contudo, seu processamento tem custo maior que o método *"K-Means"*. Mas, sempre vale ressaltar, ambos os métodos requerem a prévia especificação de k, o número de grupos.

2.1.2 Métodos hierárquicos

Um método hierárquico de segmentação trabalha agrupando objetos em uma árvore de grupos. A qualidade dos métodos hierárquicos puros pode ser prejudicada pela sua inabilidade em executar ajustes após uma fusão ou divisão terem sido

executadas. Os métodos hierárquicos podem ser classificados de acordo com a forma com que a decomposição hierárquica é realizada: *bottom-up* ou *top-down*, e são conhecidos como:

Segmentação hierárquica aglomerativa: esta estratégia *bottom-up* inicia por considerar cada objeto como sendo seu próprio grupo e, então, vai criando sucessivas fusões destes grupos atômicos em conjuntos maiores, até que todos os objetos estejam em um único grupo ou até que certas condições de conclusão sejam satisfeitas. Os métodos hierárquicos, em sua maioria, pertencem a esta categoria, diferindo entre si apenas pela definição da semelhança “intergrupo”.

Segmentação hierárquica divisiva: esta estratégia *top-down* faz o contrário da segmentação hierárquica aglomerativa, começando com todos objetos num único grupo. A partir daí os grupos são subdivididos em conjuntos menores, até que cada objeto forme o próprio grupo; até que o algoritmo satisfaça determinadas condições de conclusão, tais como a obtenção de um número desejado de grupos ou que a distância entre dois grupos próximo ultrapasse um certo limiar.

A Figura 2.2 apresenta, de maneira esquemática, os passos da segmentação aglomerativa e divisiva.

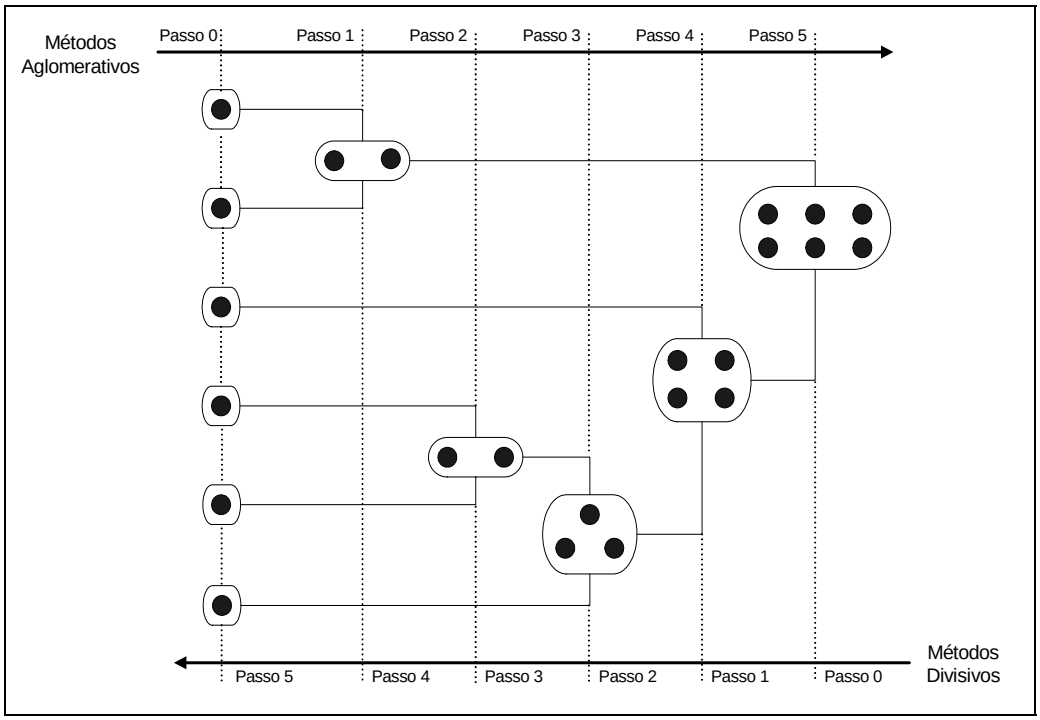


Figura 2.2: Esquema dos métodos de segmentação aglomerativos e divisivos. Baseada em [1].

Os métodos de segmentação hierárquicos, embora simples, freqüentemente encontram dificuldades no que diz respeito à seleção de pontos de fusão ou de divisão. Tal decisão é crítica, pois uma vez que um grupo de objetos é fundido ou dividido, o próximo passo será processado com base no grupo recém gerado. Considerando-se que tais algoritmos não desfazem o que já foi processado, tampouco trocam objetos entre grupos, se as fusões ou divisões não são bem feitas em algum passo a qualidade do resultado final da segmentação pode ficar seriamente comprometida. Além disso, o método não possui critérios para examinar e avaliar se a fusão ou divisão redunda em um número ideal de objetos ou grupos.

Um promissor campo de ação para melhorar a qualidade da segmentação dos métodos hierárquicos é a integração destes métodos com outras técnicas de segmentação, tais como métodos iterativos de realocação.

2.1.3 Métodos baseados em densidade

Devido à falta de habilidade que os algoritmos baseados em métodos particionais têm em descobrir grupos com formas arbitrárias, isto é, grupos que apresentam objetos dispersos em uma distribuição espacial bastante irregular, foram desenvolvidos métodos de segmentação baseados em densidade. Estes métodos consideram tipicamente os grupos como regiões densas de objetos separados no espaço por regiões de baixa densidade – que, normalmente, representam algum tipo de ruído.

Alguns algoritmos de implementação dos métodos de segmentação baseados em densidade podem ser especificamente citados, como abaixo:

DBSCAN (*Density-Based Spatial Clustering of Application with Noise*): é um algoritmo baseado em conexões de regiões com suficientemente alta densidade que emprega a estratégia de estabelecer tais regiões como grupos. O DBSCAN tem a capacidade de descobrir grupos com formas arbitrárias em espaços de dados, inclusive com presença de ruídos, definindo os grupos como a combinação máxima de pontos densamente conectados.

Para realizar a segmentação, o DBSCAN verifica a vizinhança próxima de cada objeto dentro do banco de dados. Se a vizinhança de um objeto p contém ao menos o número mínimo de ocorrências, então p será considerado como o núcleo (ou centro) de um novo grupo a ser criado em seu entorno, chamado de vizinhança

próxima. O raio de definição da vizinhança - ε , bem como o número mínimo de ocorrências, MinPts, são parâmetros pré-definidos. A partir daí o algoritmo irá agrupar, iterativamente, os objetos com “densidade de alcance” direta em relação ao núcleo dos grupos, ou seja, são grupados os objetos mais próximos ao núcleo de maneira que o grupo mantenha alta densidade. Este processo pode envolver a fusão de grupos com pouca densidade e termina quando nenhum ponto novo pode ser incluído em qualquer grupo. Todo objeto não contido em qualquer um dos grupos formados será considerado como ruído.

OPTICS (*Ordering Points To Identify the Clustering Structure*): método que emprega a identificação da estrutura dos grupos a partir da ordenação de seus objetos, OPTICS apresenta uma importante vantagem em relação ao método apresentado anteriormente. Como visto, o algoritmo DBSCAN depende da introdução de parâmetros, como o raio de definição da vizinhança e o número mínimo de ocorrências para realizar a segmentação de objetos, o que deixa com o usuário a responsabilidade de definir valores de entrada adequados para a formação de agrupamentos satisfatórios. Na verdade, este é um problema associado a muitos outros algoritmos de segmentação. Tais parâmetros são usualmente difíceis de ponderar e determinados de maneira empírica, principalmente para dados do mundo real com muitas dimensões. Naturalmente, muitos algoritmos são bastante sensíveis aos valores de parâmetros. Por exemplo, combinações de parâmetros de entrada apenas levemente diferentes podem conduzir a formação de agrupamentos muito discrepantes em relação aos dados analisados. Além disso, coleções de dados reais com muitas dimensões têm freqüentemente uma inclinação de distribuição induzida por sua estrutura intrínseca, que não pode ser caracterizada por parâmetros de densidade globais.

O algoritmo OPTICS foi desenvolvido justamente procurando superar a fragilidade causada pela dependência dos parâmetros de entrada. Ao invés de formar agrupamentos explícitos, este algoritmo faz o cômputo da “ordenação aumentada de agrupamentos” para análise interativa e automática. Esta ordenação representa a estrutura dos dados agrupados por densidade e contém informações que são equivalentes às obtidas em um largo espectro de parâmetros de entrada.

No algoritmo DBSCAN, a constante MinPts diz respeito ao grau de densidade dos grupos formados e ε é um parâmetro de distância que representa o raio da vizinhança mais próxima ao núcleo. Portanto, a fim de criar uma ordenação

aumentada de agrupamentos, pode-se estender o DBSCAN para processar um conjunto de parâmetros de distância ao mesmo tempo. Para construir diferentes segmentações simultaneamente, os objetos devem ser processados em uma ordem específica. Esta ordem seleciona os objetos dentro da densidade de alcance da vizinhança com menor raio, de forma que os grupos com maior densidade sejam completados primeiro. Com base nesta idéia, dois valores precisam ser armazenados para cada objeto:

- a distância de núcleo é o mínimo valor de ε que faz de p um objeto núcleo. Se p não é um objeto núcleo, a distância de núcleo fica indefinida;

- a distância de alcance de um objeto q em relação a outro objeto p é definida como o maior valor entre a distância de núcleo de p e a distância Euclidiana entre p e q. Se p não é um objeto núcleo, a distância de alcance fica indefinida.

Em seu processamento, o algoritmo OPTICS cria uma ordenação dos objetos, armazenando adicionalmente a distância de núcleo e a distância de alcance mais satisfatória para cada objeto, que serão posteriormente usadas para extrair os grupos baseados em densidade.

DENCLUE (*DENsity-based CLUstEring*): método baseado em um conjunto de funções de distribuição de densidade, constituído das seguintes idéias:

1. a influência de cada objeto de uma base de dados pode ser formalmente modelada usando uma função matemática, chamada função de influência, a qual descreve o impacto de um ponto de dados dentro de sua vizinhança;

2. a densidade global do espaço de dados pode ser modelada analiticamente como a soma das funções de influência de todos os objetos;

3. os grupos podem ser determinados matematicamente identificando-se a densidade de atração, que é o máximo local da função de densidade global.

Em comparação com outros algoritmos de segmentação, existem algumas significativas vantagens apresentadas pelo DENCLUE: forte fundamentação matemática e generalização de outros métodos de segmentação; boa capacidade de segmentação em coleções de dados com elevada presença de ruídos; possibilidade de descrição matemática para formas arbitrárias de grupos de dados com muitas dimensões; utilização de malhas que contêm somente informações sobre a estrutura

das células, sendo assim significativamente mais rápidos que alguns algoritmos mais influentes como o DBSCAN.

2.1.4 Métodos baseados em malhas

A abordagem utilizada pelos métodos baseados em malhas utiliza-se de uma malha de estrutura de dados com múltipla resolução. Dessa forma, o espaço de estudo é quantificado em um número finito de células que formam uma malha sobre a qual todas as operações de segmentação serão realizadas. A vantagem principal desta abordagem é seu rápido processamento, pois não é diretamente dependente do número de objetos a serem analisados e sim do número de células em cada dimensão do espaço quantificado.

Alguns exemplos típicos dos métodos baseados em malhas são o STING, que explora informações estatísticas armazenadas nas células da malhas; *WAVECLUSTER*, que agrupa os objetos usando o método de transformada ondaleta [14] e CLIQUE, que apresenta uma combinação entre métodos baseados em malhas e baseados em densidade para segmentação de bases de dados com alto número de dimensões. Estes exemplos são melhor explicados a seguir.

STING (*STatistical INformation Grid*): técnica baseada em malhas de múltipla resolução, na qual o espaço a ser analisado é dividido em células retangulares. Normalmente, existem diversos níveis de células, correspondendo a níveis diferentes de resolução que formam uma estrutura hierárquica: cada célula de um nível qualquer é particionada em um certo número de células que formam o nível imediatamente inferior. Informações estatísticas com respeito aos atributos em cada célula, como média, máximo e mínimo, são computadas e armazenadas.

As informações estatísticas das células de mais alto nível podem ser obtidas através das células do nível mais baixo. Estas informações incluem: um atributo independente, *count*, e atributos dependentes m (média), s (desvio padrão), min (mínimo), max (máximo) e o tipo de distribuição que os valores na célula obedecem: normal, uniforme, exponencial ou nenhum (se a distribuição é desconhecida). Estes parâmetros são calculados diretamente a partir dos dados carregados. O tipo de distribuição pode ser fornecido interativamente, caso seja previamente conhecido,ou obtido por um critério de hipótese de teste.

As informações estatísticas obtidas são utilizadas em um processo de pergunta-resposta da seguinte forma: primeiramente, é determinada a camada que contém a estrutura hierárquica a partir da qual o processo de pergunta-resposta será iniciado. Esta camada tipicamente contém um número pequeno de células. Para cada célula da camada atual, é computado o intervalo de confiança (ou alcance da estimativa de probabilidade), refletindo a relevância da célula em questão. As células irrelevantes são desconsideradas e o processo do nível imediatamente abaixo examina unicamente as células que permaneceram pertinentes. Este processo é repetido até que a camada mais inferior seja atingida

Como o STING utiliza uma aproximação de resolução múltipla para executar a análise de agrupamento, a qualidade da segmentação depende diretamente da granularidade do nível mais inferior da estrutura da malha. Se a granularidade é muito fina, o custo do processo aumenta substancialmente. Por outro lado se é muito grossa, pode reduzir a qualidade dos grupos formados. Além disso, STING não considera o espaço de relacionamento entre células filhas e sua vizinhança para a construção de uma célula pai. Como resultado, a forma do agrupamento resultante tem contorno bem definido apenas em uma dimensão, vertical ou horizontal, e nenhum marco diagonal pode ser detectado, o que pode diminuir a qualidade e a acuidade dos grupos.

WAVECLUSTER: um algoritmo de segmentação de múltipla resolução que gera um resumo dos dados em um malha multidimensional, usando transformadas ondaleta para transformar as características do espaço original, descobrindo regiões densas no espaço transformado.

Transformada ondaleta é uma técnica de processamento de sinais que decompõe um sinal em diferentes sub-bandas de freqüência. O modelo ondaleta pode ser aplicado para sinais com n dimensões, aplicando-se a transformada em cada dimensão n vezes. A aplicação da transformada ondaleta conserva a distância relativa entre os objetos em diferentes níveis de resolução. Isto permite que os grupos naturais tornem-se mais distintos, podendo ser identificados como regiões densas no novo domínio.

A transformada ondaleta tem boa aplicabilidade na segmentação de dados, devido a algumas de suas vantagens:

- é um método não-supervisionado de segmentação. O uso de filtros “*hat-shape*” consegue enfatizar as regiões de maior concentração de pontos, enquanto suprime informações mais fracas. Assim, regiões densas no espaço exercem uma

forte atração dos pontos vizinhos e como inibidores dos pontos mais distantes. Dessa forma, outra vantagem da transformada ondaleta é a sua capacidade de remover automaticamente pontos isolados;

- a propriedade de múltipla resolução pode colaborar diretamente na descoberta de grupos com diversos níveis de variação de acuidade;

- a segmentação com transformadas ondaleta é muito rápida, permitindo, ainda, implementação de algoritmos com processamento paralelo.

O *WAVECLUSTER* é um algoritmo baseado em malhas e em densidade, que atende muitos requerimentos de um bom algoritmo de segmentação: manipula grandes coleções de dados eficazmente; descobre grupos com formas arbitrárias; processa dados fora do padrão com sucesso; é insensível para a ordem de introdução dos dados e não exige a especificação de parâmetros como o número desejado de grupos ou raio de vizinhança.

CLIQUE (*CLustering In QUEst*): algoritmo de segmentação que integra métodos de malha e densidade. É de grande utilidade para segmentar dados de muitas dimensões em banco de dados extensos, baseado nas seguintes características:

- em uma grande base de dados multidimensionais, os objetos normalmente não se distribuem uniformemente pelo espaço. A segmentação de CLIQUE identifica as áreas esparsas e as áreas densas (unidades), determinando assim os padrões de distribuição globais da base de dados.

- uma unidade é considerada densa se a fração de objetos nela contidos excederam um parâmetro fornecido. Um grupo é definido como a combinação máxima de unidades densas conectadas.

O método CLIQUE realiza a segmentação multidimensional em dois passos: no primeiro passo, é particionado o espaço de cada dimensão dos dados em unidades retangulares não sobrepostas, identificando unidades densas; no segundo passo, é gerada uma descrição mínima para cada grupo, determinando a região máxima coberta pelas unidades densas conectadas para, então, ser determinada a cobertura mínima de cada grupo.

A efetividade do CLIQUE pode ser avaliada por algumas de suas principais características: capacidade de encontrar automaticamente sub-espaços de alta

densidade e os grupos existentes neles; não é sensível à ordem de introdução dos objetos e apresenta boa escalabilidade. Contudo, a acuidade do resultado da segmentação pode ser degradada devido à simplicidade do método.

2.1.5 Métodos de modelos

Os métodos baseados em modelos criam um modelo hipotético para cada grupo desejado e procuram ajustar os dados da melhor maneira ao modelo criado. Os algoritmos baseados neste método são capazes de descobrir os grupos através de funções de densidade que refletem a distribuição espacial dos objetos. Também possibilitam modos de determinar automaticamente o número de grupos baseados em padrões estatísticos, inclusive em presença de ruídos ou dados fora do padrão, o que proporciona o desenvolvimento de algoritmos bastante robustos. Como tentativa de otimização dos métodos de segmentação, com uma melhor adequação entre os dados apresentados e alguns modelos matemáticos, tais métodos são usualmente baseados na hipótese de que os dados são originados de acordo com uma probabilidade estatística de distribuição. Os métodos de modelos seguem duas principais abordagens: a abordagem estatística ou a abordagem de redes neurais.

A abordagem estatística é um tipo de segmentação que utiliza aprendizado de máquina, também conhecida como segmentação conceitual. Dada uma coleção de objetos sem rótulo, este método produz um esquema de classificação que vai além dos dados. Ao contrário de métodos de segmentação convencionais, os quais identificam basicamente grupos de objetos afins, a segmentação conceitual descobre também descrições das características de cada grupo, onde os grupos representam um conceito ou classe. Assim, é um processo feito em dois passos: primeiro, é realizada a segmentação; depois, a caracterização. Neste método, a avaliação da qualidade da segmentação não é somente uma função dos objetos individuais, ela passa por fatores como a generalidade e a simplicidade derivadas da descrição dos conceitos.

Como exemplo desta abordagem, podemos citar o COBWEB, que é um método popular e simples de segmentação conceitual. Nele, os objetos introduzidos são descritos por pares de atributos categóricos e a segmentação é hierárquica, em forma de uma árvore de classificação, como pode ser visto na Figura 2.3.

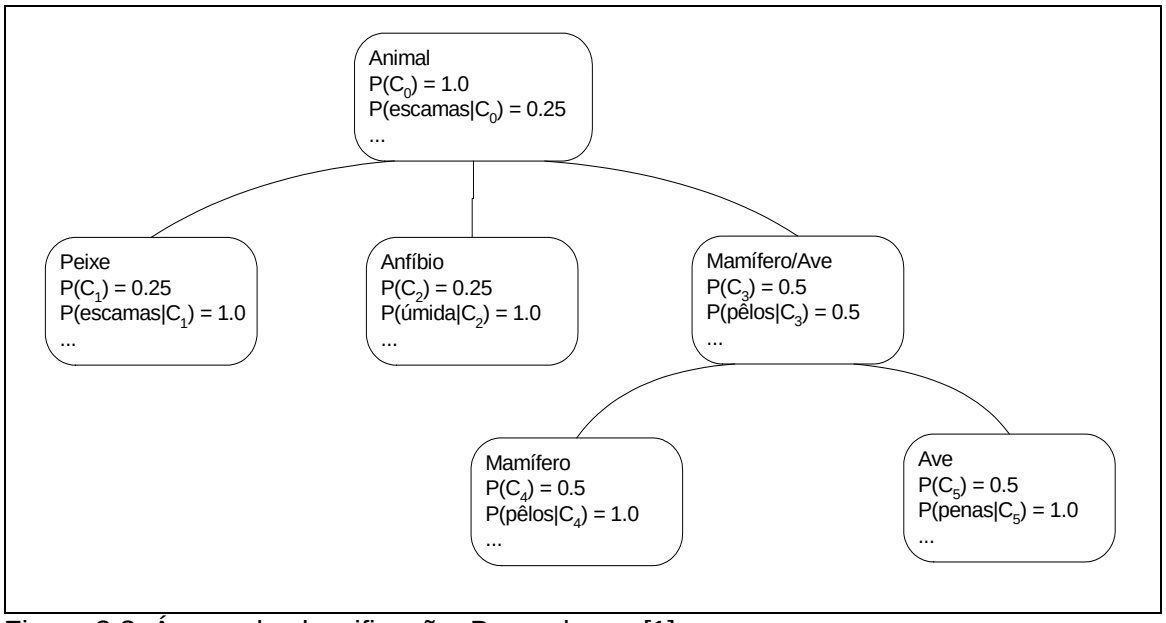


Figura 2.3: Árvore de classificação. Baseada em [1].

Árvore de classificação é diferente de árvore de decisão. Cada nó numa árvore de classificação se refere a um conceito e contém a descrição probabilística que resume os objetos classificados sob o nó. A descrição probabilística inclui a probabilidade do conceito e as probabilidades condicionais na forma $P(A_i = V_{ij}|C_k)$, onde $Ai = V_{ij}$ é um par de atributos e C_k é o conceito da classe. Contadores são acumulados e armazenados para cada nó a fim de realizar o cálculo das probabilidades. Os nós de um mesmo nível de uma árvore de classificação formam uma partição. Para classificar um objeto usando uma árvore de classificação, a estratégia é descer a árvore ao longo do caminho de melhor adaptação dos nós. COBWEB usa uma medida de avaliação heurística chamada “utilidade de categoria” para guiar a construção da árvore.

Enquanto percorre o melhor caminho pela árvore em busca do melhor nó hospedeiro ao qual classificar o objeto, COBWEB aloca temporariamente o objeto em cada nó e calcula a “utilidade de categoria” da partição resultante. O arranjo que resultou na mais alta “utilidade da categoria” é indicado como hospedeiro ideal para o objeto. Se um objeto não pertencer a quaisquer dos conceitos já representados na árvore, uma nova classe é criada e automaticamente é ajustado o número de classes existentes em uma partição, dispensando que este seja um parâmetro a ser introduzido pelo usuário.

Para diminuir o impacto da sensibilidade apresentada por COBWEB para a ordem dos objetos, dois operadores adicionais podem ser utilizados: *merging* e *splitting*. Quando um objeto está sendo analisado, os dois melhores hospedeiros

podem ser fundidos em uma classe única ou os objetos do melhor hospedeiro podem ser divididos entre as categorias existentes. Estas decisões são baseadas na “utilidade da categoria” e permitem executar um busca bidirecional, por exemplo, uma fusão pode desfazer uma divisão anterior.

Algumas limitações desta abordagem podem ser apontadas: probabilidade estatística baseada na hipótese de atributos independentes, porém freqüentemente existe uma forte correlação entre eles; a representação da probabilidade dos grupos representa um alto custo de atualização e armazenamento, principalmente quando os atributos têm um largo espectro de valores; a árvore de classificação pode facilmente perder equilíbrio em sua altura, com a introdução de dados que reflitam uma forte inclinação, o que pode causar degradação dramática de tempo e espaço.

A abordagem de redes neurais busca representar cada grupo como um exemplar. Um exemplar atua como protótipo do grupo e não tem necessariamente que corresponder a um objeto específico. Novos objetos podem ser alocados ao grupo quando o seu exemplar apresentar a maior semelhança baseada em alguma medida de distância

Dois dos mais proeminentes métodos de abordagem de redes neurais são o aprendizado competitivo e os mapas auto-organizáveis de características, ambos envolvendo competição entre as unidades neurais.

Aprendizado competitivo: envolve uma arquitetura hierárquica de várias unidades (neurônios artificiais) que competem na forma vencedor-leva-tudo pelo objeto que está sendo apresentado no momento ao sistema. A Figura 2.4 mostra um exemplo de um sistema de aprendizado competitivo. Cada círculo representa uma unidade. A unidade vencedora dentro de um grupo se torna ativa (representada por um círculo preenchido), enquanto as outras estão inativas (indicadas por círculos vazios). As conexões entre as camadas são excitatórias - uma unidade numa dada camada pode receber entradas de todas as unidades do nível imediatamente abaixo. A configuração das unidades ativas em uma camada representa o padrão de entrada do nível imediatamente acima. As unidades dentro um grupo para uma dada camada competem entre si para responder ao padrão de entrada. As conexões dentro das camadas são inibidoras, uma vez que somente uma unidade dentro de qualquer grupo pode ser ativa. A unidade vencedora ajusta os pesos de suas conexões com as outras unidades do grupo, de maneira que haja uma resposta ainda mais forte para os futuros objetos que lhe sejam semelhantes. Se os pesos forem vistos como o exemplar, então

os novos objetos alocados ao grupo são os mais aproximados ao exemplar. O número de grupos e o seu número de unidades são parâmetros de entrada.

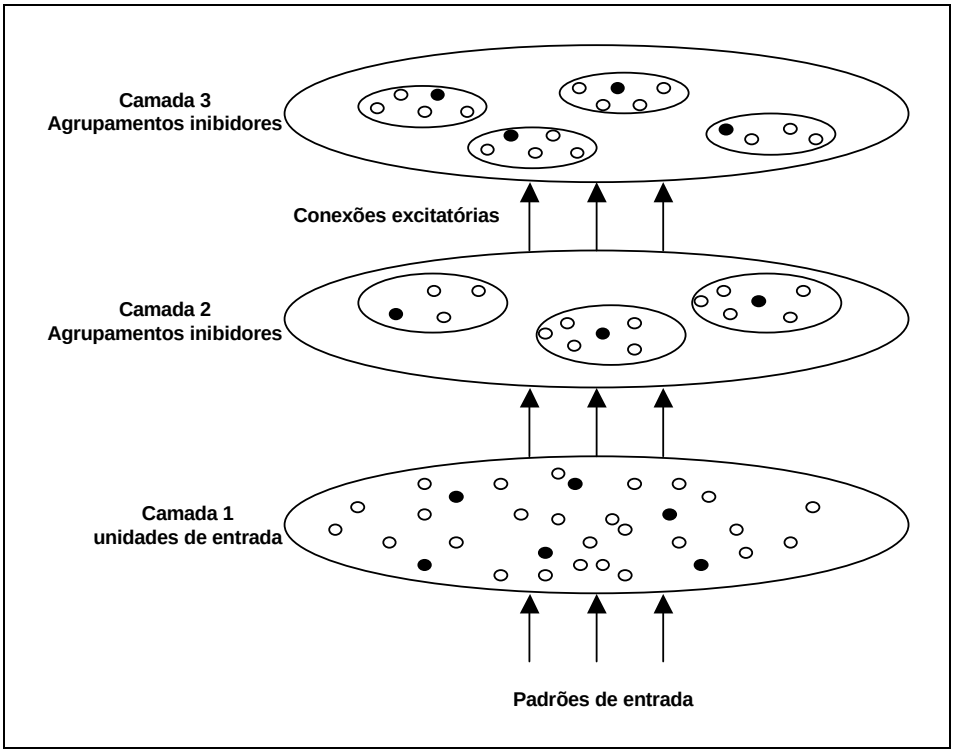


Figura 2.4: Uma arquitetura de aprendizado competitivo. Baseada em [1].

Ao fim da segmentação (e de qualquer outra segmentação, em geral), cada grupo pode ser encarado como uma nova característica descoberta com alguma regularidade na coleção de objetos. Assim, os grupos resultantes podem ser vistos como um mapeamento de características de baixo nível para características de alto nível.

A segmentação através de mapas auto-organizáveis de características (SOMs, sigla em inglês) é também realizada com a competição de várias unidades pelo objeto atual. A unidade cujo vetor de pesos é mais próximo ao objeto atual se torna o vencedor, ou unidade ativa. Ao fim do movimento de alocação do objeto mais semelhante, os pesos da unidade vencedora são ajustados, bem como aquele de seus vizinhos mais próximos, detalhe que faz a diferença entre os dois métodos. Este método de segmentação pressupõe que exista alguma topologia ou ordem entre os objetos introduzidos no algoritmo e que eles irão configurar esta estrutura ao final da segmentação. Acredita-se que a forma de processamento dos mapas auto-organizáveis de características assemelha-se ao processo que ocorre no cérebro humano e são muito úteis para visualização espacial de dados em duas ou três dimensões.

2.2 O método *"K-Means"*

Método particional clássico, o *"K-Means"* é baseado no conceito estatístico do centróide, de forma que a similaridade dos objetos que formem um mesmo grupo ("intragrupo"*)* seja a mais alta possível e a similaridade dos objetos de um grupos em comparação aos componentes dos outros grupos ("intergrupo"*)* seja a mais baixa possível. Como usualmente o *"K-Means"* utiliza medidas diretas de distância para calcular a similaridade entre os objetos, torna-se muito sensível a dados fora do padrão, ou seja, objetos que tenham um valor extremamente alto ou baixo, o que pode causar relevantes distorções nos resultados, com formação de agrupamentos de baixa qualidade de uma maneira geral. Outro cuidado que deve ser tomado é com a unidade (m^3, kg, $) em que os dados estão expressos: valores de atributos coerentes entre si e com os objetos que representam, medidos com unidades diferentes podem necessitar de alguma normalização antes de proceder-se à segmentação dos dados para minimizar as possíveis distorções causadas por valores absolutos muito distantes.

Antes de iniciar seu processo, o algoritmo *"K-Means"*, necessita receber o parâmetro k, que representa o número de grupos que se deseja formar com os dados a serem analisados. Assim, o primeiro passo do processamento propriamente dito será selecionar k objetos do conjunto de dados, aleatoriamente. Cada um desses objetos será o representante de seu grupo ainda unitário. Em seqüência, os demais objetos serão testados em cada grupo e serão alocados naquele ao qual tenham a maior similaridade, ou seja, a menor distância. A cada novo objeto alocado será calculada a média dos objetos do grupo – o centróide, contra quem será medida a distância dos objetos que ainda não foram alocados.

2.2.1 Medidas de distância

Para calcular a similaridade dos objetos é necessário utilizar alguma técnica de medida de distância. São diversas as medidas que podem ser utilizadas, podendo ser baseadas numa dimensão única ou em múltiplas dimensões. De qualquer forma, o algoritmo não faz distinção entre as técnicas que estão sendo empregadas, quem for empregar o algoritmo deve selecionar o melhor método para sua aplicação específica [15].

Distância Euclidiana: esta é provavelmente a mais comumente utilizada medida de distância.É simplesmente a distância geométrica dos objetos *x* e *y* em um espaço multidimensional, dada pela equação (2.1).

$$d(x,y) = \sqrt{\sum_i (x_i - y_i)^2} \tag{2.1}$$

Distância Euclidiana quadrática: é uma forma de empregar o padrão de distância Euclidiana dando, progressivamente, maior peso aos objetos mais distantes. É calculada pela equação (2.2).

$$d(x,y) = \sum_i (x_i - y_i)^2 \tag{2.2}$$

Note-se que a distância Euclidiana e a distância Euclidiana quadrática normalmente computam dados brutos e não dados normalizados, o que representa certas vantagens. Por exemplo, a distância entre dois objetos quaisquer não é afetada pela adição de novos objetos, ainda que sejam dados fora do padrão. Contudo, as distâncias podem ser grandemente afetadas por diferenças de escala entre as dimensões dos objetos analisados.

Distância *Manhattan* (*City-block*): esta distância é simplesmente a média da diferença entre os objetos. Na maioria dos casos, esta medida de distância obtém resultados parecidos com a distância Euclidiana. Contudo, note-se que, nesta medida, o efeito de dados fora do padrão é amortecido. É calculada com (2.3).

$$d(x,y) = \sum_i \left| x_i - y_i \right| \tag{2.3}$$

Distância *Chebychev*: é uma medida de distância apropriada em casos onde se deseja definir dois objetos como diferentes se eles forem diferentes em alguma de suas dimensões, dada por (2.4).

$$d(x,y) = Máximo \left| x_i - y_i \right| \tag{2.4}$$

Distância de Força: pode ser usada quando há interesse em aumentar ou diminuir sucessivamente o peso de dimensões nas quais os objetos apresentem muita diferença e é calculada pela equação (2.5), onde p e r são parâmetros definidos pelo usuário. O parâmetro p controla o peso que será sucessivamente atribuído às diferenças de uma mesma dimensão e o parâmetro r controla o peso que será atribuído à maior diferença entre os objetos.

$$d(x,y) = \left(\sum_i \left| x_i - y_i \right|^p \right)^{1/r} \quad (2.5)$$

2.2.2 Descrição do processamento

O modo de processamento do algoritmo *"K-Means"* pode ser descrito em breves passos:

1. primeiramente é esperada a entrada do parâmetro k, representando o número de grupos que se deseja formar;

2. são selecionados aleatoriamente, dentre os objetos da coleção a ser analisada, o mesmo número k de objetos que serão, inicialmente, grupos unitários e, por conseguinte, o próprio centro do grupo;

3. em seqüência, os objetos são comparados, um a um, com cada um dos centros dos grupos, sendo o objeto do momento alocado no grupo cujo centro lhe seja o mais próximo, utilizando para isso uma das medidas de distância;

4. todo grupo que recebe um novo objeto tem seu centro recalculado, através da média dos objetos que o compõem;

5. o processo é repetido iterativamente até que uma determinada função objetivo convirja. Tipicamente é utilizada soma do erro quadrático, dada por (2.6), onde E é a soma do erro quadrático de todos os objetos da base de dados, p é o ponto que representa um determinado objeto no espaço e m_i é o valor médio do grupo C_i (p e m_i são multidimensionais). Este critério procura formar grupos compactos e separados entre si.

$$E = \sum_{i=1}^{k} \sum_{p \in C_i} (p_i - m_i)^2 \quad (2.6)$$

A Figura 2.5 representa esquematicamente o algoritmo do método "K-Means".

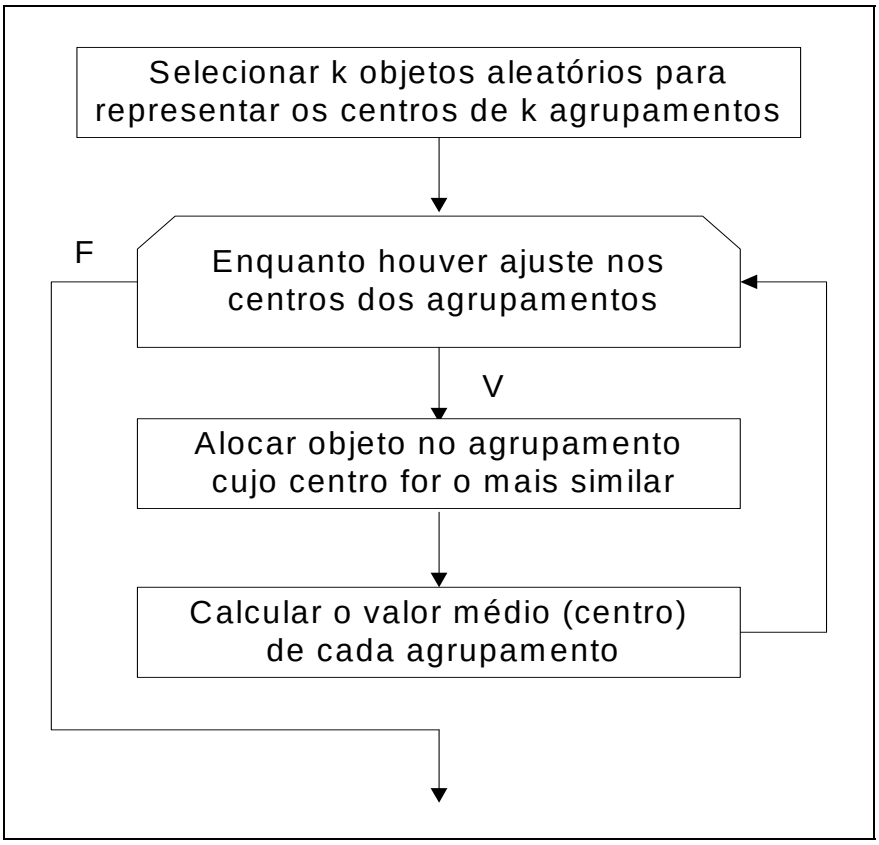


Figura 2.5: algoritmo do método *"K-Means"*.

O método *"K-Means"* deve ser aplicado nos casos em que o centro de um grupo pode ser bem definido, o que não é aplicável em alguns casos, quando por exemplo atributos categóricos estão envolvidos. A necessidade de k ser conhecido antecipadamente é uma considerável desvantagem. O método *"K-Means"* não é o mais eficiente para descobrir grupos com formas não convexas ou grupos de tamanho muito diferente. Além disso, ele é sensível a ruídos e dados fora do padrão, os quais, mesmo em ocorrências pequenas podem ter substancial influência no cálculo das médias (centro) dos grupos.

A Figura 2.6 ilustra sucintamente os passos do processamento do método *"K-Means"*. Inicialmente, são escolhidos 3 objetos aleatórios para representar o centro de cada um dos 3 grupos que deverão ser formados (os centros dos grupos estão marcados com o símbolo "+" em todas as figuras) e os objetos remanescentes são agrupados de acordo com suas distâncias em relação aos centros, como mostra a (a). Após a realocação dos objetos, formando nova composição de grupos, os novos centros são calculados, iterativamente, conforme mostrado na Figura 2.6(b). Quando não há mais possibilidade de realocar qualquer objeto, o processo dá-se por terminado, conforme visto na Figura 2.6(c), sendo a configuração dos grupos obtidos o resultado da segmentação realizada pelo algoritmo.

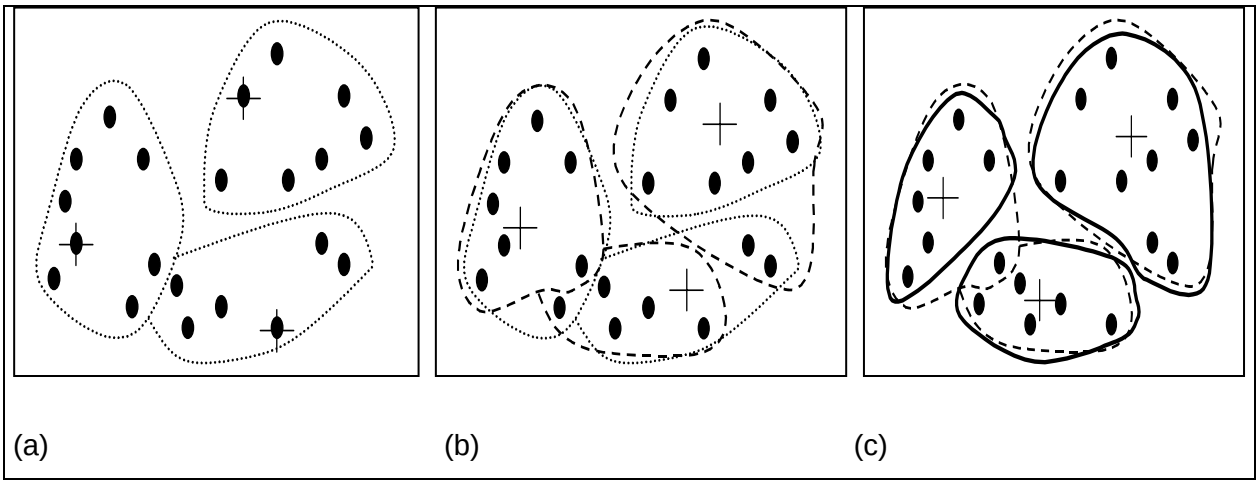


Figura 2.6: representação do método “K-Means” para agrupamento de objetos. Baseada em [1].

2.3 O problema da definição do número de grupos

A maior desvantagem do método *“K-Means”* certamente é a exigência do valor de k como parâmetro inicial. Para uma aplicação real, em grandes bases de dados, esta informação normalmente é desconhecida, podendo ser necessária uma prévia análise de um especialista para que houvesse possibilidade de percepção da quantidade de grupos em que a coleção poderia dividir-se de forma satisfatória, o que por si só já representa uma grande carga de trabalho que, até mesmo, diante de uma base muito grande poderia tornar a análise inviável ou desinteressante.

Existem vários critérios para a determinação do número de grupos e quase todos funcionam da seguinte maneira: realizar o agrupamento dos dados considerando 2 grupos e calcular o valor de uma função proposta que tenha o número de grupos como um de seus parâmetros, realizar o agrupamento dos dados considerando 3 grupos e calcular o valor da mesma função, repetir este procedimento até atingir um número máximo de grupos estabelecido. O agrupamento que ocasionar o valor máximo (ou, em alguns casos, mínimo) da função, deve ser considerado como o melhor agrupamento possível para a base de dados [16].

Existem algumas técnicas capazes de lidar com problemas de segmentação em que o valor de k é desconhecido. As técnicas hierárquicas que criam agrupamentos para a base de dados através de uma série de junções ou divisões sucessivas são bastante populares. Tais técnicas possuem o inconveniente de não

estabelecerem respostas prontas para o problema de segmentação. A saída (*output*) destes sistemas requer uma análise cuidadosa sobre o número de grupos e sobre a distância entre eles para se obter respostas adequadas [17].

Muitas outras técnicas têm sido sugeridas, baseadas em noções de densidade, variações do nível de quantização do espaço ou modelos matemáticos e estatísticos. Algumas destas técnicas têm alcançado bons resultados, mas apresentam em geral um custo computacional muito elevado [18].

Existem técnicas diversas capazes de avaliar a qualidade do resultado final de uma segmentação de dados. O método de *Calinski* e *Harabasz*, que estabelece o cálculo de um índice em função de k; o Critério de *Condorcet*, que faz a comparação entre pares de registros e o CCC- *Cubic Clustering Criterion*, baseado no método de Monte Carlo, são três exemplos que serão vistos a seguir em detalhes.

2.3.1 Método *Calinski* e *Harabasz*

Apresentado por CALINSKI e HARABASZ [19], este método estatístico utilizado para determinação da distribuição ótima de objetos em grupos apresenta uma boa margem de precisão em seus resultados tendo sido, inclusive, apontado por MILLIGAN e COOPER [20] em seu trabalho de 1985, comparando 30 critérios que podem ser adotados para otimização de agrupamentos, como o critério que apresentou os melhores resultados, conforme exposto em [18].

A função de otimização preconizada por CALINSKI e HARABASZ [19], é representada pela equação (2.7).

$$G(k) = (n-k) * B \Big/ (k-1) * W \tag{2.7}$$

Na equação apresentada em (2.7), *n* é o número total de objetos componentes da base analisada, e *k* é o número de grupos em que foi dividida essa base. O valor de *W* é obtido pela aplicação da equação (2.8) e *B* é a diferença entre *T* e *W*, demonstrado em (2.9).

Para proceder-se aos cálculos, primeiramente, devem ser formados grupos em número desejado. Como é de se esperar que a formação de tais grupos seja desconhecida, não se sabe quantos e quais objetos pertencem a quaisquer grupos.

Para a constituição inicial dos grupos, é possível adotar diversos procedimentos. Por exemplo, pode-se distribuir os objetos aleatória e eqüitativamente pelos grupos e, depois, destacar cada objeto de cada grupo e verificar a que grupo este seria mais similar, utilizando-se uma medida de distância. O grupo que guardasse a menor distância entre o objeto em foco e o seu centro receberia o novo objeto e a medida de seu centro seria atualizada. Tal procedimento seria repetido até que fizesse convergir uma determinada função objetivo que, tipicamente, é dada por:

$$W = \sum_{i=1}^{K} \sum_{j=1}^{n_i} (X_{ij} - \overline{X}_i)^2 \qquad (2.8)$$

que é a expressão da soma, realizada em *d* dimensões, dos quadrados das distâncias dos objetos ao centro de seu grupo, onde X_{ij} é o *j*-ésimo objeto do grupo *i*, $\overline{X}_i$ é o centro do grupo (média ou mediana dos objetos dentro do grupo) e n_i é a quantidade de objetos que estão no grupo *i*. É importante ressaltar que os objetos e os centros dos grupos são vetores do espaço de *d* dimensões e a equação acima costuma ser chamada de *W* por considerar a soma no interior de cada grupo.

T é o resultado da soma, também realizada em *d* dimensões, dos quadrados das diferenças entre cada objeto da base de dados e o centro de toda a base $\overline{X}$, expressa por (2.9).

$$T = \sum_{i=1}^{K} \sum_{j=1}^{n_i} (X_{ij} - \overline{X})^2 \qquad (2.9)$$

O valor representado por *B*, que pode ser obtido pela diferença entre T e W, é o somatório dos produtos entre o número de objetos componentes da base e os quadrados das diferenças entre o centro de toda a base e o centro de cada grupo, definido por (2.10).

$$B = T - W = \sum_{i=1}^{k} n_i (\overline{X_i} - \overline{X})^2 \qquad (2.10)$$

Ao ser atingida a configuração final de conjunto de grupos formados, deve ser calculado o valor de *G*. O número de grupos ideal é aquele que maximizar a função de *Calinski* e *Harabasz*.

Apesar desta função ficar indefinida para k = 1 e assumir o valor 0 quando $k=n$ não a torna menos útil, por serem, sobretudo, casos extremos que fogem aos casos onde seja preciso alguma análise de agrupamento. Ora, se o número de grupos ideal para uma determinada base é apenas 1, esta base não precisa, ou não admite, ser segmentada. Por outro lado, se uma base tem tantos grupos quantos objetos a sua segmentação também já está definida de origem e não há necessidade de nenhum método de segmentação para reconhecer-lhes os grupos, que são obviamente individuais.

Para facilitar a compreensão das linhas gerais deste método, é apresentado um algoritmo em forma de fluxograma na Figura 2.7.

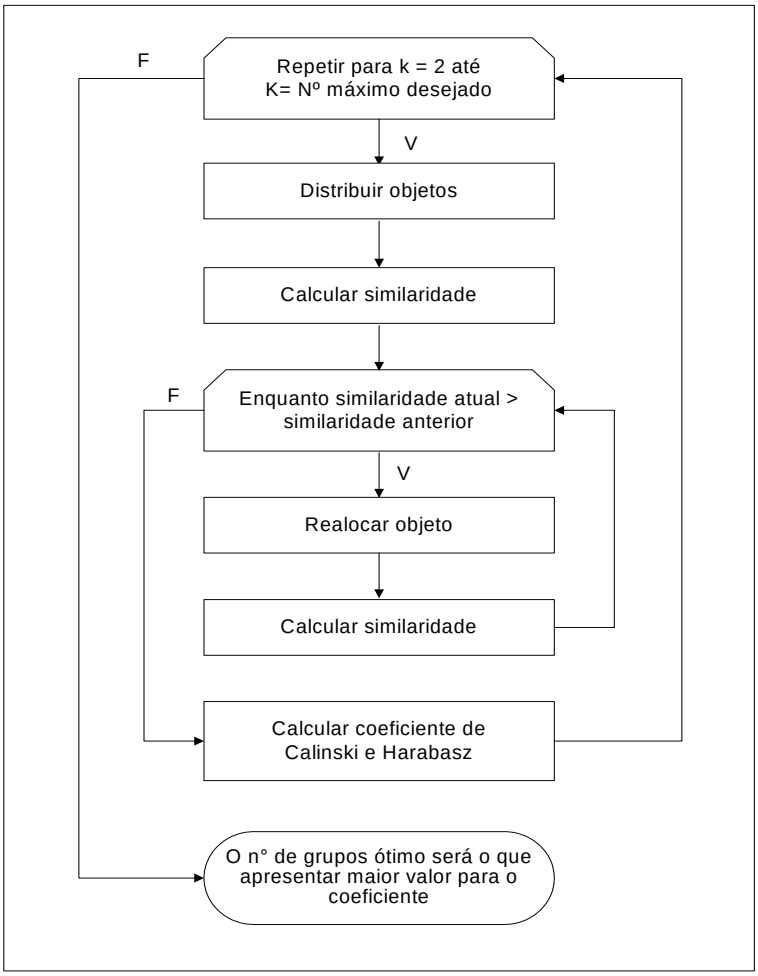


Figura 2.7: Algoritmo do método de *Calinski* e *Harabasz*.

2.3.2 Critério *Condorcet*

A produção de dados pelos sistemas de informação baseados em computador vem se ampliando numa escala vertiginosa. Conseqüentemente, bases

com gigantescos volumes de dados vêm sendo armazenadas. Para análise de agrupamento em bases excessivamente volumosas são necessárias técnicas mais especificas, com algoritmos mais robustos, capazes não só de poder tratar do imenso número de dados, como de empregar critérios que apresentem razoável desempenho em sua função de estabelecer grupamentos ótimos.

Um dos métodos utilizados com tal finalidade é a maximização do Critério de *Condorcet* [21], cujo postulado é:

(soma de todos os pares de registros de similaridades no mesmo grupo) – (soma de todos os registros de similaridade em diferentes grupos).

A semelhança entre um par de registros é determinada através da comparação dos valores de cada um de seus campos. Toda coincidência aumenta a semelhança, ao passo que valores diferentes diminuem a similaridade. Assim, estes algoritmos requerem a criação de categorias para os valores dos atributos. Por exemplo, se um determinado atributo é referente às cores de um objeto, um código numérico arbitrário utilizado para representá-las é a própria categoria. Em outra possibilidade, se um atributo possui valores contínuos, como temperaturas ou salários, estes são agrupados por faixa de valores, sendo a seqüência das faixas a categorização. Neste caso é preciso arbitrar um número inicial de categorias e verificar a qualidade do histograma obtido, realizando os devidos ajustes até encontrar o valor ideal.

Para cada grupo, é calculado o Índice de *Condorcet*, o qual reflete a homogeneidade ou similaridade de cada grupo, dado por (2.11).

$$\sum_{t} \pi_t^2 \tag{2.11}$$

onde π_t representa a freqüência relativa da categoria *t* do atributo dentro do grupo específico. O Índice de *Condorcet* final para um dado agrupamento é a média aritmética dos índices para cada grupo. Um índice igual a 1 indica que aquele grupo é completamente homogêneo, ou seja, cada registro desse grupo tem exatamente os mesmos valores em campos coincidentes. Agrupamentos com formações ótimas, verificadas na prática, possuem Índices de *Condorcet* situados de 0,5 a 0,8.

2.3.3 CCC – *Cubic Clustering Criterion*

O *Cubic Clustering Criterion – CCC* é utilizado para estimar números de grupos utilizando *Ward's minimum variance method*, *"K-Means"* ou outros métodos baseados na minimização da soma quadrática intra-grupos. O desempenho do CCC é avaliado pelo método de Monte Carlo [22].

Os critérios de separação em grupos mais empregados são conhecidos como soma intra-grupos quadrática, soma de erros quadrática, soma residual quadrática, mínimos quadrados, erros mínimos quadrados, variância mínima e distância euclidiana. Para definição deste critério é usada a seguinte notação:

n: número de objetos;

n_k: número de objetos no *k*-ésimo grupo;

p: número de variáveis;

q: número de grupos;

X: matriz de dados *n* x *p*;

$\overline{X}$: matriz q x p com as média dos grupos;

Z : matriz indicadora de grupos, onde o elemento z_{ik} é igual a 1 se a *i*-ésima observação pertence ao *k*-ésimo grupo ou igual a 0, caso contrário.

Assumindo-se que cada variável tem média 0, sem prejuízo da generalização, nota-se que $Z'Z$ é uma matriz diagonal contendo os n_ks e que: $\overline{X}$ e dado por (2.12).

$$\overline{X} = \left(Z'Z\right)^{-1} Z'X \tag{2.12}$$

A matriz *sum-of-squares and cross products* (SSCP) da amostra total é dada por (2.13).

$$T = X'X \tag{2.13}$$

A matriz SSCP "intergrupos" é dada pela equação (2.14).

$$B = \overline{X}'Z'Z\overline{X} \tag{2.14}$$

A matriz SSCP “intragrupos” tem a sua forma geral apresentada em (2.15).

$$W = \left(X - Z\overline{X}\right)' \left(X - Z\overline{X}\right) = X'X - \overline{X}'Z'Z\overline{X} = T - B \qquad (2.15)$$

A soma “intragrupos” dos quadrados das variáveis de (2.15) é, por conseguinte, o traço de W. Mudando a ordem das adições, também pode ser mostrado que o traço de W é igual à soma das distâncias Euclidianas quadráticas de cada objeto ao centro de seu grupo.

Já que T é constante para uma dada coleção de dados, a minimização de *traço(W)* é equivalente a maximização de R^2, apresentada em (2.16), o qual é comumente interpretado como a razão de discrepância para dado grupo. R^2 também pode ser obtido por regressão múltipla, se as colunas de *X* assumem a forma de uma pilha, formando um vetor com *np* ocorrências. Este vetor é regredido pelo produto de *Kronecker* de *Z* com uma matriz identidade de ordem *p*.

$$R^2 = 1 - \frac{Tr(W)}{Tr(T)} \qquad (2.16)$$

O maior problema em análise de agrupamentos é mesmo determinar o número de ótimo de grupos. Se um bom critério de ajuste, tal como R^2, está sendo usado, conhecer a distribuição da amostra obtida pela aplicação do critério facilitaria a realização de testes do significado da segmentação.

Testes de significado ordinários, tais como análise de discrepância, não são aplicáveis para verificar diferenças entre grupos. Visto que os métodos de segmentação buscam a maximização da separação entre os grupos, as hipóteses usuais dos testes de significância, paramétricos ou não-paramétricos, são drasticamente violados.

O primeiro passo para a criação de um teste de significado válido para formação de agrupamentos é especificar uma hipótese de alternativa nula. Dentre os métodos de segmentação baseados em matrizes de distâncias, uma das mais populares hipóteses de alternativa nula considera que a permutação dos valores da matriz de distâncias é uniforme. Pode-se utilizar tal hipótese para realizar um teste de permutação ou de classificação. A grande dificuldade é que, em dados reais, esta hipótese nula é totalmente improvável, até mesmo se os dados não formam grupos.

Outra hipótese nula bastante comum considera os dados como uma amostra aleatória de uma distribuição normal multivariada. Esta hipótese nula é melhor do que a de permutação nula, mas não é satisfatória porque existe uma probabilidade alta de rejeição se os dados da amostra apresentam uma distribuição com curtose abaixo da distribuição normal, como é o caso de uma distribuição uniforme.

O CCC é uma aproximação para o critério R^2 sob a hipótese nula que os dados da amostra têm uma distribuição uniforme como um *hiperbox* (paralelepípedo regular com *p* dimensões). Esta aproximação é útil para determinação do melhor número de grupos de dados, tanto univariados quanto multivariados. A aproximação do valor esperado de R^2 é baseada na hipótese dos grupos terem uma forma similar a um hipercubo. Esta aproximação tende a ser conservadora para um número pequeno de grupos e levemente liberal para um número grande (por volta de 25 grupos com duas dimensões). O CCC é obtido por comparação entre o R^2 obtido e o esperado, usando uma aproximação de discrepância. Valores positivos do CCC significam que o R^2 obtido é maior do que seria esperado para uma amostra com distribuição uniforme e, portanto, indicam a possível presença de grupos. Considerando-se o CCC como um teste estatístico padrão, chega-se a duas hipóteses gerais:

os dados fazem parte de uma amostra com distribuição uniforme, como uma *hiperbox*;

os dados fazem parte de uma mistura de distribuições normais multivariadas esféricas com iguais discrepâncias e amostras com iguais probabilidades.

O CCC é baseado na hipótese de que grupos obtidos com uma distribuição uniforme são *hipercubos* de mesmo tamanho. A suposição de um *hipercubo* é obviamente falsa na maioria dos casos, mas é razoável a menos que o número de grupos seja muito extenso em duas ou mais dimensões.

Produzido pelo SAS *Institute*, o *software* FASTCLUS gera segmentações baseado no CCC. A Figura 2.8 apresenta uma simulação do resultado de um caso no qual a suposição de um *hipercubo* (ou quadrado, neste caso, já que existem duas únicas dimensões) é aplicável corretamente: uma amostra de 10.000 objetos com uma distribuição uniforme foi dividida em nove grupos por FASTCLUS. Cada grupo assume uma forma muito próxima de um quadrado cujos lados são 1/3 do comprimento total do conjunto.

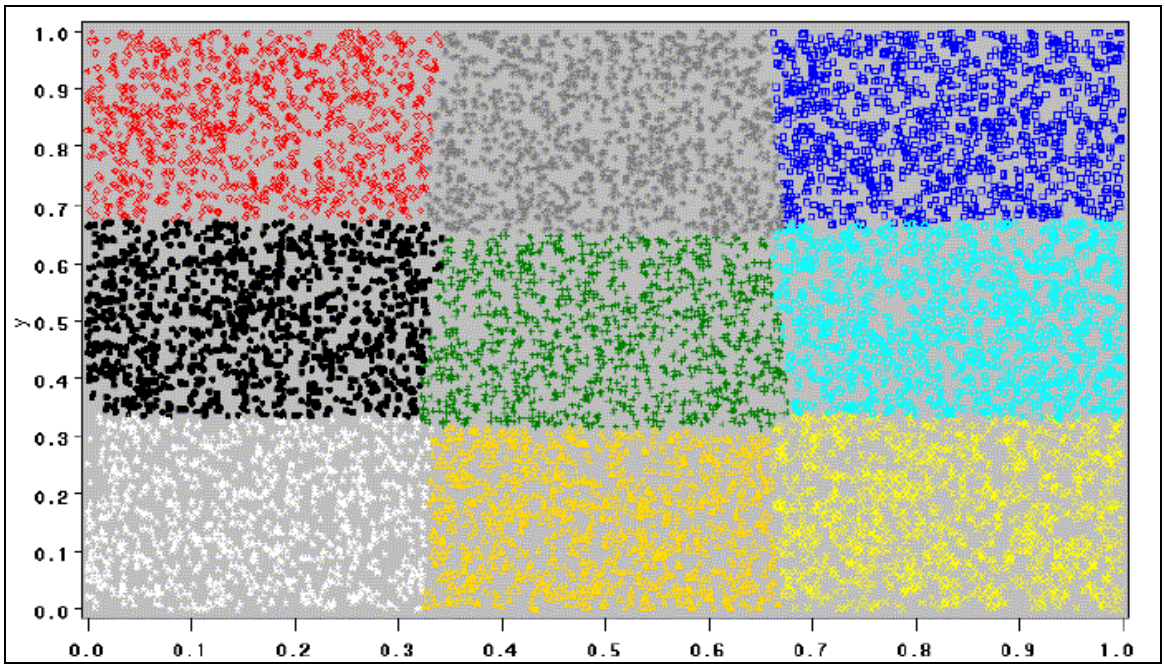


Figura 2.8: segmentação de amostra em nove grupos. Baseada em [22].

A primeira aproximação do valor de R^2 para uma população distribuída uniformemente numa *hiperbox* pode ser obtida da seguinte forma: presumir que as bordas do *hiperbox* estão alinhadas com os eixos de coordenadas. Considerar *Sj*, que é o tamanho da borda na direção j, coincidente com a extremidade da *hiperbox* ao longo da j-ésima dimensão e presumir, também, que os *Sj*s são decrescentes. O volume do hiperbox *v*, com *p* dimensões, é dado pela equação (2.17).

$$v = \prod_{j=1}^{p} Sj \tag{2.17}$$

Se a *hiperbox* está dividida em q *hipercubos* com aresta de tamanho *C*, então o volume da *hiperbox* é igual ao volume total dos *hipercubos*. Conseqüentemente, encontra-se (2.18).

$$C = \left(\frac{v}{q}\right)^{\frac{1}{p}} \tag{2.18}$$

Pode-se considerar, assim, u_j, dado por (2.19), como o número de *hipercubos* ao longo da j-ésima dimensão da *hiperbox*. A variação total da amostra ao longo da j-ésima dimensão é proporcional S_j^2, enquanto a variação “intragrupo” é proporcional a C^2.

$$u_j = \frac{S_j}{C} \tag{2.19}$$

Assim, chega-se à equação (2.20).

$$R^2 = 1 - \frac{\sum_{j=1}^{p} C^2}{\sum_{j=1}^{p} S_j{}^2} = 1 - \frac{p}{\sum_{j=1}^{p} u^2{}_j} \tag{2.20}$$

Na Figura 2.8, por exemplo, $S_1 = S_2 = 1$, $C = 1/3$ e $u_1 = u_2 = 3$. Portanto, chega-se a aproximação expressa em (2.21), enquanto para a amostra R^2 = 0,88967+.

$$R^2 = 1 - \frac{2}{(3^2 + 3^2)} = 0{,}8888... \tag{2.21}$$

A aproximação apresentada em (2.21) falha se o número de colunas efetivas para uma segmentação ideal, chamado *p**, é menor que *p*. Obviamente, *p** deve ser menor que o número de grupos, *q*. Da mesma forma, $u_j < 1$ implica em $p^* < 1$. Para uma melhor aproximação deve-se considerar que os grupos são hiperboxes com aresta de tamanho *C* nas primeiras *p** dimensões e aresta S_j nas demais dimensões. Devendo-se considerar, então v^*, como dado em (2.22), C, como dado em (2.23) e u_j como dado em (2.19), onde p* está destinado a ser um inteiro menor do que q, de tal forma que u_{p^*} não seja menor que 1. Assim chega-se a aproximação para R^2, expressa em (2.24).

$$v^* = \prod_{j=1}^{p^*} S_j \tag{2.22}$$

$$C = \left(\frac{v^*}{q} \right)^{\frac{1}{p^*}} \tag{2.23}$$

$$R^2 = 1 - \frac{p^* + \sum_{j=P^*+1}^{p} u_j{}^2}{\sum_{j=1}^{p} u_j{}^2} \tag{2.24}$$

Em pequenas amostras com distribuição uniforme numa *hiperbox*, as amostras R^2 s tendem a superar a população R^2 devido a um fenômeno conhecido como "*capitalization on chance*". Extensas simulações conduziram à heurística de aproximação do valor esperado de R^2, expressa em (2.25), para pequenas amostras.

$$E(R^2) = 1 - \left[\frac{\sum_{j=1}^{p^*} \frac{1}{n+u_j} + \sum_{j=p^*+1}^{p} \frac{u^2{}_j}{n+u_j}}{\sum_{j=1}^{p} u^2{}_j} \right] \left[\frac{(n-q)^2}{n} \right] \left[1 + \frac{4}{n} \right] \qquad (2.25)$$

Dada uma amostra *X*, considere-se S_j como sendo a raiz quadrada da j-ésima transformação de *T / (n – 1)*, assumindo-se como hipótese nula o tamanho da *hiperbox* na j-ésima dimensão ser proporcional ao padrão de divergência da j-ésima componente principal dos dados. O CCC é computado pela fórmula (2.26).

$$CCC = \ln\left[\frac{1 - E(R^2)}{1 - R^2} \right] \frac{\sqrt{\frac{np^*}{2}}}{(0{,}001 + E(R^2))^{1,2}} \qquad (2.26)$$

A fórmula (2.26) foi derivada empiricamente numa tentativa de estabilizar a discrepância através de diferentes números de objetos, atributos e grupos.

O melhor caminho para usar o CCC é comparar o seu valor contra o número de grupos, variando de 1 até um décimo do número de objetos. O CCC pode não ter um bom comportamento se o número médio de objetos por grupo for menor que dez.

Ao obter-se o CCC para as segmentações realizadas com os diversos números de grupos, usa-se um gráfico, traçando-se o valor do CCC x número de grupos, utilizando-se os seguintes passos para interpretação:

- picos de valores para CCC maiores que 2 ou 3 indicam boa segmentação;

- picos entre 0 e 2 indicam possíveis agrupamentos, mas o resultado da segmentação deve ser analisado mais cautelosamente;

- vários picos indicam que os dados possuem uma estrutura hierárquica;

- agrupamentos esféricos não hierárquicos muito distintos normalmente mostram uma subida afilada antes do pico e um declínio gradativo em seguida;

- agrupamentos de forma elíptica não hierárquicos muito distintos freqüentemente mostram uma subida afilada em direção ao número correto de grupos, seguido por um aumento gradativo e, finalmente, um declínio gradativo;

- se todos os valores do CCC são negativos e diminuem para dois ou mais agrupamentos, a distribuição é provavelmente unimodal ou de cauda longa;

- grandes valores negativos do CCC, por volta de –30, podem ser decorrência de dados fora do padrão, os quais deveriam ter sido removidos antes da segmentação;

- se o CCC aumenta continuamente, em paralelo com o número de grupos, a distribuição pode ter uma alta granularidade, ou os dados podem ter sido excessivamente arredondados ou foram utilizadas poucas casas decimais;

Finalmente, uma nota importante: o CCC não é um critério apropriado para grupos com formas extremamente compridas ou irregulares. Se não há indícios substanciais que apontem para grupos compactos, o mais indicado seria utilizar um método de segmentação não paramétrico.

2.3.4 Índice PBM

O Índice PBM, desenvolvido por PAKHIRA, BANDYOPADHYAY e MAULIK [23], pode ser utilizado para avaliar a qualidade de segmentação de bases de dados em diferentes partições, podendo ser aplicado tanto em segmentações clássicas quanto *fuzzy*. Este índice é obtido pela composição de três fatores, conforme apresentado na equação (2.27), e a sua maximização dá-se em torno de formações com pequeno número de grupos compactos e com uma grande separação entre dois deles, ao menos.

$$PBM(K) = \left(\frac{1}{K} * \frac{E_1}{E_K} * D_K \right)^2 \qquad (2.27)$$

o primeiro fator, $\frac{1}{K}$, da equação (2.27), onde K é o número de grupos, mostra claramente que o valor do Índice PBM é inversamente proporcional ao número de grupos encontrados, ou seja, o índice tende a apontar como ótimas as segmentações formadas pelos menores números de grupos.

O segundo fator, $\frac{E_1}{E_K}$, onde E_K é dado pela equação (2.28), representa a razão entre E_1, constante para uma mesma base e E_K, o qual diminui com o aumento de K. Conseqüentemente o Índice PBM aumenta na proporção que E_K diminui.

Com o intuito de evitar que o segundo fator atinja um valor muito pequeno emprega-se um numerador constante, E_1, obtido pela soma das distâncias de todos os objetos ao centro da base, como se formassem um único grupo. Como este fator mede a densidade dos grupos formados, é sempre desejável que seu valor seja o mais alto possível.

$$E_K = \sum_{k=1}^{K} E_k \tag{2.28}$$

O fator E_k da equação (2.28) é dado pela equação (2.29), onde n é o número total de objetos da base de dados, $U(X) = \lfloor u_{kj} \rfloor_{Kxn}$ é a matriz particional dos dados, sendo que u_{kj} representa a pertinência do objeto x_j ao grupo C_k, e Z_k é o centro do *kth* grupo. No método de segmentação clássica, pode-se seguramente utilizar $u_{kj} = 1$, se $x_j \in C_k$, caso contrário $u_{kj} = 0$.

Nos casos de uso de segmentação *fuzzy*, a intenção é evoluir adequadamente a matriz particional $U(X) = \lfloor u_{kj} \rfloor_{Kxn}$, onde $u_{kj} \in \lfloor 0,1 \rfloor$, tal que u_{kj} denote o grau de pertinência do *jth* objeto ao *kth* grupo.

$$E_K = \sum_{j=1}^{n} u_{kj} \left\| x_j - z_k \right\| \tag{2.29}$$

O terceiro fator da equação (2.27), D_K, é dado pela equação (2.30) e pode ser traduzido na medida da maior distância encontrada entre os grupos formados no processo de segmentação, tomados dois a dois. O valor deste componente, que

aumenta de acordo com o valor de K, é puxado para cima em função da separação máxima entre dois objetos da base.

$$D_K = \max_{i,j=1}^{K} \left\| z_i - z_j \right\| \tag{2.30}$$

A situação pretendida, além da maximização do índice, é manter o menor número de grupos possível, enquanto a densidade interna e a separação externa dos grupos aumentam tanto quanto possível.

Reconhecidamente, são diversos os índices que podem ser empregados na avaliação da qualidade de agrupamentos, porém, dificilmente, encontra-se algum que cumpra a missão em largo espectro de bases de dados. Proposto sob uma detalhada análise matemática, o Índice PBM pode ser utilizado em conjuntos de dados com as mais diversas configurações, podendo-se utilizá-lo tanto no método clássico quanto no *fuzzy*, a fim de avaliar as segmentações realizadas.

CAPÍTULO 3
MÉTODOS DE VISUALIZAÇÃO DE RESULTADOS

Visualização e mineração de dados são duas técnicas usadas freqüentemente para criar e desenvolver bem sucedidas soluções de inteligência na área de negócios ou de pesquisa. Aplicando visualizações e técnicas de mineração de dados torna-se possível descobrir tendências e anomalias previamente desconhecidas:

Ferramentas de visualização de dados e suas técnicas permitem criar quadros em duas e três dimensões relativos a conjuntos de dados que podem ser facilmente interpretados pelos usuários, gerando ganho de conhecimentos e *insights*. Podendo ajudá-los, ainda, a criar modelos que descobrem padrões, predizendo novas oportunidades e colaborando no processo de tomada de decisão.

Em ambos casos, a visualização é a chave, auxiliando no esforço de análise de dados para descoberta de novos padrões e tendências encontrados, mas inicialmente desconhecidos, dentro dos grupos de dados comerciais. A visualização é um método de eficiência comprovada para comunicar tais descobertas aos responsáveis pela tomada de decisão [24].

Neste capítulo, serão exploradas as possibilidades das técnicas de visualização para análise de dados. Os gráficos, chamados genericamente de padronizados, apresentados nos primeiros exemplos a seguir, são uma forma de visualização que auxiliam na análise de agrupamentos, porém, quando apresentam relações que envolvem simples e diretamente os atributos dos objetos, ficam muito dependentes do analista, o qual terá grande esforço em analisar e apreender os padrões e tendências que podem ser revelados pelos gráficos. Ao utilizarem valores obtidos por um tratamento prévio, como as técnicas de segmentação utilizadas nesses mesmos exemplos, a visualização permite uma riqueza maior de conclusões, porém exige que conhecimento e aplicação avançados das ferramentas necessárias ao agrupamento de dados.

Além dos gráficos padronizados, existem gráficos que não seguem uma padronização, tendo formatos e modos de apresentação bastante peculiares, específicos de determinadas ferramentas. Combinam recursos de texto, gráficos

padronizados e computação gráfica avançados. Para exemplificar esta classe de técnicas de visualização, serão apresentados gráficos particulares de quatro *softwares* de análise e mineração de dados: o IBM DB2 *Intelligent Miner*, que concentra um conjunto de funções de pesquisas, estatísticas e de pré-processamento destinadas à análise de grandes bancos de dados; o SAS *Enterprise Miner*, poderosa ferramenta de mineração e visualização de dados, que fornece ajuda na solução dos problemas encontrados no estudo do comportamento de conjuntos de dados, atuando tanto na segmentação quanto na obtenção de modelos de predição; o *Mineset Enterprise Edition*, da *Silicon Graphics*, Inc, aplicativo que oferece recursos para classificação, regressão, associação e agrupamento (*clustering*) que podem ser combinados com ótimas técnicas de visualização para *data mining*, como visualizadores de dispersão, de árvore e estatísticos. A fim de demonstrar alguns dos gráficos produzidos por estes três softwares, usaremos exemplos descritos por PUNTAR [25]. E, por fim, o *gCLUTO – Graphical CLUstering TOokit*, da Universidade de *Minnesota*, interface amigável para o CLUTO, versátil biblioteca de funções de segmentação de dados, que provê diversos modos de visualização interativa dos resultados do processo de agrupamento.

Os gráficos baseados em “Coordenadas Estrela” e dendrograma fornecem uma visualização mais complexa e muito transparente em relação às ferramentas utilizadas para o cálculo dos valores plotados nos gráficos. O analista deve ser especialista na base de dados, ficando dispensado do conhecimento técnico das técnicas empregadas para a segmentação e apresentação dos resultados, o que potencializa a análise dos dados e a conseqüente descoberta de conhecimento.

3.1 – Gráficos padronizados

A ferramenta de visualização a ser usada é diretamente relacionada à estrutura interna do conjunto de dados que está sendo manipulado, além do aspecto específico que está sendo investigado.

Na maioria dos casos, quando é traçado o gráfico apropriado com os dados em estudo, é possível observar os diversos tipos de relacionamentos que podem ocorrer entre os atributos de uma base. A partir dessas observações e outras descobertas visuais são obtidos *insights* que ajudam a encontrar respostas para as

questões sob investigação. Além disso, a visualização de dados pode ser usada como uma ferramenta facilitadora da comunicação entre especialistas e tomadores de decisão.

Alguns gráficos como os mostrados abaixo, dentre os mais conhecidos, podem ser citados a título de exemplo do potencial da visualização para análise de dados e descoberta de conhecimento. Para construção de tais gráficos, foi utilizada a base *iris plants database*, largamente conhecida nos ambientes de pesquisa, possuindo 150 objetos com 4 atributos, largura da pétala, largura da sépala, altura da pétala, altura da sépala. A base tem uma divisão natural em 3 grupos de espécies de flores. Inicialmente, foram gerados gráficos com os dados brutos da base, com o intuito de explorar as possibilidades de análise oferecidas pela visualização das relações entre os objetos e seus atributos. Em seguida, utilizando-se técnicas de segmentação, no caso o algoritmo *k-means*, os objetos da base foram agrupados em três grupos distintos, a fim de serem apresentados recursos de visualização empregando valores compostos, obtidos com algum tipo de cálculo realizado sobre os objetos.

3.1.1 – Gráfico de Linhas

Um gráfico de linhas, utilizado como meio de visualização, facilita a percepção das tendências existentes nos dados em intervalos iguais. Ainda utilizando como exemplo a base *iris plants database*, pode-se notar, na Figura 3.1, que três dos quatro atributos apresentam uma tendência de elevação de seus valores, enquanto o outro atributo mantém uma constância em seus valores, com uma ligeira queda fazendo contraposição ao demais.

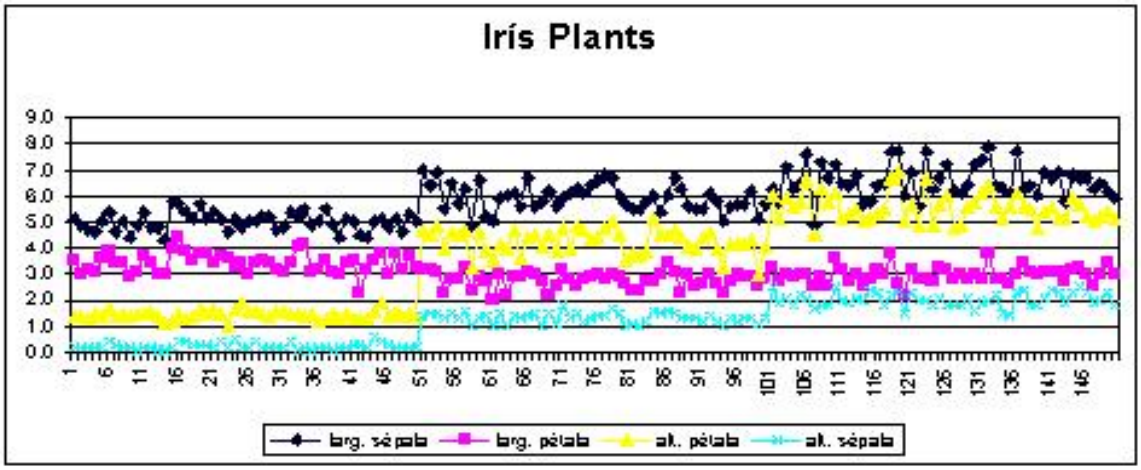


Figura 3.1: gráfico de linha dos objetos.

3.1.2 – Gráfico de Dispersão

Um gráfico de dispersão permite a visualização de dados agrupados ou em intervalos irregulares. Esse gráfico mostra as relações entre os valores numéricos em várias seqüências.

A Figura 3.2 traz como exemplo quatro gráficos que mostram as relações entre os atributos dos objetos da base *iris plants database*. Estes gráficos, analisados sob a ótica de um especialista, podem fornecer insumos que permitam vislumbrar o comportamento dos grupos de objetos através da dispersão dos seus atributos.

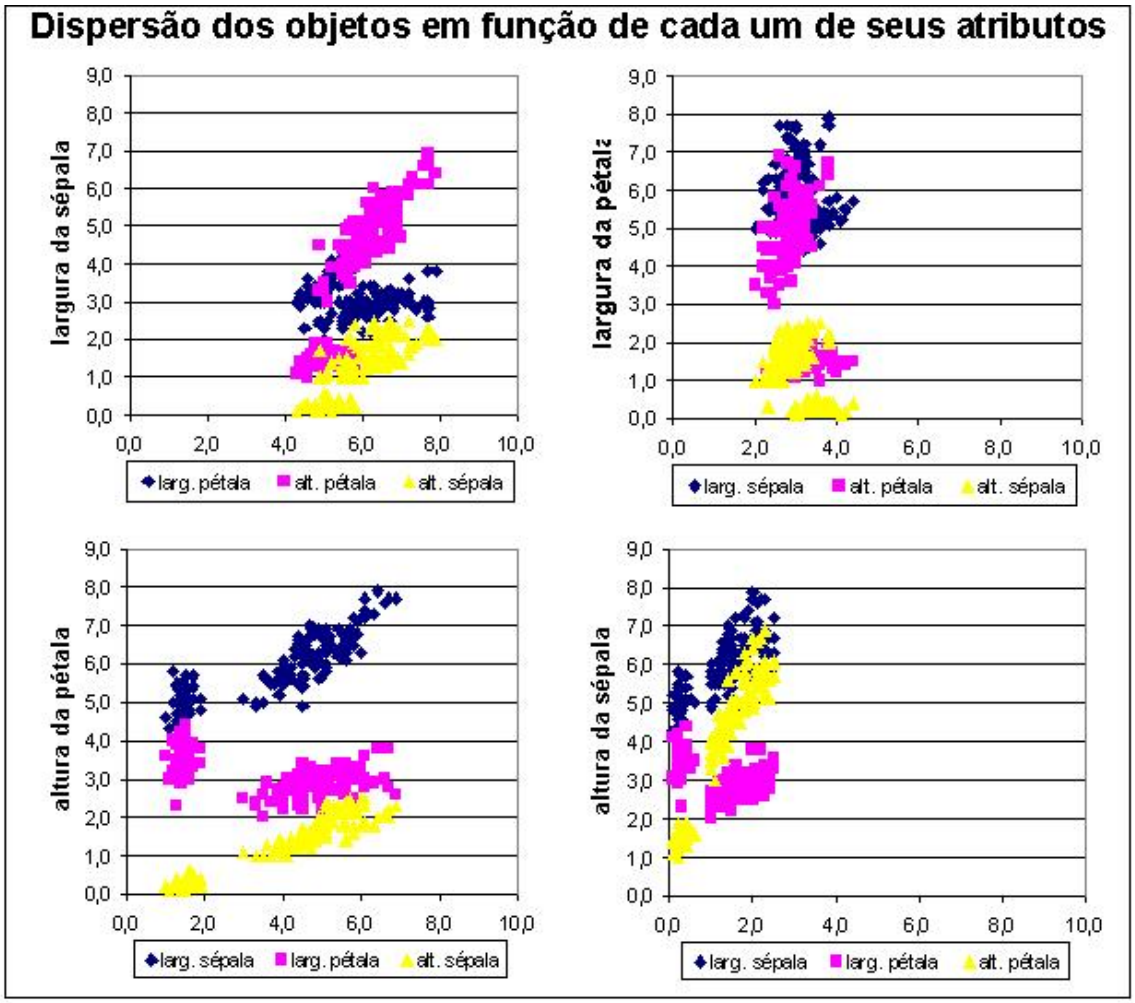


Figura 3.2: gráfico de dispersão dos atributos.

A Figura 3.3, a seguir, apresenta um gráfico de dispersão que facilita a visualização da similaridade entre os objetos dos grupos formados, ou seja, quanto mais próximos do centro os objetos se encontrem, melhor será a qualidade da segmentação. Assim, torna-se fácil concluir, pela visualização de um gráfico que permite a comparação das distâncias dos vários objetos ao centro de seus grupos,

que o grupo 1 é muito bem formado, com os objetos que o compõem muito próximos uns dos outros e do próprio centro. Os grupos 2 e 3 já apresentam uma dispersão maior, sendo alguns objetos do grupo 2 tão distantes que podem ser objetos pertencentes a outro grupo, mas que por características individuais não permitem uma separação linear, confundindo os algoritmos de agrupamento.

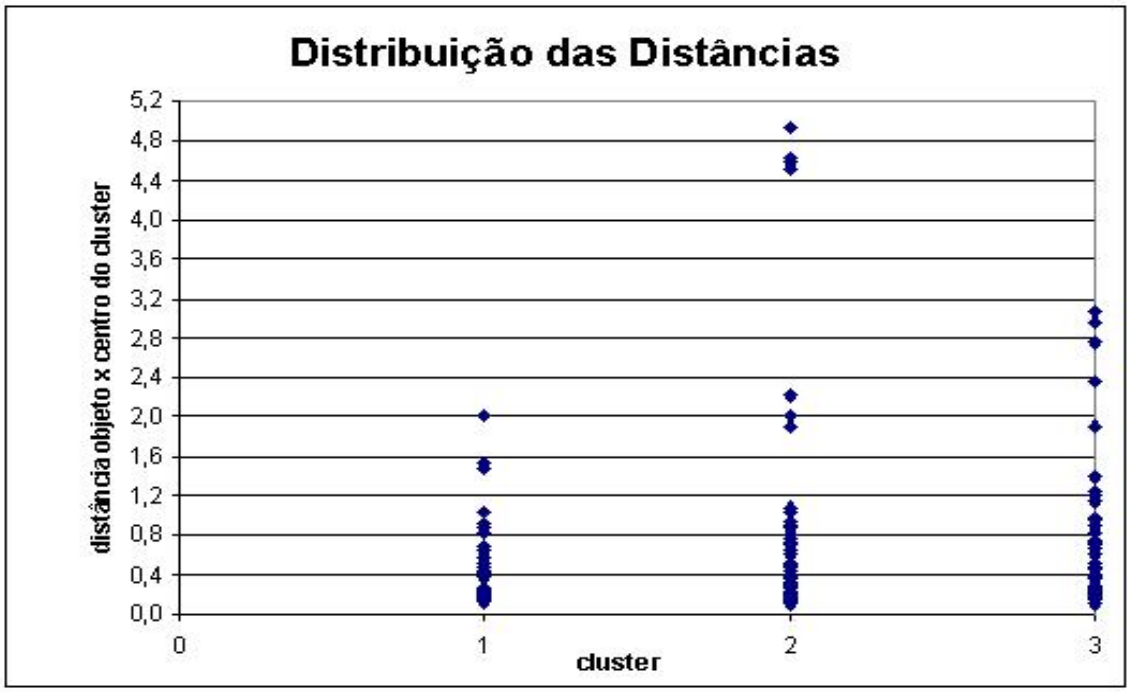


Figura 3.3: gráfico de dispersão das distâncias dos objetos ao centro de seu grupo.

3.1.3 – Gráfico de Radar

Após o cálculo da distância entre o centro dos 3 grupos formados pode ser traçado um gráfico do tipo radar, que apresenta, de forma simples e direta, a similaridade entre os grupos. Na Figura 3.4, é apresentado tal gráfico com as distâncias euclidianas, em escala logarítmica de base 10. Um especialista certamente poderá utilizar as informações condensadas em único gráfico, como este, para fundamentar a sua percepção na análise do conjunto de dados. Contudo, como mostra o exemplo, a visualização depende da obtenção prévia de uma relação direta entre os grupos.

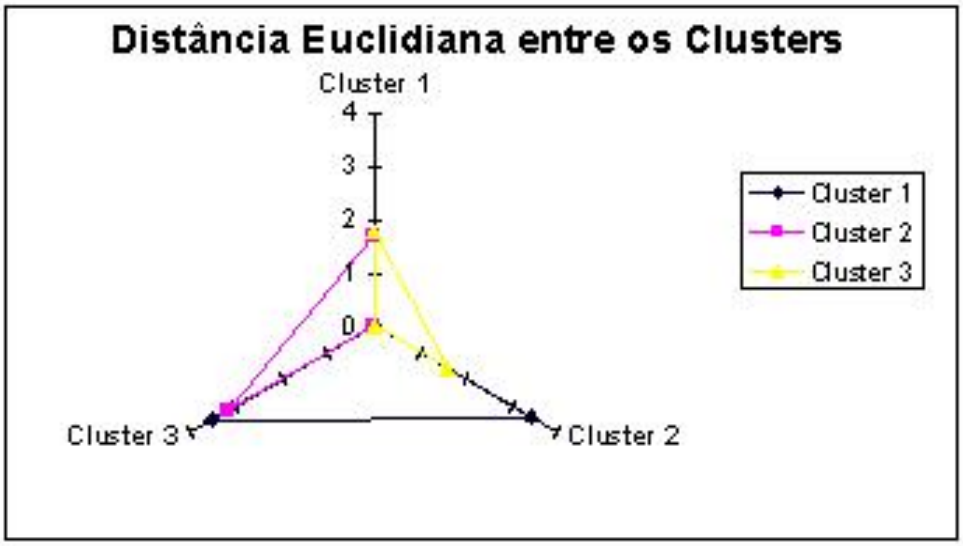


Figura 3.4: gráfico de radar, em escala logarítmica, de distância “intergrupos”.

3.1.4 – Gráfico de Colunas

Para uma comparação de forma visual entre os valores que representam o centro dos grupos, neste exemplo, foi utilizado um gráfico de colunas, que também permite a visualização dos dados ao longo de um período, organizando as categorias na horizontal e os valores na vertical. É possível, com auxílio deste recurso mostrado na Figura 3.5, verificar rapidamente os atributos que mais influenciam a formação de cada grupo.

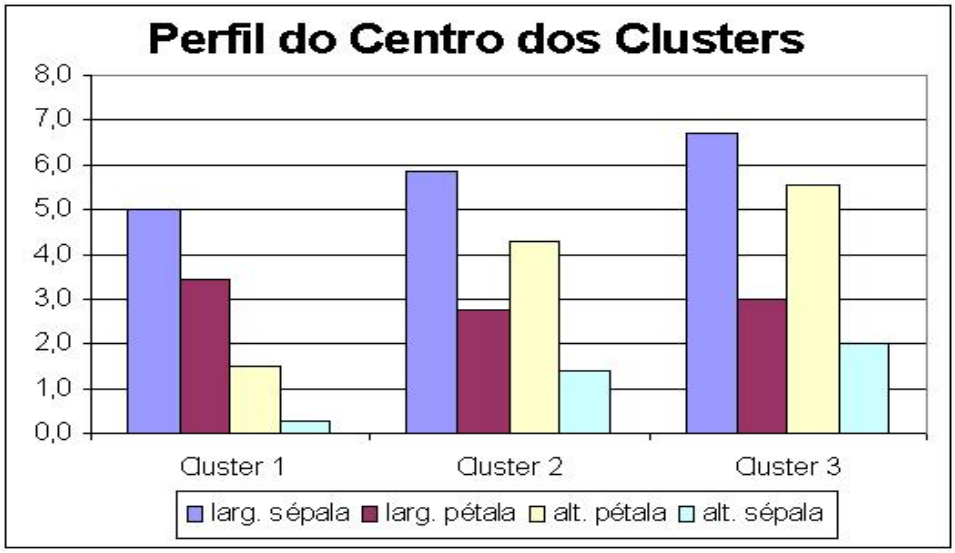


Figura 3.5: gráfico de colunas com o perfil do centro dos grupos.

3.2 – IBM *Intelligent Miner for Data*

Conjunto de funções destinadas à análise de grandes volumes de dados, o *Intelligent Miner* faz a comunicação entre as funções de pesquisa e pré-processamento numa maquina servidora e as ferramentas de visualização e administração numa máquina cliente.

O *Intelligent Miner* fornece diversas funções de mineração, tais como associação, classificação, grupo, previsão e estatísticas, para construção e aplicação de modelos de mineração sobre grandes bases de dados ou simples arquivos.

A função de mineração de grupos pode ser empregada sob implementação de dois algoritmos diferentes. Os gráficos a seguir exemplificam a exibição dos grupos produzidos pelas funções de mineração empregando os algoritmos de agrupamento demográfico – Figura 3.6 e agrupamento neural - Figura 3.7.

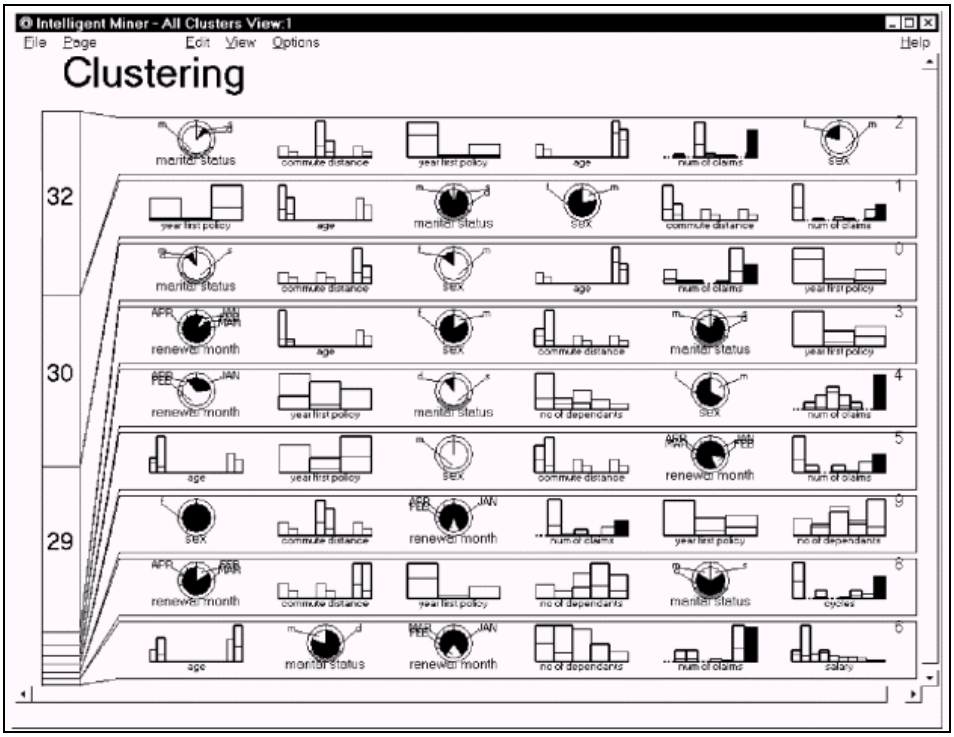


Figura 3.6: visualização dos agrupamentos demográficos. Baseada em [25].

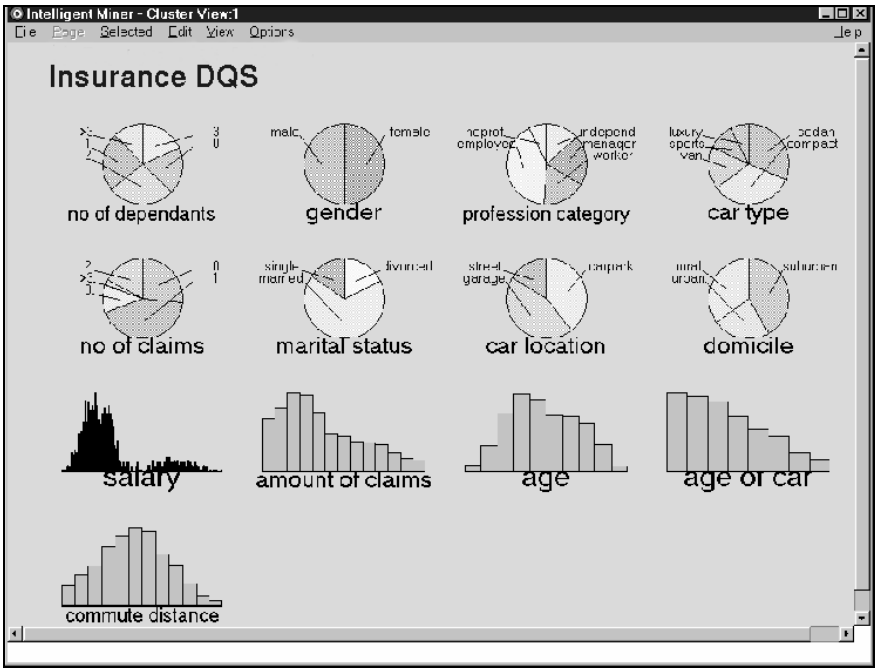


Figura 3.7: visualização dos agrupamentos neurais. Baseada em [25].

A função de mineração por agrupamento demográfico oferece ainda outras opções de resultados gráficos, como visto na Figura 3.8, que apresentam características mais pormenorizadas do resultado do processo de segmentação. Na Figura 3.9, é possível observar o detalhamento que o *software* oferece para cada grupo, tomando por base o primeiro grupo mostrado no gráfico que o antecede.

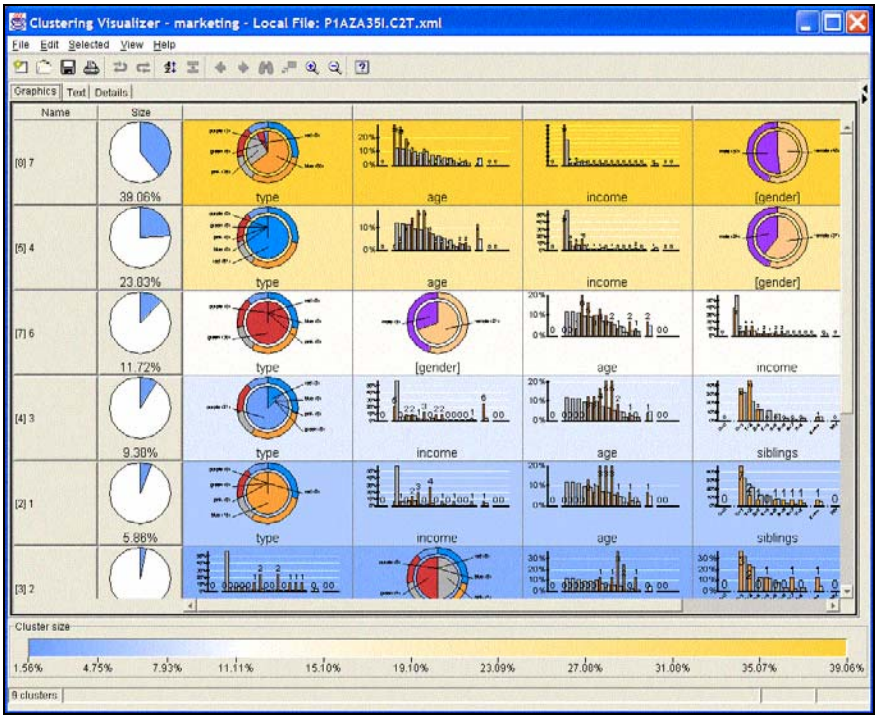


Figura 3.8: principais características dos grupos formados. Baseada em [25].

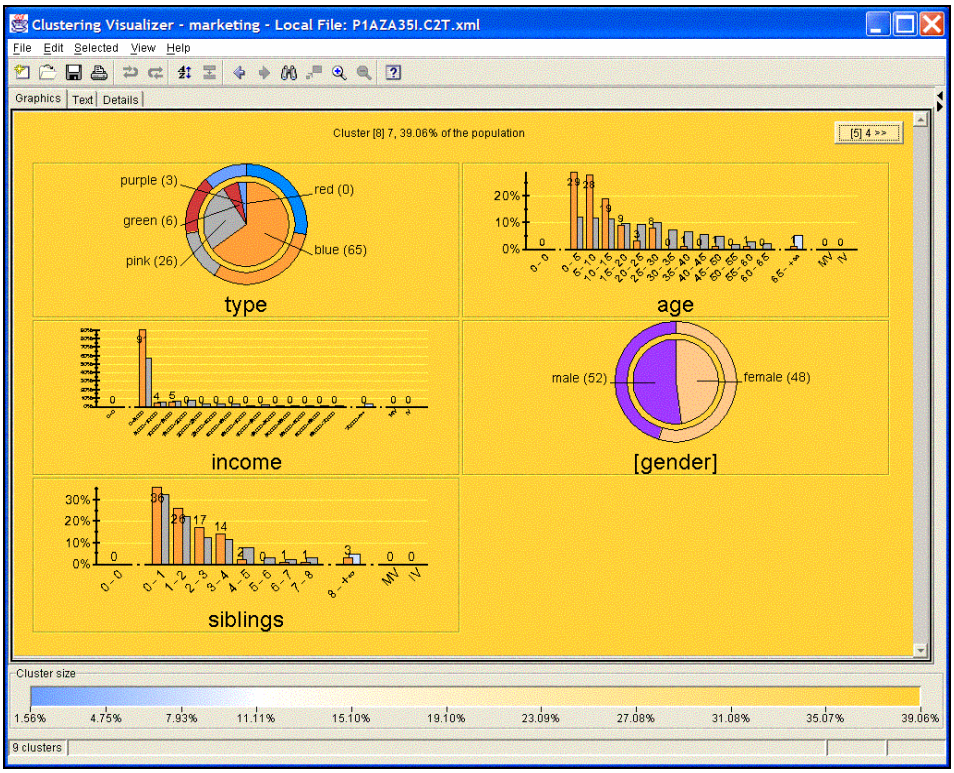


Figura 3.9: detalhamento do primeiro grupo. Baseada em [25].

Como mais um recurso de análise de agrupamentos, o resultado obtido pela aplicação da função estatística “analisar” também permite a visualização de um gráfico específico, como pode ser visto na Figura 3.10.

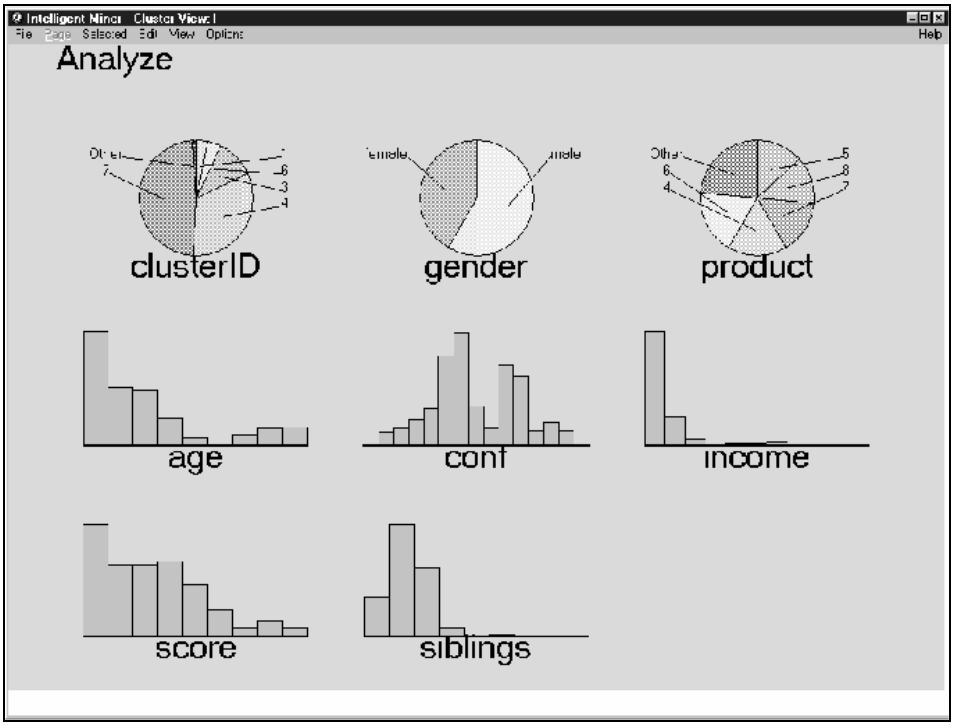


Figura 3.10: apresentação gráfica da função estatística analisar. Baseada em [25].

O *Intelligent Miner* também oferece visualizadores de associações, de classificações e de grupos. Cada um deles pode apresentar o mesmo modelo em exibições diferentes, cada uma delas contendo informações que não estão disponíveis ou têm dificuldades para serem representadas nas outras. Sendo de interesse específico para este trabalho, serão apresentados os três modos de visualização do visualizador de grupos: “Gráficos”, “Texto” e “Detalhes”. Na Figura 3.11, pode ser visto um exemplo de exibição do modo “Gráficos”.

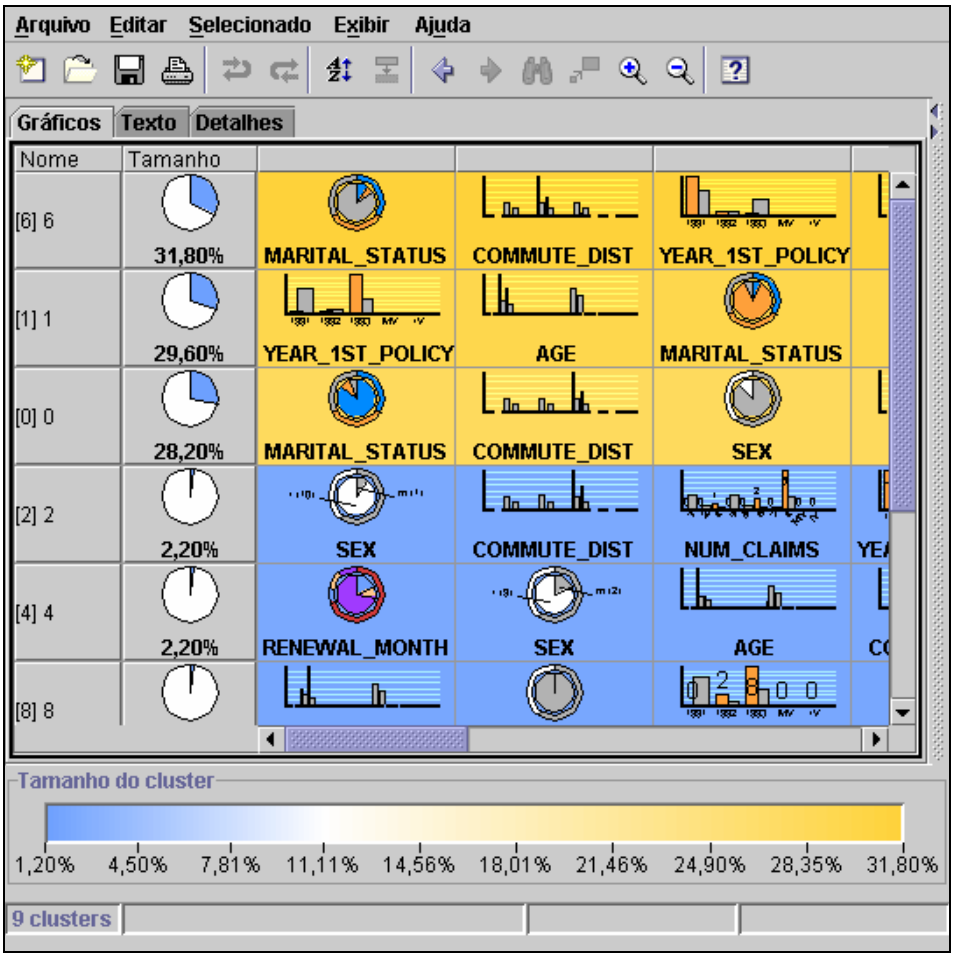


Figura 3.11: modo Gráficos do visualizador de grupos. Baseada em [25].

O modo “Gráficos” ainda permite opções de visualização, quando simples atributos de sua composição sejam expandidos para uma apreciação em maiores detalhes, como visto na Figura 3.12(a) e na Figura 3.12(b), que apresentam um determinado campo categórico, “*marital status*”, ajustado ao tamanho da janela ou exibido em forma de tabela, respectivamente.

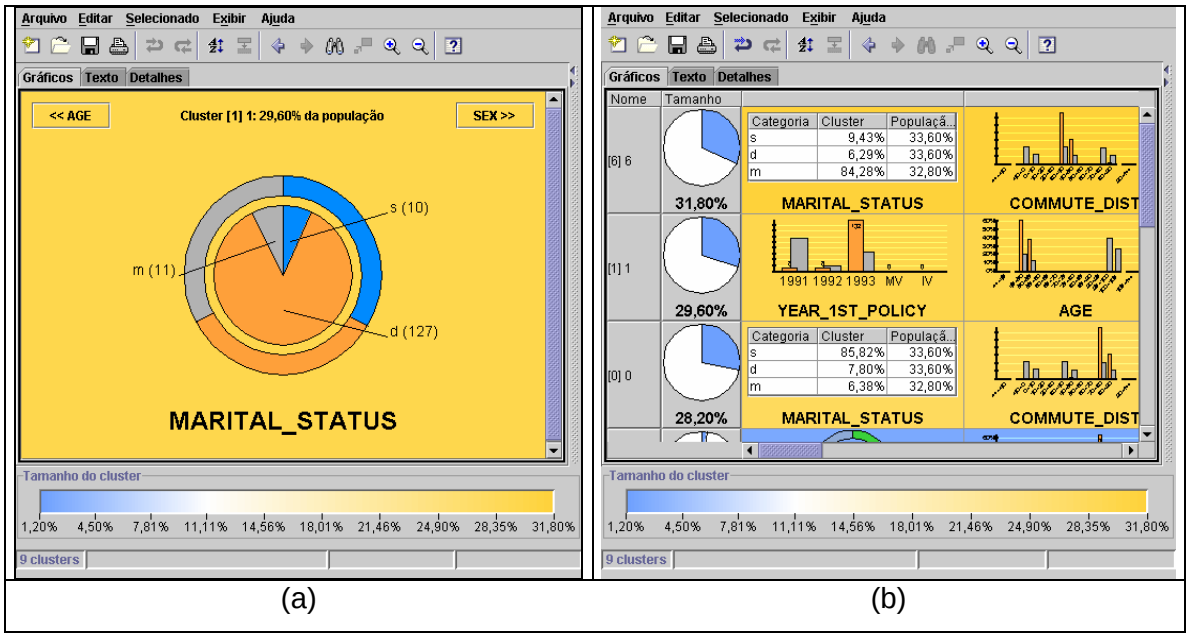


Figura 3.12: exibição de atributo ajustada à janela e em tabelas. Baseada em [25].

O visualizador de grupos em modo “Texto” exibe um resumo descritivo textual da composição dos grupos formados pelo software, como pode ser observado na Figura 3.13.

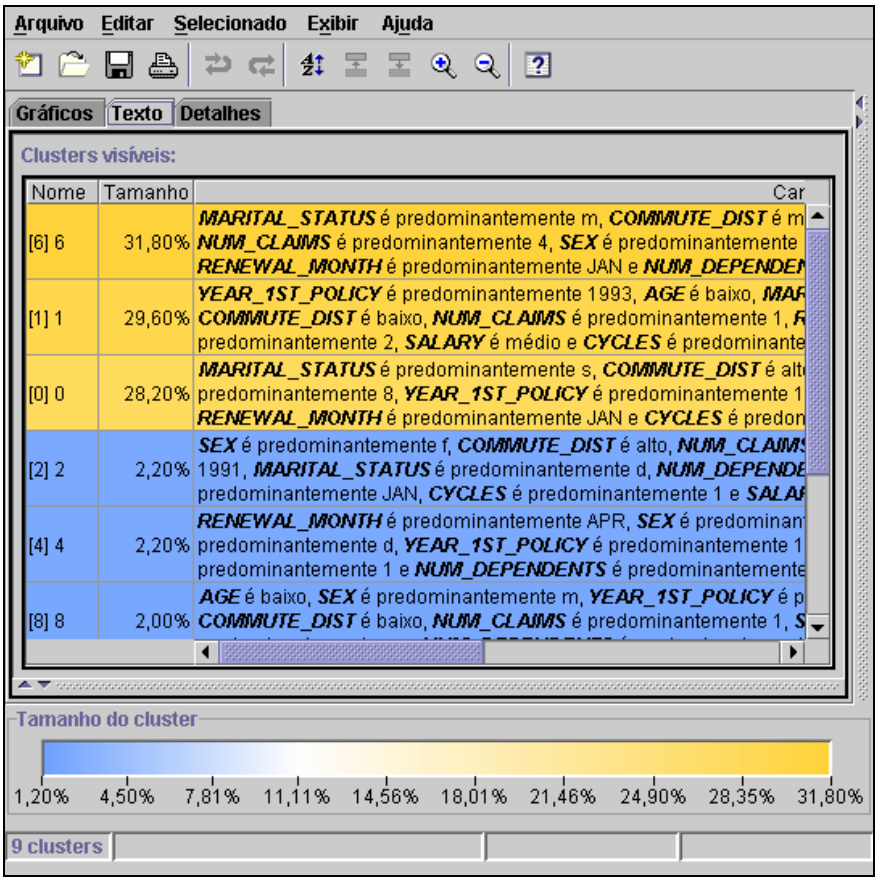


Figura 3.13: modo Texto do visualizador de grupos. Baseada em [25].

O modo Detalhes do visualizador de grupos apresenta os grupos separadamente, fornecendo detalhes da sua composição no formato de um formulário, que contém informações sobre os atributos, tais como nome, tipo e valores estatísticos. Este modo de exibição está exemplificado na Figura 3.14.

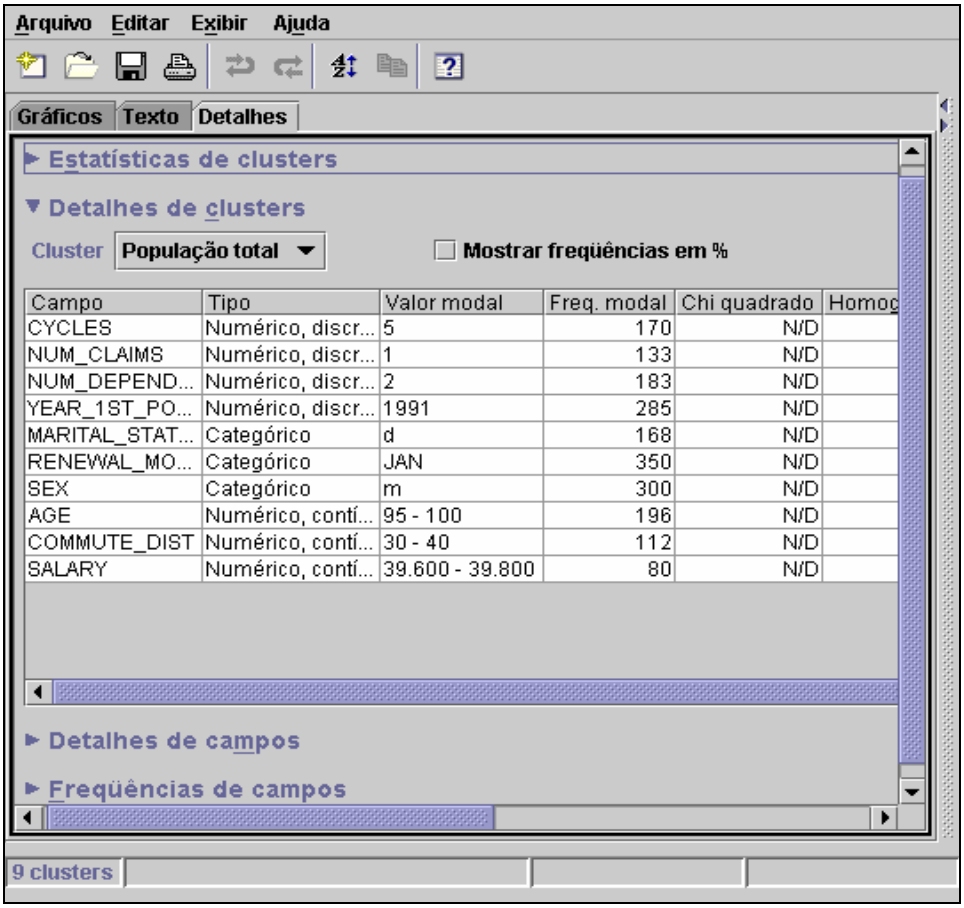


Figura 3.14: modo Detalhes do visualizador de grupos. Baseada em [25].

3.3 – SAS *Enterprise Miner*

Ferramenta de mineração e visualização de dados desenvolvida pelo SAS Institute, é de grande ajuda no estudo do comportamento de conjuntos de dados, atuando tanto na segmentação, efetuada através de diversos métodos, quanto na obtenção de modelos de predição.

A utilização do software dá-se pela escolha de uma entre as muitas funções de mineração disponíveis, por meio de um diagrama de fluxo na forma de árvore, como pode ser visto na Figura 3.15, onde cada nó é capaz de habilitar funções específicas.

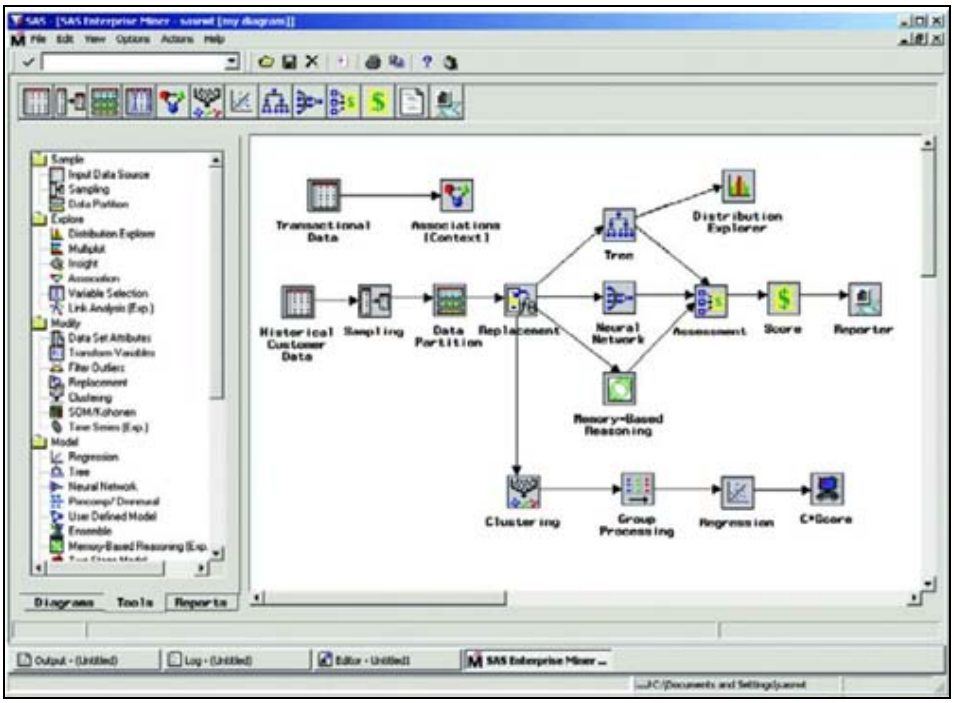


Figura 3.15: diagrama de fluxo do *Enterprise Miner*. Baseada em [25].

Enquanto é exibido o diagrama de fluxo, há uma segunda forma de apresentação dos nós, em que estes apresentam-se agrupados conforme uma metodologia conhecida pela sigla SEMMA, onde cada letra representa a inicial de cada uma das palavras que denotam os 5 tipos de funcionalidade: *Sample, Explorer, Modify, Model, Access*.

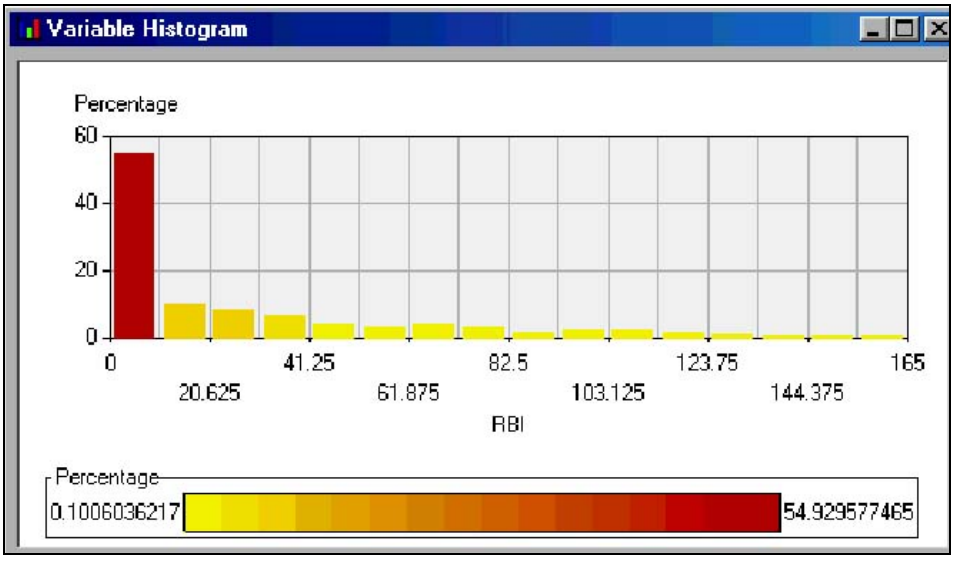


Figura 3.16: histograma de variável (atributo). Baseada em [25].

Utilizando-se uma das opções oferecidas pelo diagrama de fluxo, pode-se chegar a diversas formas de visualização, como um histograma de distribuição de um determinado atributo, apresentado na Figura 3.16 ou uma janela de resultados da

segmentação efetuada pelo software, contendo um conjunto de gráficos que apresentam a composição dos grupos formados, apresentada na Figura 3.17.

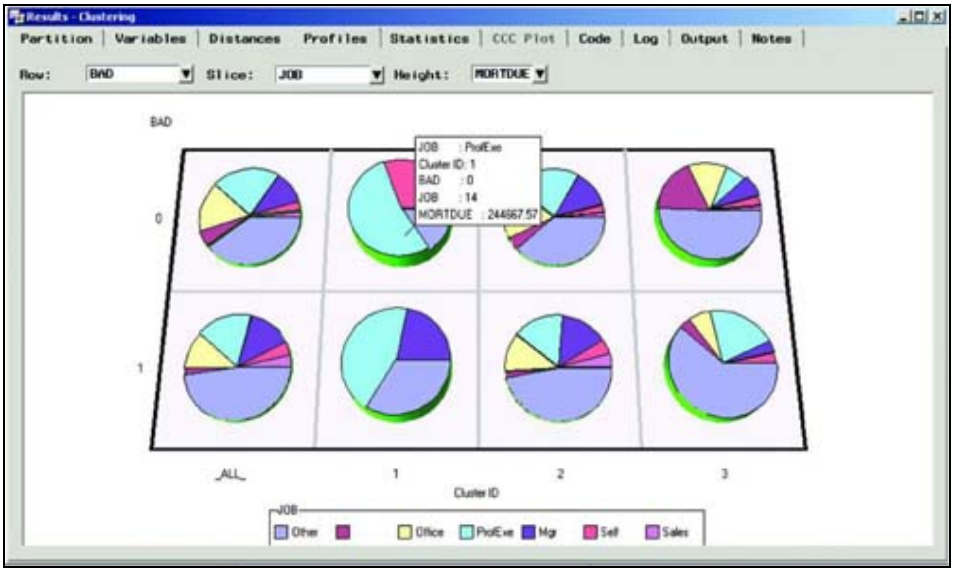


Figura 3.17: resultado da segmentação. Baseada em [25].

3.4 – *Mineset Enterprise Edition*

O *Mineset* é um produto da *Silicon Graphics*, Inc, desenvolvido para auxiliar no descobrimento de modelos ocultos, tendências mais importantes e novos conhecimentos.

O software traz ferramentas que possibilitam a mineração, análise e visualização dos dados, facilitando a exploração de conjuntos de dados, descobrindo padrões automaticamente e a aplicação dos diversos recursos de visualização colaboram decisivamente no entendimento do comportamento da base, ou até de alguma variável em detalhe. As principais tarefas envolvidas no processo de descoberta de conhecimento, como identificação e preparação dos dados e construção, avaliação e exploração dos modelos estão contempladas no *Mineset*, inclusive com possibilidade de visualização gráfica.

Dentre as técnicas de visualização disponíveis no *Mineset*, algumas podem ser destacadas pelo seu relacionamento com as demais técnicas exploradas no presente trabalho.

O recurso de visualização dispersa, ou *scatter visualizer*, possibilita que os dados sejam exibidos em um plano de até três dimensões, além de oferecer recursos extras como cor, tamanho, forma e animação. Cada ponto individual do gráfico corresponde a um objeto da base. Esta visualização funciona bem quando o número de objetos é menor que 50.000 ou quando algum processamento reduziu a base a um pequeno conjunto de grupos [25]. A Figura 3.18 apresenta o visualizador disperso exibindo uma paisagem tridimensional, com os objetos separados em 3 grupos, identificados por cores diversas.

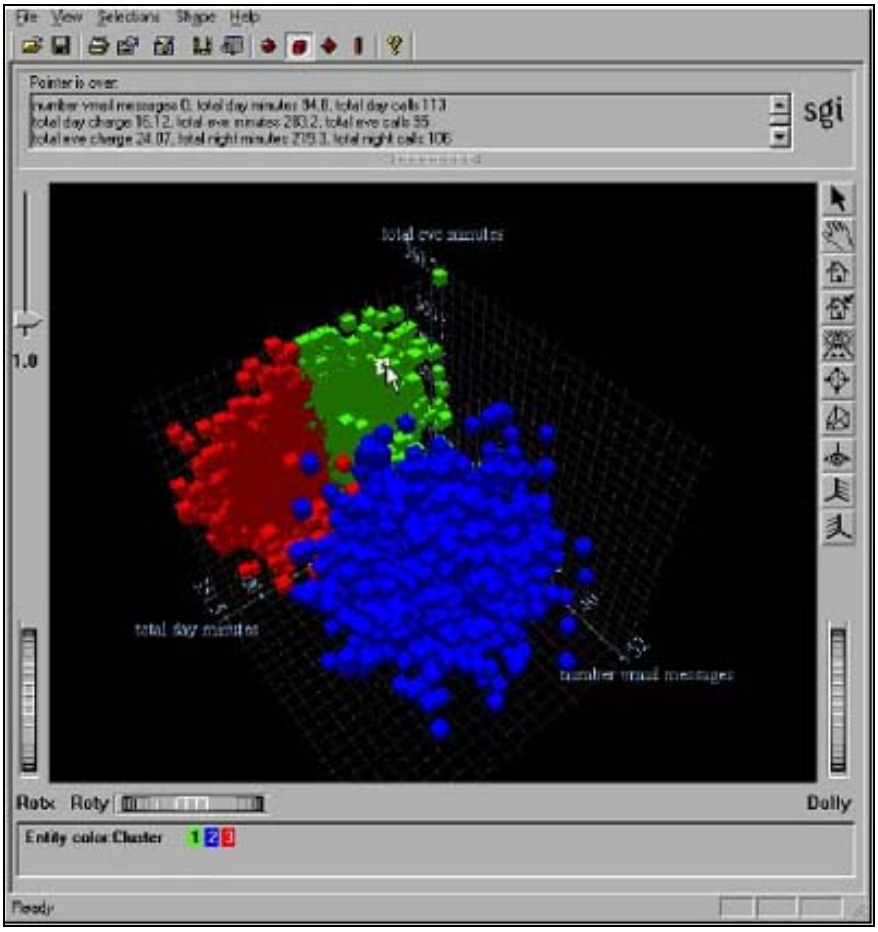


Figura 3.18: visualizador disperso (scatter visualizer). Baseada em [25].

O visualizador de árvore, apresentado na Figura 3.19, é uma interface gráfica que mostra os dados numa paisagem tridimensional, apresentando-os em forma de uma árvore hierárquica. Cada nível dos ramos refere-se a um valor de atributo diferente. Cada nó da árvore mostra um quadro representando todos os dados nas ramificações abaixo dele. O quadro é composto basicamente de um bloco cujas altura e cor dependem dos atributos dos dados [26].

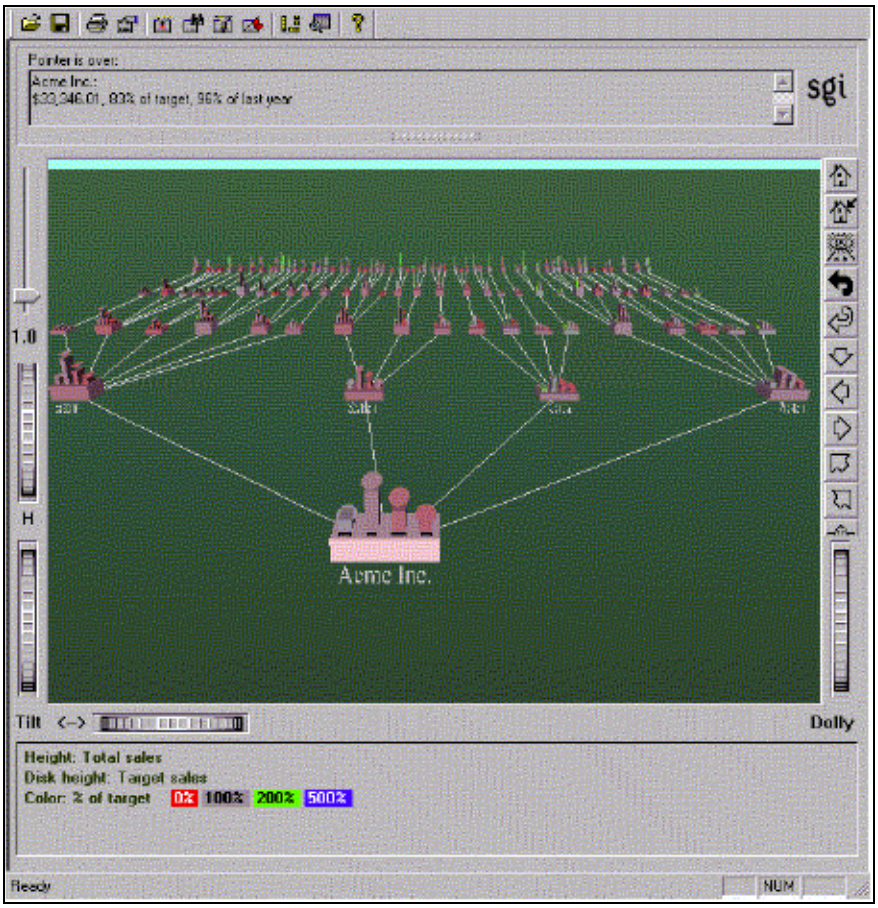


Figura 3.19: visualizador de árvore. Baseada em [26].

O visualizador estatístico oferece formas de representação diferenciadas, de acordo com a natureza dos atributos. Variáveis categóricas, que representam valores como sim/não, nome, origem, etc, aparecem em um histograma onde há uma coluna para cada valor, com a indicação do número de objetos que possuem igual valor para o atributo em questão. Para os demais atributos, a exibição é feita com uma coluna que representa o atributo, acompanhada de informações estatísticas que incluem valores máximo, mínimo, mediano e médio. O gráfico mostra, ainda, a concentração de valores nos quartis percentuais de 25% e 75%, permitindo a comparação da distribuição dos valores do atributo em relação ao conjunto de objetos da base. Os resultados da aplicação deste visualizador estão representados na Figura 3.20.

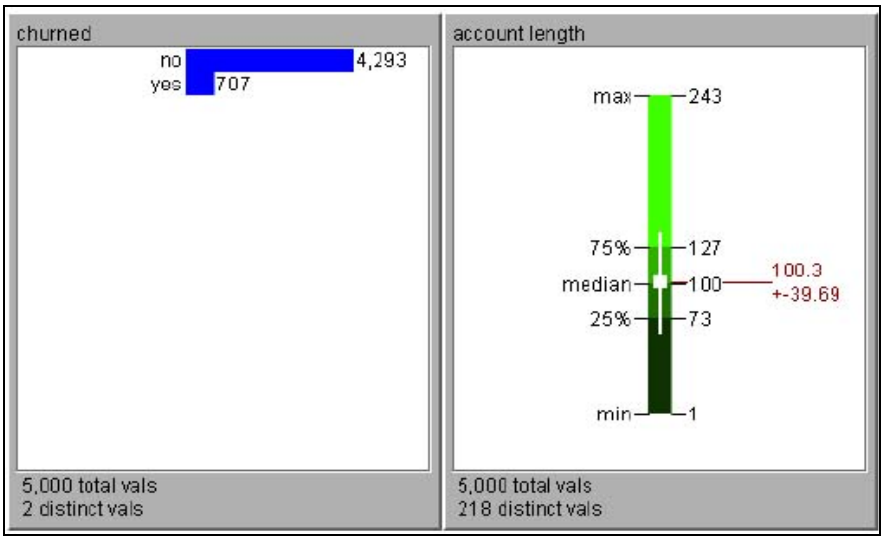


Figura 3.20: visualizador estatístico para a base de dados. Baseada em [25].

Na Figura 3.20, resultado da aplicação do *Mineset* sobre uma base de 5000 clientes, algumas conclusões são facilmente extraídas, como por exemplo, o atributo "*churned*" é categórico, com 707 ocorrências do valor *yes* e 4293 ocorrências do valor *no*; o atributo *account length* é do tipo contínuo, com 218 valores distintos, entre 1 e 243, sendo a mediana igual a 100 e o desvio padrão de ± 39,69. 25% dos atributos de menor valor está entre 1 e 73, 50% entre 73 e 127 e 25% restantes tem valor mínimo 127 e máximo 243.

Se a visualização é referente ao resultado do processo de segmentação da base de dados, a interpretação continua sendo feita da mesma forma, porém cada atributo tem sua representação estatística mostrada no âmbito de cada grupo formado pelo software. A Figura 3.21 apresenta os resultados do processo de agrupamento em três grupos, com foco no atributo *number vmail messages*. Dessa forma, torna-se possível analisar o comportamento de cada variável dentro dos grupos e a sua influência no processo de segmentação, por exemplo.

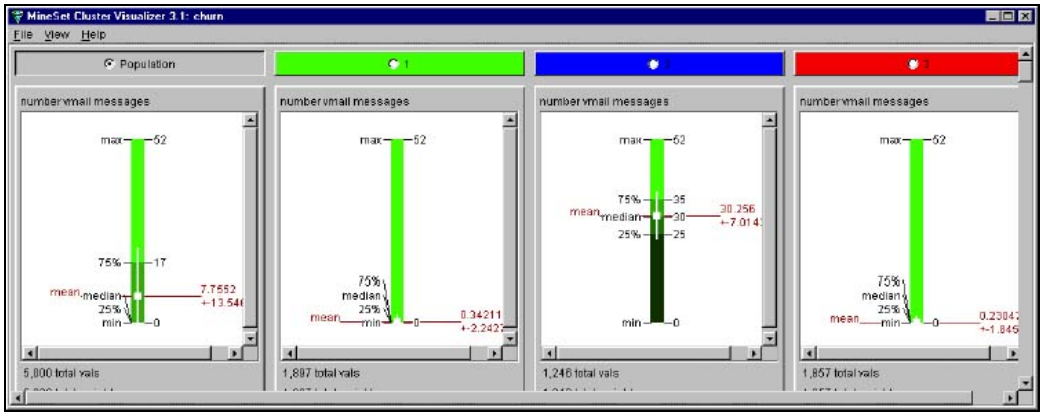
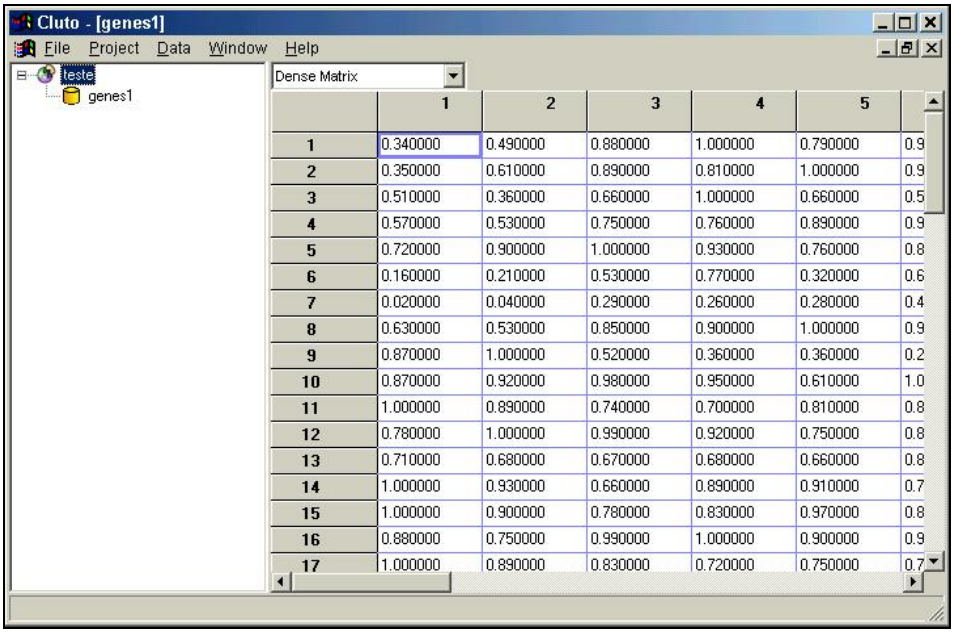


Figura 3.21: visualização estatística da segmentação da base. Baseada em [25].

3.5 – *gCLUTO - Graphical CLUstering TOolkit*

Desenvolvido na Universidade de Minnesota, o *gCLUTO* é uma interface que possibilita a utilização do potencial das funções de análise de dados da biblioteca CLUTO, de uma forma muito mais amigável e com algumas características próprias, tais como: gerenciador em forma de árvore de projetos que permite fácil percepção dos arquivos de dados, soluções de agrupamento e formas de visualização existentes e seus relacionamentos; formulário para rápida visualização dos valores dos dados, visualização de matriz e visualização de montanha, recurso bastante poderoso que permite a visualização em 3D, usando o critério de escala multidimensional (MDS – sigla em inglês).

O *gCLUTO* utiliza o conceito de "projeto" para organizar os dados e o fluxo de trabalho. Na Figura 3.22, é possível observar uma tela do *software*, com gerenciador em forma de árvore, onde cada ícone representa um item já adicionado ao projeto e uma janela com o formulário de dados, onde é possível ver os valores dos atributos (colunas) de cada objeto (linha) a ser analisado.



	1	2	3	4	5	
1	0.340000	0.490000	0.880000	1.000000	0.790000	0.9
2	0.350000	0.610000	0.890000	0.810000	1.000000	0.9
3	0.510000	0.360000	0.660000	1.000000	0.660000	0.5
4	0.570000	0.530000	0.750000	0.760000	0.890000	0.9
5	0.720000	0.900000	1.000000	0.930000	0.760000	0.8
6	0.160000	0.210000	0.530000	0.770000	0.320000	0.6
7	0.020000	0.040000	0.290000	0.260000	0.280000	0.4
8	0.630000	0.530000	0.850000	0.900000	1.000000	0.9
9	0.870000	1.000000	0.520000	0.360000	0.360000	0.2
10	0.870000	0.920000	0.980000	0.950000	0.610000	1.0
11	1.000000	0.890000	0.740000	0.700000	0.810000	0.8
12	0.780000	1.000000	0.990000	0.920000	0.750000	0.8
13	0.710000	0.680000	0.670000	0.680000	0.660000	0.8
14	1.000000	0.930000	0.660000	0.890000	0.910000	0.7
15	1.000000	0.900000	0.780000	0.830000	0.970000	0.8
16	0.880000	0.750000	0.990000	1.000000	0.900000	0.9
17	1.000000	0.890000	0.830000	0.720000	0.750000	0.7

Figura 3.22: tela do *gCLUTO* com árvore de projetos e formulário de dados.

Após a realização do agrupamento de dados, através de um dos quatro métodos disponíveis, a solução é exibida em uma janela de texto, como mostrado na Figura 3.23, contendo os parâmetros utilizados no processamento, informações estatísticas e características descritivas dos grupos formados pelo *software*.

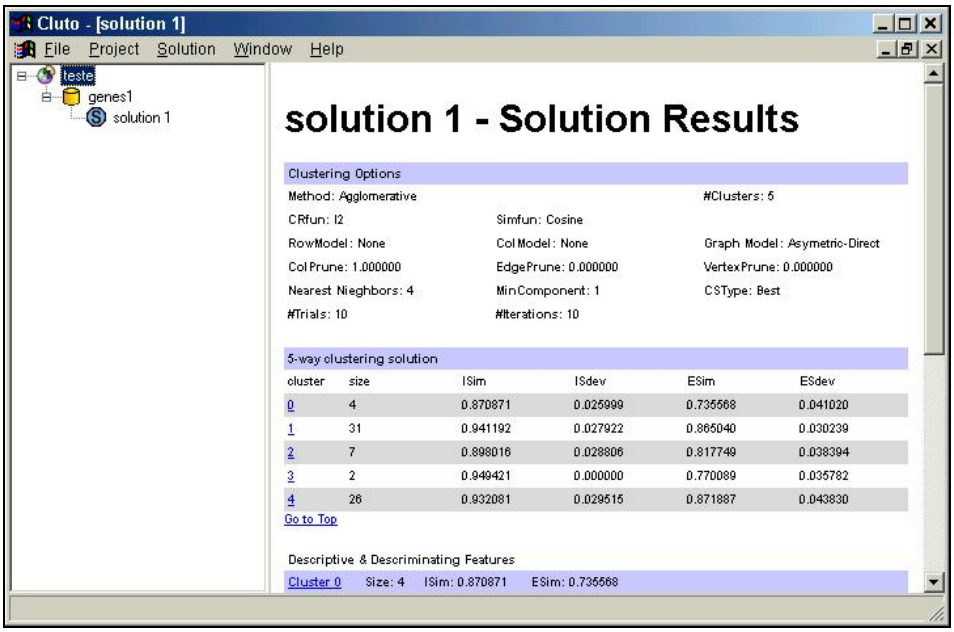


Figura 3.23: exibição da janela de texto da solução.

As duas formas de visualização gráfica podem ser obtidas após o final do processo de segmentação: visualização de matriz e visualização de montanha.

A visualização de matriz apresenta uma escala de cores para representar graficamente os dados de origem. A cor branca representa valores iguais ou muito próximos a zero; tons crescentes de vermelho representam valores positivos e tons crescentes de verde representam valores negativos, sendo os extremos no tom mais escuro. As linhas estão ordenadas de maneira que os objetos de um mesmo agrupamento apareçam juntos. Um divisor horizontal preto separa os grupos. A escala da matriz pode ser alterada interativamente, através dos controles de largura e comprimento. Este modo de visualização também disponibiliza dendrograma de linhas com a solução da segmentação e dendrograma de colunas, usando agrupamento aglomerativo no inverso da matriz. Tais árvores podem ser utilizadas para reduzir e tornar a expandir alguma área de interesse específico dentro da matriz. A Figura 3.24 exemplifica uma tela do *gCLUTO* contendo uma janela de visualização de matriz.

A visualização de montanha tem seu uso aplicado na comparação visual de semelhanças relativas entre os agrupamentos, tais como tamanho, semelhança interna e divergência externa. Nesta forma de visualização, cada grupo está representado como um pico 3D. A localização do pico, o volume, a altura e as cores são usados para retratar informações sobre o grupo a que se referem, como pode ser visto na Figura 3.25.

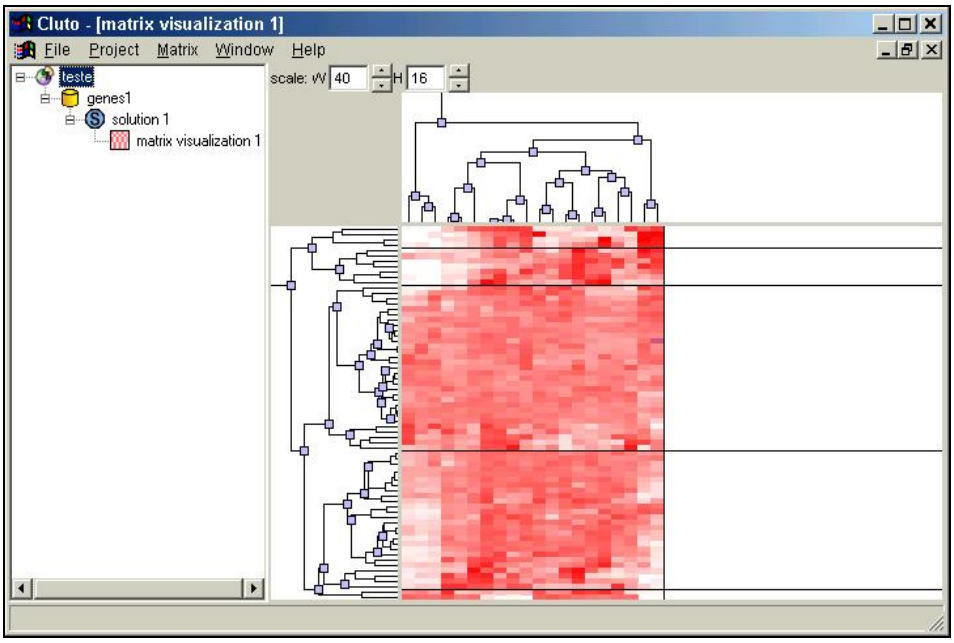


Figura 3.24: exibição da janela de visualização de matriz com dendrogramas.

A localização dos picos no plano é determinada pela aplicação de escala multidimensional sobre os pontos médios de cada grupo formado, que são considerados como vértices das montanhas, tentando preservar da melhor maneira possível as distâncias relativas entre os vértices.

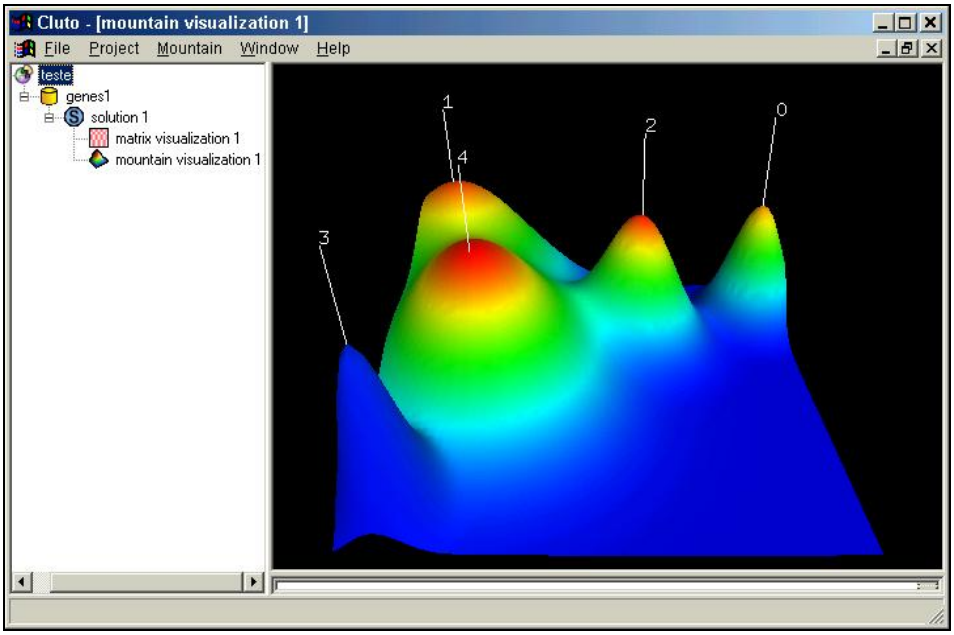


Figura 3.25: exibição da janela de visualização de montanha.

A escala multidimensional permite que os usuários do software possam tirar conclusões sobre seus dados utilizando a visualização de montanha. Observando-se o exemplo contido na Figura 3.25, onde um conjunto de dados foi segmentado em cinco

grupos, a visualização de montanha representa tal segmentação como cinco picos etiquetados com o identificador do grupo ao qual o pico se refere.

A forma de cada pico é uma curva gaussiana, usada como mera estimativa da distribuição dos dados dentro de cada grupo, sua altura é proporcional à semelhança interna do grupo, seu volume é proporcional ao número de elementos contidos no grupo e as cores são proporcionais à divergência interna: vermelho indica baixa divergência e azul, alta.

3.6 – “Coordenadas Estrela”

O principal objetivo do emprego de técnicas de visualização de dados é facilitar o entendimento da distribuição, estrutura e relacionamentos dos objetos de uma coleção. Através da observação de uma representação espacial dos dados, ainda que de forma virtual, um especialista pode, por exemplo, realizar prognósticos a cerca de uma segmentação, entender melhor o resultado obtido pela aplicação de uma técnica ou realizar desdobramentos mais aprofundados com base nos padrões descobertos.

Uma dessas técnicas, conhecida como “Coordenadas Estrela” [2], permite a visualização de objetos com diversos atributos, n dimensões, em um plano de duas dimensões, dando uma percepção virtual da distribuição espacial relativa dos objetos, a forma das bases, dos grupos e a posição de seus centros e, ainda, a possibilidade de explorar interativamente o relacionamento entre os atributos dos objetos e a forma de seus agrupamentos.

Basicamente, a técnica das “Coordenadas Estrela” consiste em distribuir os eixos coordenados dentro de um círculo, empregando um plano bidimensional, de forma que tenham a origem no centro e ângulos iguais entre si, mas não necessariamente são ortogonais. Cada eixo representa um atributo dos objetos da base e o valor mínimo encontrado para cada atributo, entre todos os objetos, é mapeado na origem e o valor máximo é mapeado no outro extremo do eixo coordenado, como demonstrado na Figura 3.26.

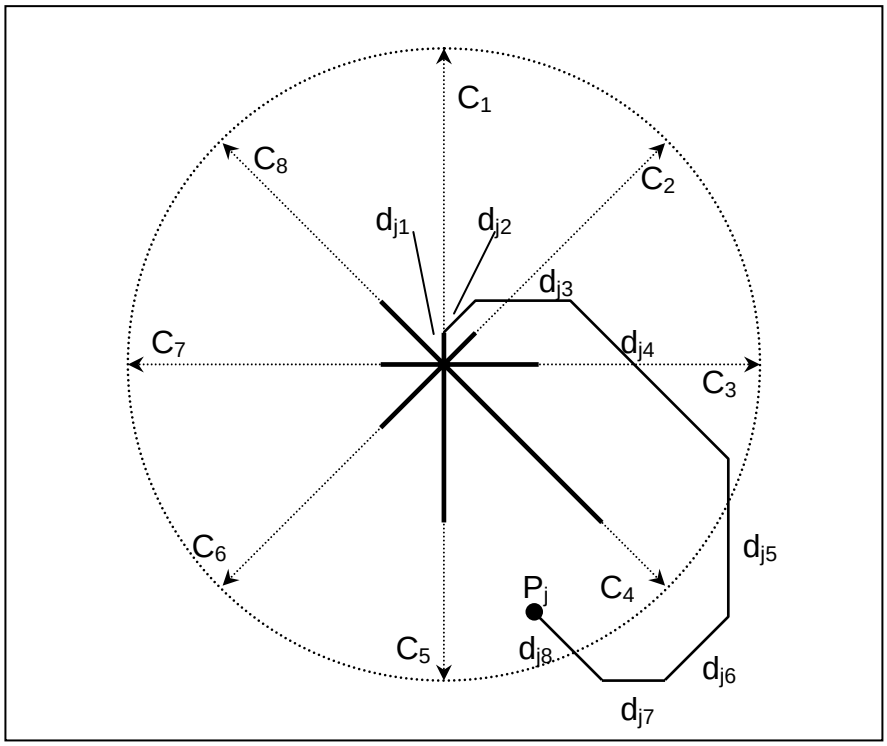


Figura 3.26: representação de um ponto de 8 dimensões em um plano bidimensional. Baseada em [2].

Cada ponto P_j representa um objeto da base, equivalendo a um vetor de n ocorrências, refletindo cada uma das linhas de uma matriz D [m x n], que é a coleção de todos os dados da amostra em análise.

A localização P_j (x,y) é determinada pelo vetor soma de todos os vetores unitários (uxi,uyi) em cada eixo Ci, multiplicado pelo valor do atributo do objeto relativo àquele eixo, descontado o valor mínimo, origem do eixo, como mostrado em (3.27).

$$P_j(x,y) = \left(\sum_{i=1}^{n} \vec{u}_{xi} * (d_{ji} - \min_i), \sum_{i=1}^{n} \vec{u}_{yi} * (d_{ji} - \min_i) \right) \tag{3.27}$$

onde u_i é dado por (3.28) e $\min_i$ e $\max_i$ são, respectivamente, o menor e o maior valores encontrados para o atributo *i* de todos os objetos da coleção.

$$u_i = \frac{\vec{C}_i}{\max_i - \min_i} \tag{3.28}$$

O método de “Coordenadas Estrela” permite visualizações simples em duas e três dimensões, mas acomoda espaços com mais dimensões, sobrepondo-as através de eixos coordenados não ortogonais. A sobreposição, porém, produz algumas distorções, como por exemplo, a localização de um ponto no espaço não identificar

unicamente seus valores em cada eixo coordenado. Apenas com a ajuda de transformações interativas dinâmicas, como rotações e translações comandadas pelo usuário, é possível entender melhor a real distribuição dos objetos.

Como visto, o comprimento de cada eixo é dado pela diferença entre o valor mínimo e o valor máximo de cada atributo *i* de todos os objetos da coleção, ou seja, o eixo coordenado e, por conseguinte, a posição espacial dos objetos, é diretamente ligada à escala dos valores dos atributos. Assim, é possível alterar o peso relativo de cada atributo, alterando a escala do eixo que o representa. Implementando este conceito em um procedimento interativo, têm-se a opção de alterar a contribuição de um determinado atributo a fim de verificar a sua importância relativa e até mesmo observar diferentes formações de agrupamentos, na presença de uma exacerbação ou ausência total de um eixo.

A alteração do ângulo de um eixo em relação aos demais, através da rotação em torno de sua origem, é a rotação do próprio vetor unitário deste eixo, o que altera a correlação entre os atributos, mudando assim a direção da contribuição do atributo na distribuição dos objetos.

Resumidamente, podemos apresentar os passos que levam à determinação das coordenadas de um objeto multidimensional em um espaço R^2, da seguinte forma:

1) é calculado o valor inicial dos ângulos entre os eixos, simplesmente dividindo-se o círculo pelo número de dimensões (atributos): 360 / número de atributos;

2) para cada dimensão dos objetos da base, é calculado o seno e o cosseno do ângulo de cada eixo, considerando o primeiro atributo como eixo-origem;

3) o vetor unitário é obtido pela razão entre o número seqüencial do eixo (1, 2, 3, ..., n) e a diferença entre o valor máximo do atributo referente e o valor mínimo assumido por este atributo, considerando a totalidade de objetos;

4) multiplicando-se o vetor unitário pelo cosseno e pelo seno obtidos anteriormente, encontra-se a inclinação do eixo;

5) o somatório em cada dimensão x e y do espaço R^2, do produto da inclinação do eixo pela diferença entre o valor do atributo e o valor mínimo assumido por este atributo, considerando a totalidade de objetos, representa a posição P_j (x,y) do objeto P_j.

3.7 – Dendrograma

Os métodos hierárquicos de segmentação criam uma decomposição hierárquica da base de dados, representada por uma árvore de grupos. O dendrograma é um método hierárquico aglomerativo e tem a origem de seu nome, muito justamente, na palavra *dendrology*, (dendrologia), ramo da botânica que estuda as árvores. Em tais hierarquias, cada nó da árvore representa um grupo de objetos.

Segundo HAIR [27], dentre os mais populares métodos hierárquicos aglomerativos utilizados em segmentação de dados, podem ser citados:

Simple Linkage: a segmentação é construída com base na distância mínima entre dois objetos (vizinho mais próximo).

Complete Linkage: a segmentação é construída com base na distância máxima entre dois objetos (vizinho mais afastado).

Average Linkage: a segmentação é construída com base na distância média tomada a partir de todos os objetos de um grupo em relação aos objetos de outro grupo.

Ward's Method: que utiliza a abordagem da análise de variância para avaliar a distância entre os grupos.

Centroid Method: utiliza a distância entre os centróides pra realizar a segmentação.

No presente trabalho, foi implementado o método *Simple Linkage*, ou Vizinho Mais Próximo, detalhado a seguir, de acordo com os passos sugeridos em [3].

Como primeiro passo, é calculada a dissimilaridade entre todos os objetos a classificar, comparados dois a dois. Uma tabela D(n,n) quadrada de ordem igual ao número de objetos da base de dados é necessária para guardar cada uma dessas distâncias calculadas. Supondo ser aceitável considerar que a distância entre dois grupos que contém apenas um objeto é igual à distância entre os próprios objetos, então $d_{(\{x\},\{y\})} = d_{(x,y)} \ \forall x, y \in I$.

Ao final da montagem da tabela, todas as ocorrências da diagonal principal são nulas, pois, que se d_{ij} é uma distância, então $d_{(\{x\},\{x\})} = d_{(x,x)} = 0 \ \forall x \in I$

No segundo passo é pesquisada a tabela D(n.n) em busca da ocorrência extra-diagonal mínima, ou seja $d_{(\{x\},\{y\})} = d_{(x,y)}$ mínimo, formando uma novo grupo, que aglutina esses dois objetos {x} e {y}.

A iteração recomeça no primeiro passo, sendo que agora o espaço de busca na tabela é n-1, uma vez que dois objetos estão unidos em um só grupo.

Para calcular a tabela D'(n-1,n-1) correspondente à nova situação foi utilizada a estratégia de agregação do vizinho mais próximo, como critério para calcular a distância entre um grupo que contém dois objetos e os grupos unitários restantes,.

A estratégia do vizinho mais próximo consiste em adotar como distância entre os grupos remanescentes e o novo agrupamento formado a menor entre as distâncias apontadas pelos dois objetos que o formam, ou seja, será eleita como distância do novo grupo {x; y} para um grupo {z} a menor entre $d_{(\{x\},\{z\})}$ ou $d_{(\{y\},\{z\})}$.

Em cada iteração t do processo de agregação, a tabela D(n-t, n-t) é construída com a distância ultramétrica dada por (3.29).

$$d_{(\{x,y\},\{z\})} = Min(d_{\{x,z\}}; d_{\{y,z\}}) \qquad (3.29)$$

Apesar de possuir um pequeno inconveniente ao apresentar uma tendência a um "efeito de encadeamento", é grande a vantagem do emprego deste método devido à simplicidade e à economia computacional, pois fica dispensado o cálculo de todas as distâncias componentes da tabela D(n-t,n-t) a cada iteração, uma vez que só há necessidade de atualizar a distância relativa ao novo grupo. As distâncias entre os demais, obviamente permanecem inalteradas, e, ainda, o cálculo das distâncias é feito apenas na primeira vez.

Para ilustrar o desenvolvimento do método, podemos utilizar um simples exemplo numérico, como a seguir, onde a Tabela A representa uma tabela de dados T(n,p) com 4 objetos (A, B, C, D) e seus dois atributos (X, Y).

A) Tabela de dados T(n,p)

	X	Y
A	1	2
B	2	1
C	3	5
D	3	3

B) Tabelas de distâncias D(n-t, n-t)

B.1

	A	B	C	D
A	0	1,41	3,61	2,24
B	1,41	0	4,12	2,24
C	3,61	4,12	0	2,00
D	2,24	2,24	2,00	0

B.2

	(A,B)	C	D
(A,B)	0	3,61	2,24
C	3,61	0	2,00
D	2,24	2,00	0

B.3

	(A,B)	(C,D)
(A,B)	0	2,24
(C,D)	2,24	0

Para preenchimento da tabela inicial B.1, é calculada a Distância Euclidiana, entre cada objeto, utilizando-se (3.30).

$$d_{(i,j)} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (3.30)$$

Como a menor distância verificada foi entre os objetos B e C, com eles é formado um novo grupo e sua distância para os demais é a menor entre seus objetos e os demais, formando a tabela B.2.

As iterações quando restam apenas dois grandes grupos ligados, mas, logicamente, separados por uma determinada distância, como pode ser visto na tabela final B.3.

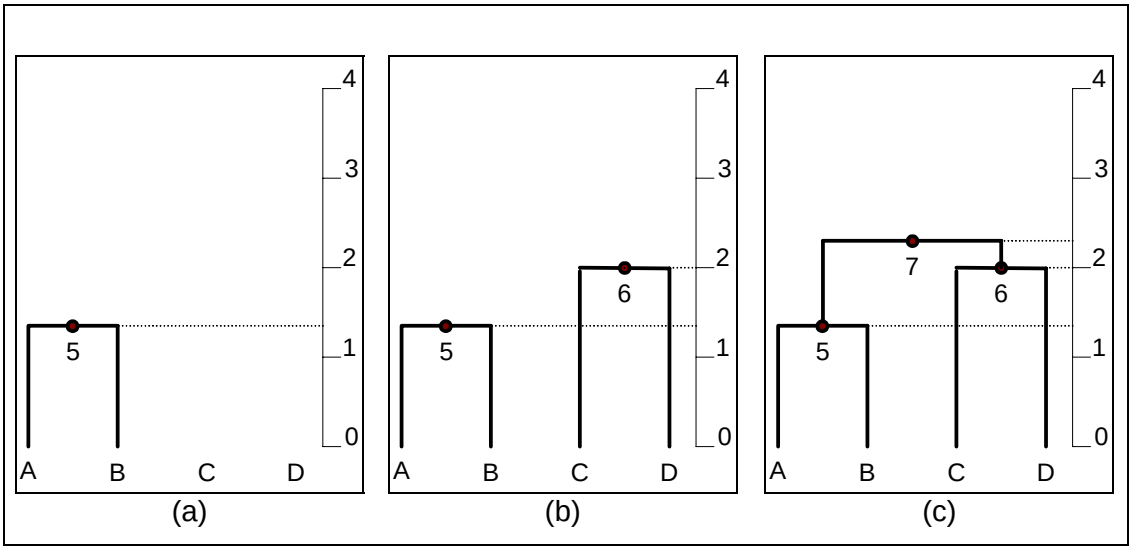


Figura 3.27: representação gráfica das agregações.Baseada em [3]. (a) agregação dos objetos A e B, que apresentam a menor distância entre todos os componentes da coleção, originando um nó (novo "objeto") 5. Este nó é marcado verticalmente em função do valor da distância entre os objetos agregados; (b) em seguida são agregados os objetos mais próximos; (c) ao final, todos os objetos estão agregados, ainda que em diversos níveis.

CAPÍTULO 4
IMPLEMENTAÇÃO DE UM PROCEDIMENTO

4.1 Introdução

Com o objetivo de verificar a validade e viabilidade de aplicação prática de alguns métodos explorados neste trabalho, foi desenvolvido um aplicativo, ao qual foi dado o nome de *SearchCluster*, que permite a descoberta de um número inicialmente desconhecido de grupos a partir da leitura de arquivos de dados em formato texto.

O aplicativo tem uma interface gráfica amigável que permite ao usuário explorar e analisar os dados optando, interativamente, pela aplicação dos métodos particional e hierárquico.

Para implementação do método particional, foi desenvolvido um algoritmo baseado no método *"K-Means"* para segmentação da base de dados em um número crescente e sucessivo de grupos, com o uso da Distância Euclidiana quadrática para determinação da similaridade entre os objetos, e no Critério de *Calinski* e *Harabasz*, para determinação do número de grupos que originaram a melhor distribuição de objetos. Para visualização da distribuição espacial relativa dos objetos, antes ou após a segmentação, a forma dos grupos e a posição de seus centros em um plano de duas dimensões também foi implementada a técnica "Coordenadas Estrela".

A aplicação do método hierárquico, em sua forma aglomerativa, é feita através de dendrogramas, método onde os objetos ou grupos são representados graficamente com uma estrutura em árvore, semelhantemente a uma classificação taxonômica. No desenvolvimento desta representação, foi utilizado o método conhecido como Vizinho Mais Próximo.

Se desejado, o software apresenta o valor do erro e da função objetivo calculados para cada seqüência de grupos, gera arquivos de saídas com informações detalhadas sobre os grupos e também sobre os objetos, identificando, inclusive a qual grupo cada um deles pertence. O usuário pode contar ainda com a exibição de uma tela com detalhes do agrupamento ótimo, tais como número de objetos e erro médio de cada grupo.

4.2 Arquitetura

Empregando-se a linguagem de programação *Microsoft Visual Basic 6.0*, desenvolveu-se um aplicativo padrão *Microsoft® Windows*, chamado *SearchCluster*, com uma interface amigável, que permite ao usuário pesquisar e analisar bases de dados de diversas configurações em formato texto.

O aplicativo permite ao usuário escolher a base de dados contida em quaisquer dos subdiretórios das unidades de disco da máquina, que será analisada e também indicará se é de seu desejo que sejam gerados arquivos de saída com os resultados finais do processo de agrupamento dos dados.

É responsabilidade do usuário indicar o número de atributos dos objetos da base, o número mínimo e o número máximo de grupos que deverão ser testados a fim de obter-se a formação ótima.

É possível utilizar o aplicativo após a indicação do número de atributos, a visualização da distribuição espacial dos objetos, quando é empregada a técnica conhecida como “Coordenadas Estrela” e também o dendrograma formado pelos objetos em sua conformação hierárquica.

O *software* é composto com base em dois formulários (janelas) centrais: o principal, com os módulos de controle do processo e apresentação dos seus resultados e o gráfico, para exibição das técnicas de visualização. Outros três formulários acessórios possibilitam a indicação das características da base de dados e outras opções do usuário, como geração de saída e alteração arbitrária de peso de quaisquer atributos; a apresentação de um resumo dos grupos formados pela melhor segmentação e a visualização do histograma obtido pelo emprego da técnica hierárquica aglomerativa.

Apesar da linguagem *Visual Basic* não originar *softwares* com grande velocidade na exibição de gráficos com alta quantidade de pontos na tela, a mesma permite a construção de aplicativos com interface bastante amigável e intuitiva, dentro do padrão Windows, que é largamente conhecido, facilitando a sua utilização. Além disso, a leitura dos arquivos texto, com os dados de entrada, é feita de forma direta, sem necessidade de outras ferramentas, e muito rapidamente, mesmo para bases contendo grande quantidade de objetos.

4.3 Utilização

A tela inicial do *SearchCluster*, da maneira que é apresentada ao usuário, pode ser vista na Figura 4.1, contendo o menu de opções e quatro botões de ação, estando habilitado somente aquele que permite cancelar o processamento e encerrar o software. Podem ser notadas duas caixas de texto: a superior destina-se à apresentação de um resumo dos parâmetros fornecidos ao aplicativo, tais como nome do arquivo de entrada, número de atributos, faixa de grupos para teste e outras disponíveis na tela de opções, além do erro e função objetivo calculados para todos os agrupamentos testados, se o usuário assim o desejar. A inferior apresentará, em destaque, o resultado do melhor agrupamento obtido no processamento. Abaixo desta caixa, há uma barra animada que permite o acompanhamento visual da progressão do processamento com uma pequena caixa de texto à direita que apresenta o número de grupos que foram testados no último passo concluído.

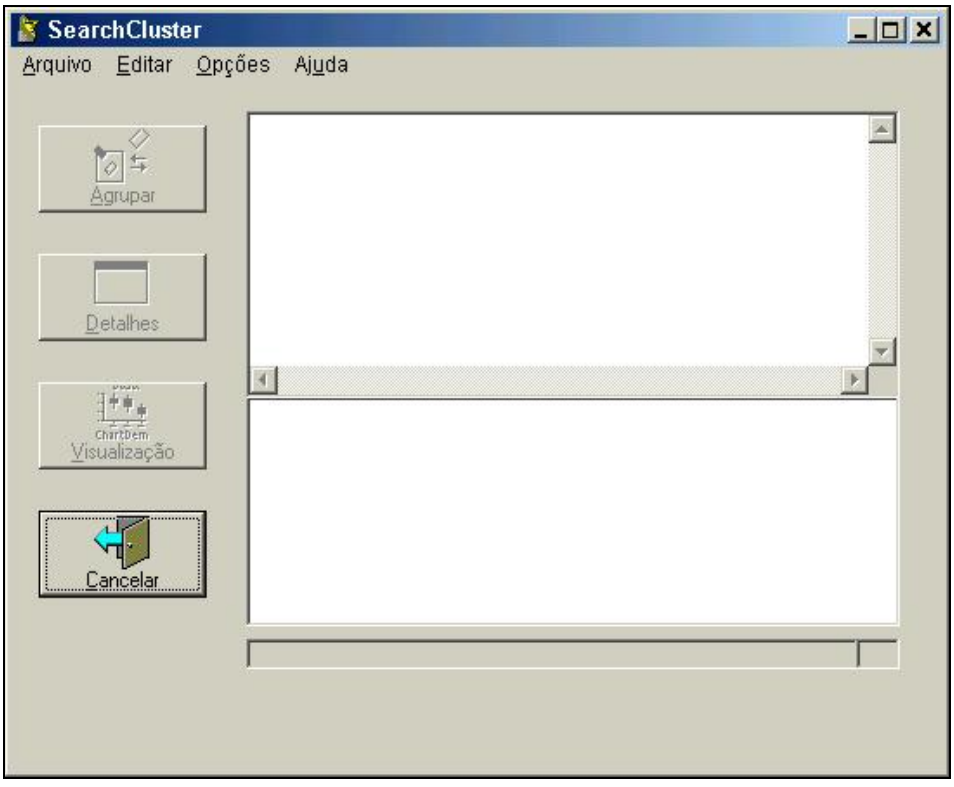


Figura 4.1: a tela inicial do *SearchCluster*.

Na barra de menus, o primeiro item encontrado é “Arquivo”, que ao ser acionado apresenta um menu para seleção dos arquivos de leitura e gravação, conforme a Figura 4.2. O sub-item “Gravação” permite ativar a geração de arquivos de saída de objetos, com os atributos originais da base mais uma coluna com um identificador do grupo ao qual aquele objeto foi alocado na finalização do processo e

de saída de agrupamentos com os detalhes de cada grupo formado na segmentação ótima: medidas do centro, quantidade de objetos e erro médio de cada grupo.

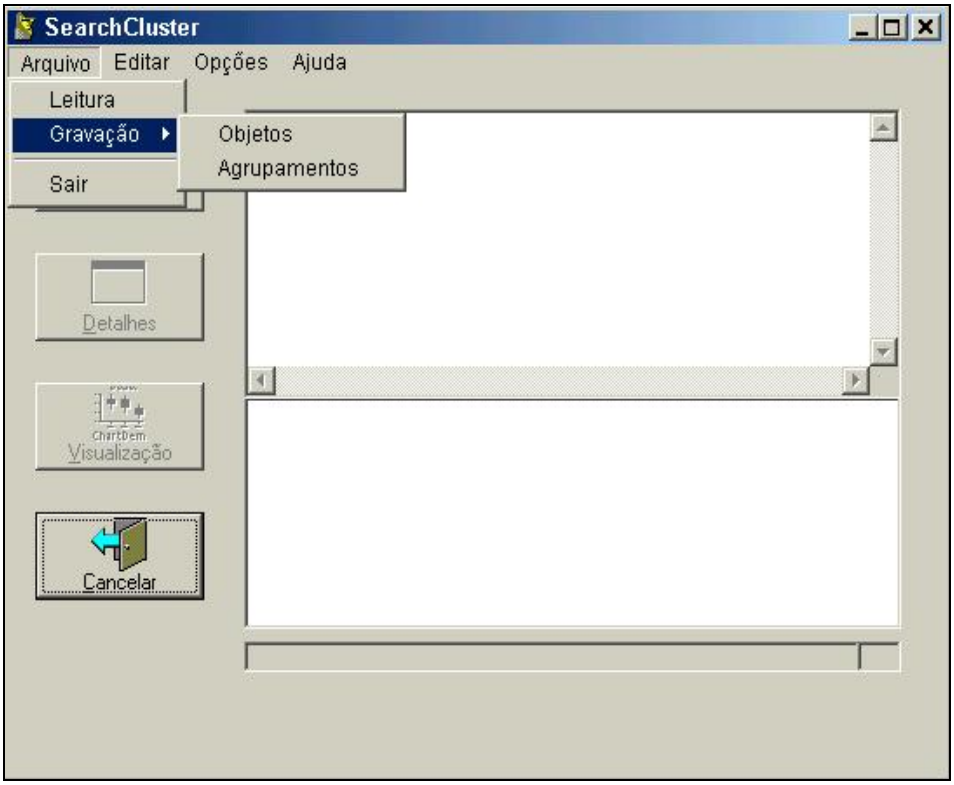


Figura 4.2: o menu “Arquivo” expandido.

O acionamento do subitem “Leitura” dá acesso a um recurso do sistema, que pode ser visto na Figura 4.3, onde o usuário pode navegar e selecionar o arquivo de entrada, ou seja, a base de dados que será processada pelo aplicativo.

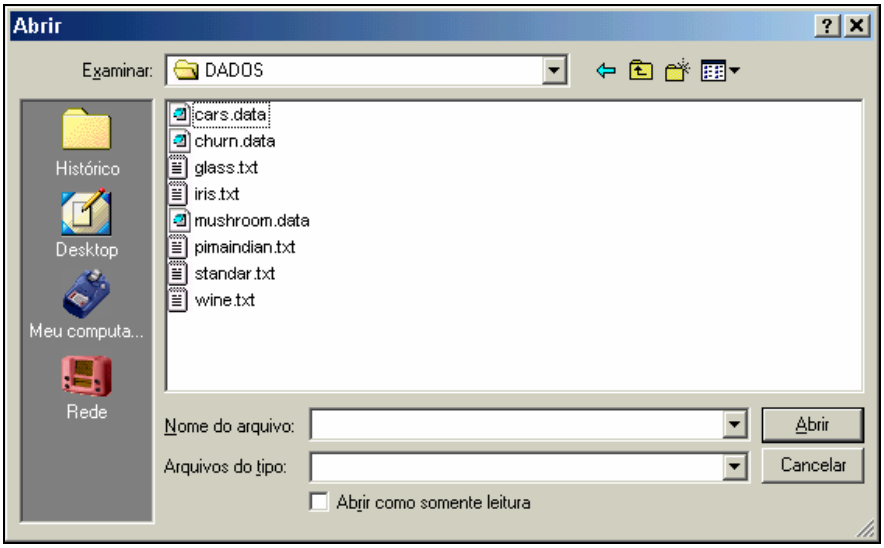


Figura 4.3: o recurso do sistema para selecionar a origem dos dados.

O acionamento do menu “Opções” exibe uma tela que possui três guias para interação do usuário.

Na primeira guia, “Atributos e Grupos”, vista na Figura 4.4, o usuário deverá indicar o “Número de atributos”, ou seja o número de colunas, de cada objeto componente da base de dados; o “Número de grupos” que deverão ser pesquisados. Este bloco de opções fornece os parâmetros necessários para o *SearchCluster* dividir e redistribuir os objetos de acordo com as técnicas já comentadas, seqüencialmente do número “Mínimo” até o número “Máximo” de grupos, testar e adotar a melhor formação de grupos encontrada dentro de tal faixa. No bloco de opções “Desprezar identificadores”, o usuário poderá assinalar quaisquer parâmetros, fazendo com que o software ignore elementos da base que sejam apenas identificadores e não atributos ou objetos, tais como linhas de cabeçalho com o título dos atributos e colunas que indiquem um mero número seqüencial do objeto dentro da base.

Figura 4.4: a guia “Atributos e Grupos”.

A guia “Exibir e Gravar” oferece duas opções apenas, conforme pode ser observado na Figura 4.5. A opção “Exibir erro e função objetivo de todos os agrupamentos testados”, se marcada, indica ao aplicativo que deverão ser exibidos, na caixa de texto superior da tela inicial, o resultado do cálculo do erro e da função objetivo para cada um dos agrupamentos testados dentro da faixa assinalada no bloco “Número de atributos”. A opção “Gravar arquivo de objetos para cada agrupamento testado”, quando marcada pelo usuário, faz com que seja gravado um arquivo com a configuração de cada objeto incluindo o identificador do grupo ao qual o objeto terminou por ser alocado, de forma que, se n agrupamentos forem testados, n arquivos de objetos serão gerados.

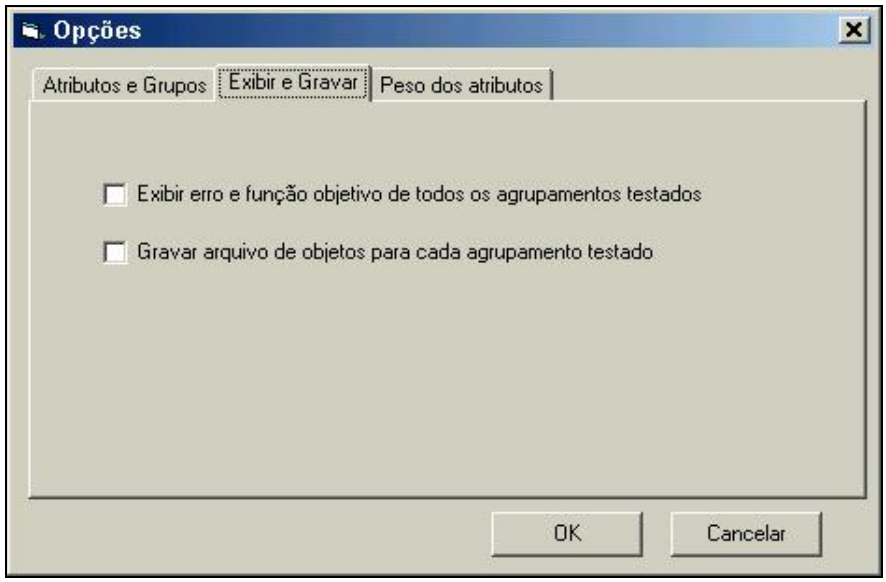


Figura 4.5: a guia “Exibir e Gravar”.

Na Figura 4.6 é apresentada a guia “Peso dos atributos”. Este é um recurso bastante interessante do software *SearchCluster*, pois possibilita que o usuário forneça um peso diferenciado para um ou mais atributos da base. Inicialmente são apresentados os atributos, em número igual àquele informado na guia “Atributos e Grupos”, opção “Número de atributos” (Figura 4.4), todos com peso 1. O usuário pode aumentar, diminuir ou até anular a participação de quaisquer dos atributos, bastando fornecer um número de zero (para anular) até 99999 (para maximizar), que será o novo peso relativo daquele atributo. Tal alteração reflete de imediato na visualização dos objetos e de seus eixos, com a aproximação proporcional ou distanciamento do centro do sistema. Também influencia na segmentação da base, podendo variar o número de grupos formados e a alocação dos objetos.

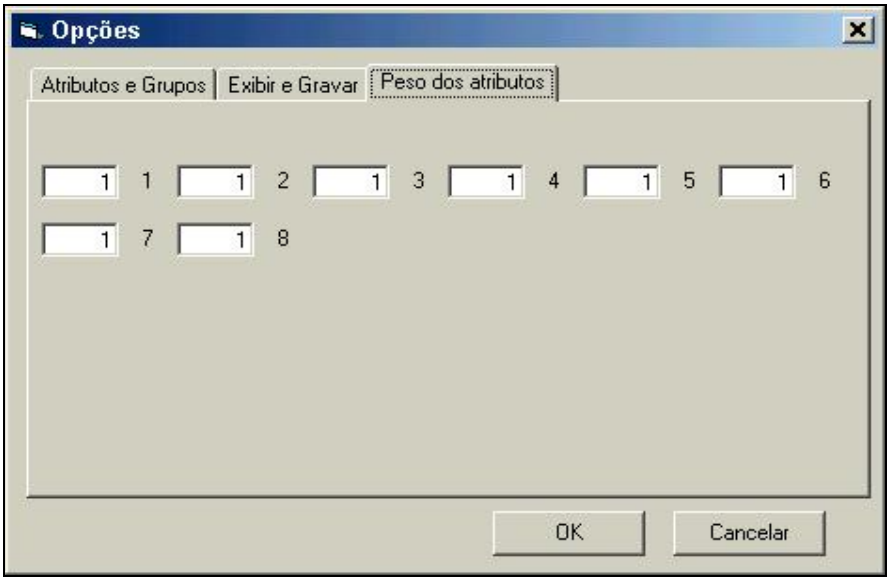


Figura 4.6: a guia “Peso dos atributos”.

Após a escolha do arquivo de entrada e das configurações das opções, os botões “Agrupar” e “Visualização” também tornam-se habilitados, como mostra a Figura 4.7. O usuário poderá escolher livremente uma das ações disponíveis. Na caixa de texto superior, pode ser vista uma relação com as configurações adotadas para os próximos exemplos.

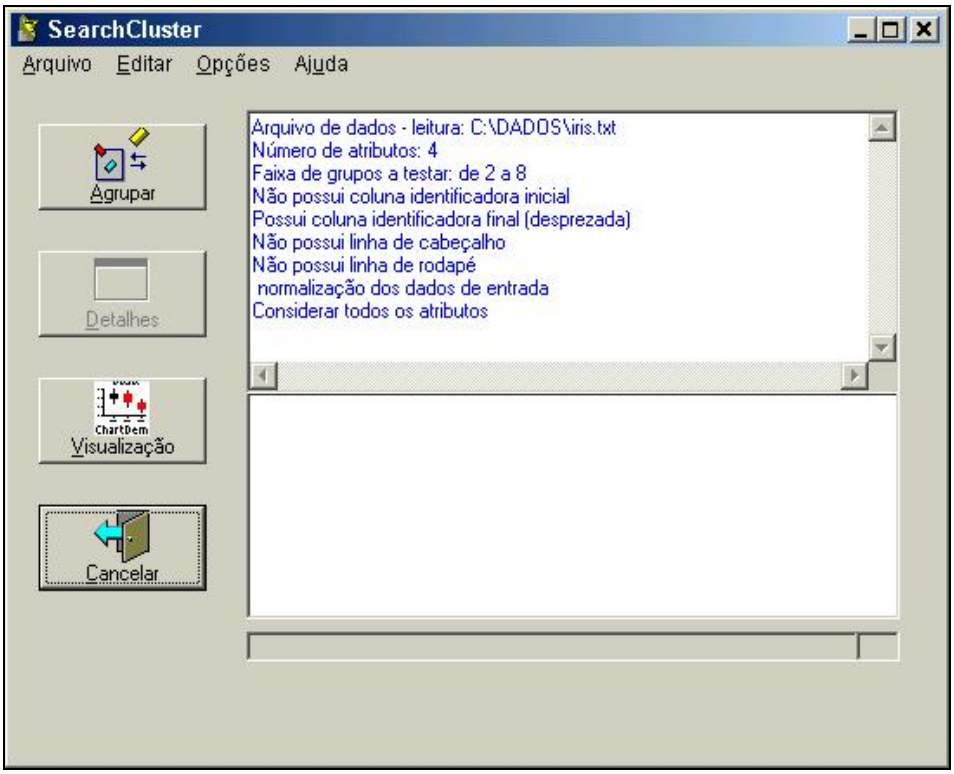


Figura 4.7: tela principal, após escolha das opções.

Os botões de ação têm seu funcionamento independente, pois o usuário pode desejar ver a disposição inicial dos objetos sem preocupação com a divisão destes em grupos. Ao ser pressionado, o botão “Agrupar” dá inicio ao processo iterativo de divisão, alocações sucessivas dos objetos e cálculo da função objetivo, tantas vezes quantas forem indicadas pela faixa de números de grupos a serem testados.

Ao término das iterações, a tela principal é atualizada com a habilitação do botão “Detalhes” e o preenchimento da caixa de texto inferior, onde é mostrado o resultado obtido pela melhor segmentação, como é mostrado na Figura 4.8. Neste exemplo, ainda pode-se ver que, na janela superior, o erro e a função objetivo de cada agrupamento testado no intervalo também são exibidos, conforme a configuração das opções realizada na tela de opções.

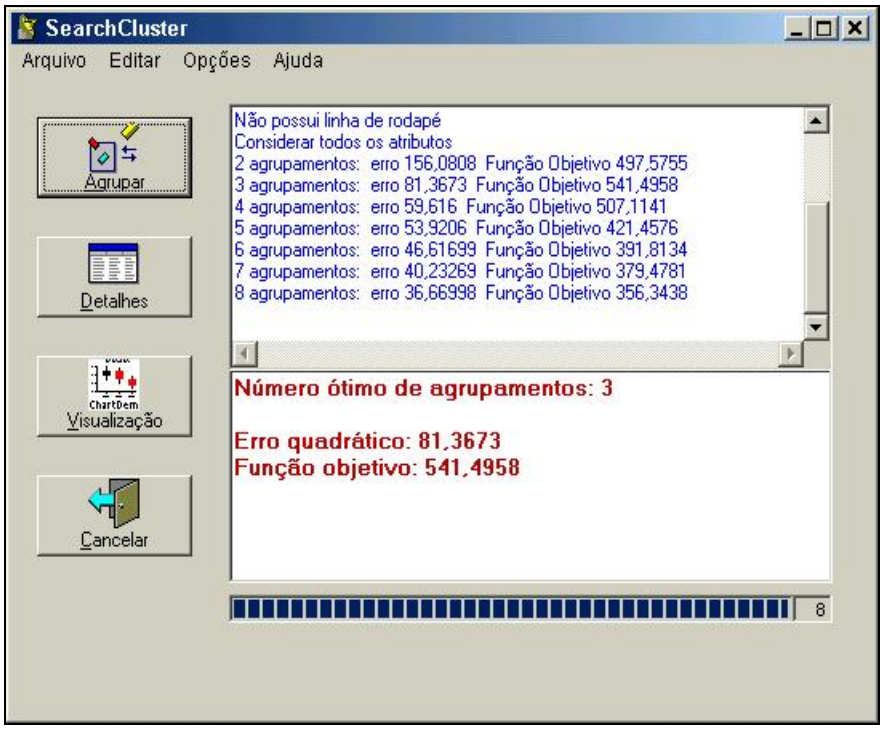


Figura 4.8: tela principal, com os resultados do processamento.

O pressionamento do botão “Detalhes” dá origem a uma tela simplesmente informativa, onde são apresentados detalhes do agrupamento em sua formação ótima: o número de objetos, o erro total e o erro médio em cada grupo, conforme pode ser visto na Figura 4.9.

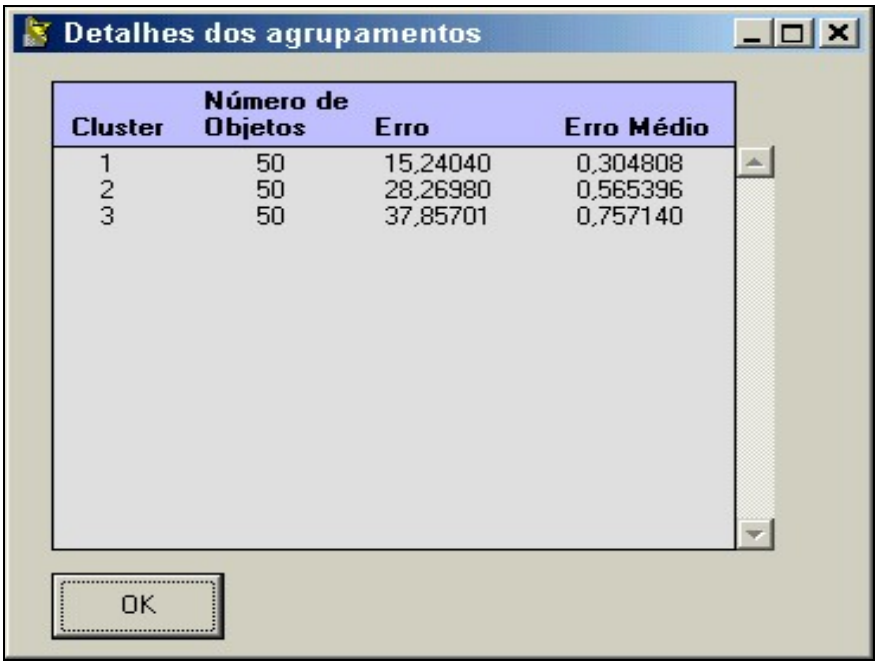


Figura 4.9: detalhes do agrupamento em sua formação ótima.

O botão “Visualização”, ao ser pressionado, dispara o processo que transforma os objetos n-dimensionais em pontos bi-dimensionais, com o emprego da

técnica de “Coordenadas Estrela”, de forma que possam ser exibidos na tela, guardando um posicionamento virtual, porém lógico e inteligível de sua distribuição em um espaço R^2. Após a efetuação da transformação de todos os objetos, inclusive o centro dos grupos, se aqueles já estiverem agrupados, é exibida uma tela, conforme visto na Figura 4.10, que apresenta os eixos formados pelos atributos, neste exemplo, quatro atributos que dão origem a quatro eixos, um identificador seqüencial dos atributos próximo ao seu respectivo eixo e os objetos, distribuídos no espaço. Quando é realizada a visualização sem que haja o agrupamento prévio, os objetos são representados com apenas uma cor. Neste exemplo, podem ser vistos objetos em três cores distintas. Com efeito, foram três os grupos detectados durante o processo de agrupamento.

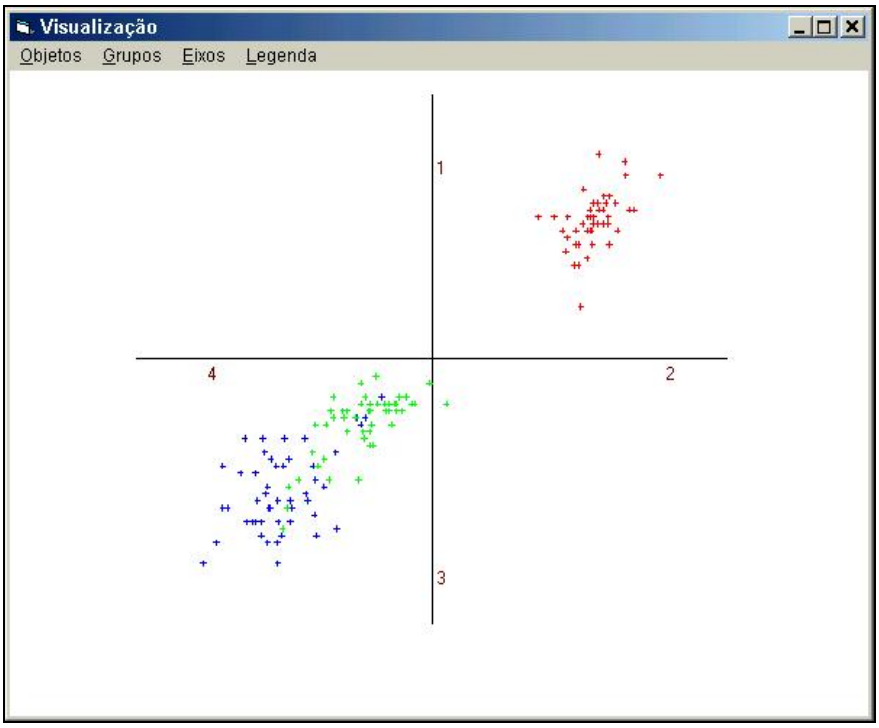


Figura 4.10: a tela gráfica: visualização dos objetos com “Coordenadas Estrela”.

Com a barra de menus da tela gráfica pode-se utilizar o item “Gráfico Estrela” do menu “Objetos” para desativar a visualização destes e o equivalente no menu “Grupos” para ativar a visualização dos pontos determinantes do centro dos grupos. O resultado pode ser visto na figura 4.9. Também é possível a visualização dos objetos juntamente com o centro dos grupos.

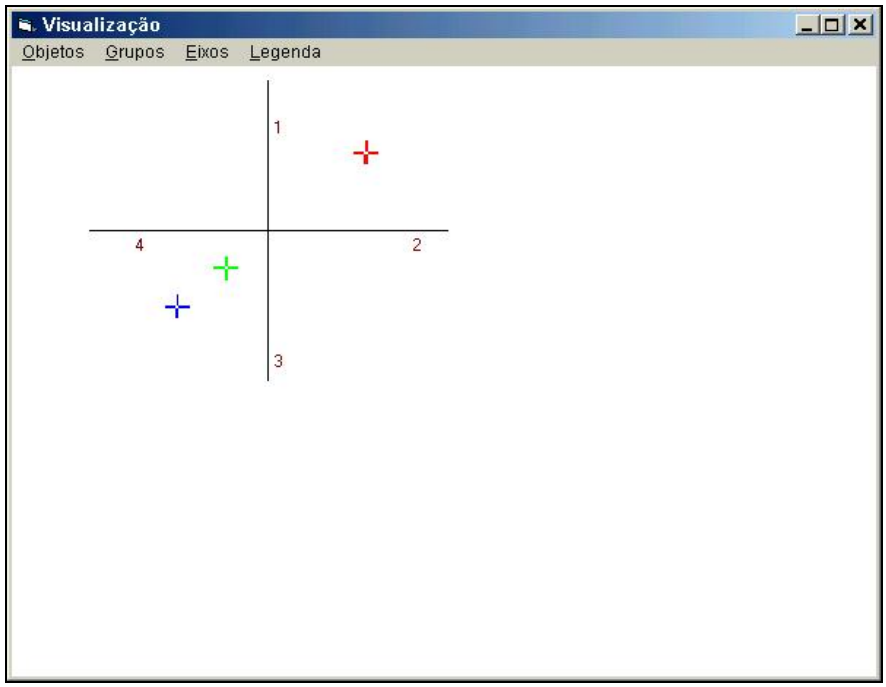


Figura 4.11: visualização dos centros dos grupos.

Na tela gráfica, podem ser acionados o botão esquerdo do *mouse* ou a tecla "-" e o botão direito do *mouse* ou a tecla "+" que, respectivamente, irão diminuir ou aumentar a escala da visualização. Utilizando-se o menu "Legenda", é possível habilitar uma legenda que relaciona o número do grupo com a cor da visualização, subitem "Grupo x Cor", e um identificador junto à representação gráfica do centro do grupo, através do subitem "Id Grupo". Estas facilidades podem ser notadas na Figura 4.12.

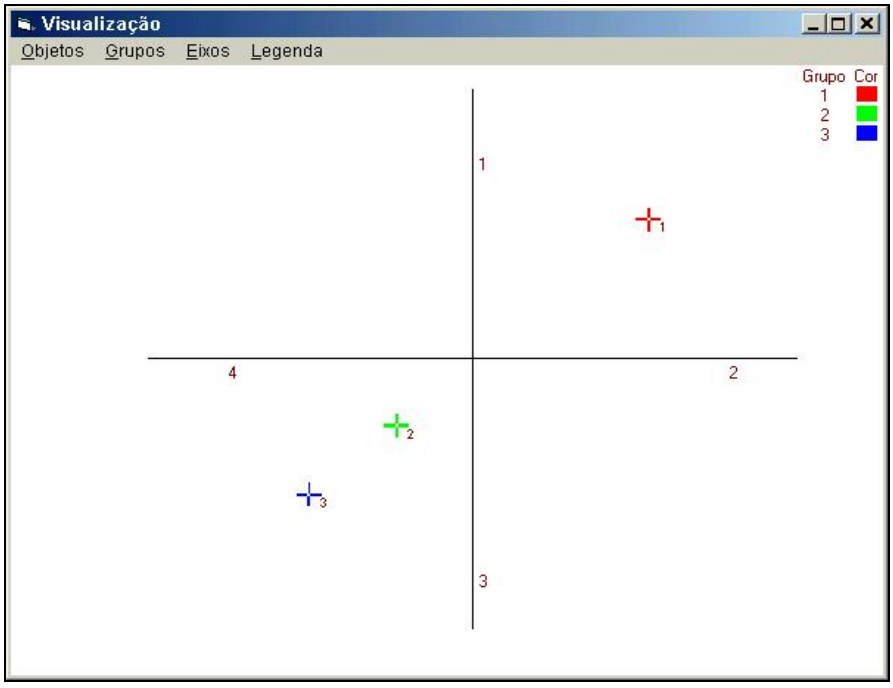


Figura 4.12: visualização com escala ampliada, sem objetos e com legenda.

Durante a visualização dos gráficos estrela, o menu “Eixos” encontra-se habilitado. Através dele pode-se ocultar ou exibir os eixos que, inicialmente, guardam a mesma proporção e ângulos entre si. Se o peso de algum atributo for alterado, a proporção do eixo e a posição dos objetos e, conseqüentemente, a posição do centro do grupo também se apresentarão deslocados. Na Figura 4.13 é representada uma visualização onde o peso do atributo 3 foi alterado do valor padrão 1 para 0,3.

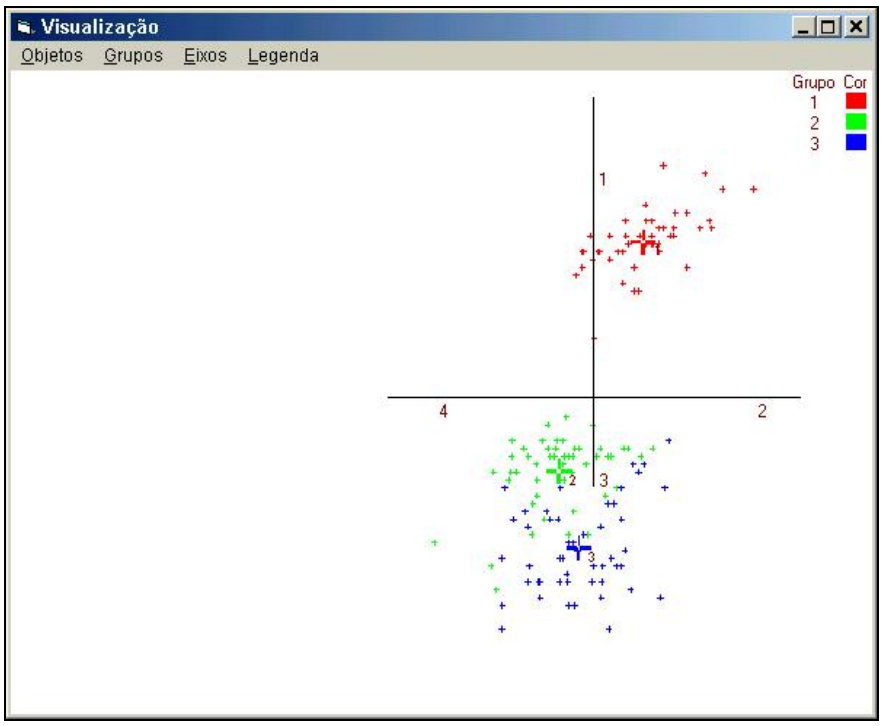


Figura 4.13: visualização com alteração do peso do atributo 3 e legenda de cores.

Outro item disponível no menu “Objetos”, que é o “Dendrograma”, permite escolher a visualização do resultado do emprego do método aglomerativo Vizinho Mais Próximo, permitindo que sejam analisadas visualmente as ligações hierárquicas dos objetos componentes da base, conforme pode ser visto na Figura 4.14. Se os objetos já estão alocados em seus grupos, são representados nas cores respectivas aos mesmos. Para ressaltar a diferenciação, as linhas verticais que não representam objetos são traçadas na cor preta.

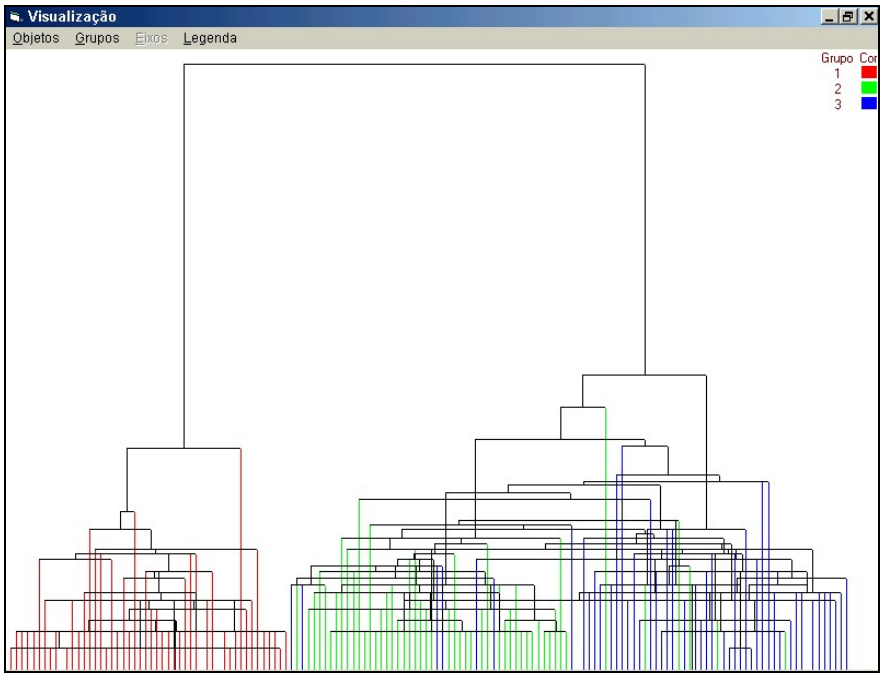


Figura 4.14: visualização do dendrograma da base de dados.

Aliado ao dendrograma, é possível ter mais informações acerca dos resultados da técnica hierárquica, exibindo uma tela com o histograma de nível, acessado através do item “Histograma”, do menu “Objetos”, conforme pode ser visto na Figura 4.15. Com o objetivo de servir a uma confrontação da análise dos objetos com o resultado do processo de agrupamento posterior, o dendrograma e o seu relativo histograma para os grupos também estão disponíveis, através dos itens respectivos no menu “Grupos”. Em ambos os casos de utilização, os histogramas somente tornam-se disponíveis após a realização do dendrograma e, naturalmente, a exibição do dendrograma de um desfaz a exibição do outro

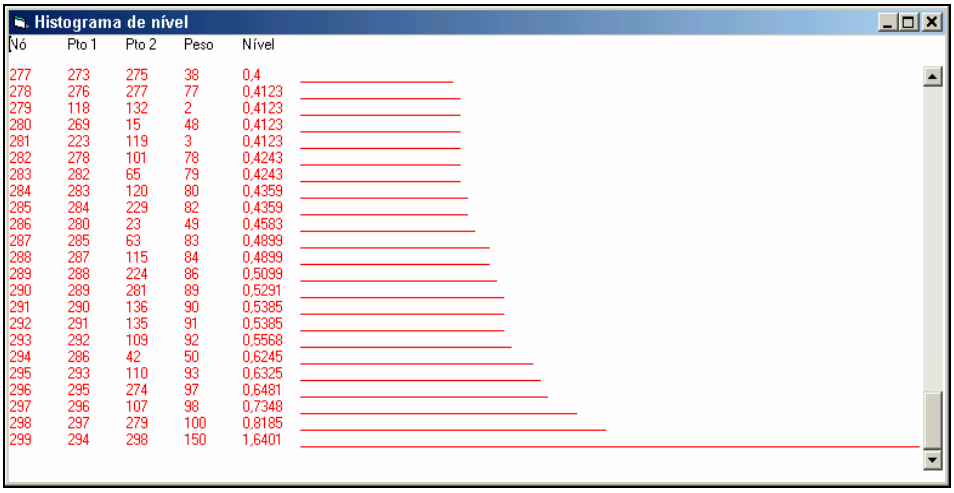


Figura 4.15: histograma de nível.

O histograma apresenta todos os objetos e suas ligações, chamadas genericamente de nós, e os representa por um número seqüencial, de maneira que ao primeiro objeto da base é associado o número 1; ao segundo, o número 2 e assim sucessivamente até o último objeto. A partir daí, as ligações entre os objetos e outras ligações também vão recebendo, pela ordem, um número seqüencial. Para que seja possível distinguir os elementos, objetos ou outras ligações, que formam ligações hierárquicas as colunas “Pto 1” e “Pto 2” apresentam os dois elementos que foram ligados. A coluna “Peso” significa a quantidade de objetos ligados naquele nível. A coluna “Nível” apresenta o índice de nível obtido pelo cálculo da distância entre os objetos mais próximos, ligados em tal nível.

4.4 Código

O código fonte, escrito na linguagem de programação Visual Basic 6.0, de todos os formulários e módulos do aplicativo *SearchCluster*, encontra-se listado integralmente no apêndice 1.

CAPÍTULO 5
ESTUDOS DE CASOS

Com o intuito de, uma vez mais, validar e comprovar a viabilidade computacional de algumas das técnicas abordadas neste trabalho, bem como demonstrar o funcionamento confiável da ferramenta desenvolvida, foram realizados três estudos de caso empregando-se quatro softwares especializados em mineração de dados existentes no mercado: o *gCLUTO*, o SAS *Enterprise Miner*, o SGI *Mineset* e o IBM *Intelligent Miner for Data*, cujas características básicas e formas de visualização principais foram abordadas no capítulo 4.

Além disso, buscando verificar a eficácia do critério de avaliação de qualidade da segmentação utilizado pelo *SearchCluster*, Critério de *Calinski* e *Harabasz*, foi efetuada uma comparação com uma implementação do método clássico do Índice PBM, descrito no capítulo 2, para estimar a melhor formação de grupos.

O primeiro estudo foi realizado sobre uma base de dados do DNA de cana de açúcar, o segundo realizou uma análise sobre automóveis de diferentes origens e o terceiro sobre dados de clientes "*churn"* de uma empresa de telecomunicações. Os três estudos são descritos a seguir.

5.1 Primeiro caso – base de dados do DNA de cana de açúcar

Os dados empregados neste estudo são oriundos de um projeto de genética da cana de açúcar, conhecido como SUCEST, que envolveu diversos laboratórios e grupos de *datamining* na produção de bibliotecas de seqüências de DNA. A aplicação de técnicas de segmentação tornam-se necessárias na formação de grupos de bibliotecas semelhantes e eliminar as redundâncias que possam vir a ser geradas no processo de sequenciamento.

Neste estudo foram utilizados 193 conjuntos de *reads* - objetos. Cada objeto, que em sua formulação original continha informações de 37 bibliotecas, teve a sua composição reduzida, pois 18 bibliotecas foram eliminadas da base por ser

considerado que suas contribuições teriam pouca ou nenhuma relevância para esta análise, reduzindo, assim, para 19 os atributos a serem processados. O que colabora, ainda, para diminuir a incidência de possíveis ruídos.

5.1.1 Resultados do *SearchCluster* sobre a base de cana de açúcar

Ao realizar a análise desta base utilizando-se o *SearchCluster* processando o algoritmo de *Calinski* e *Harabasz*, foram encontrados 18 grupos, conforme demonstrado na Figura 5.1.

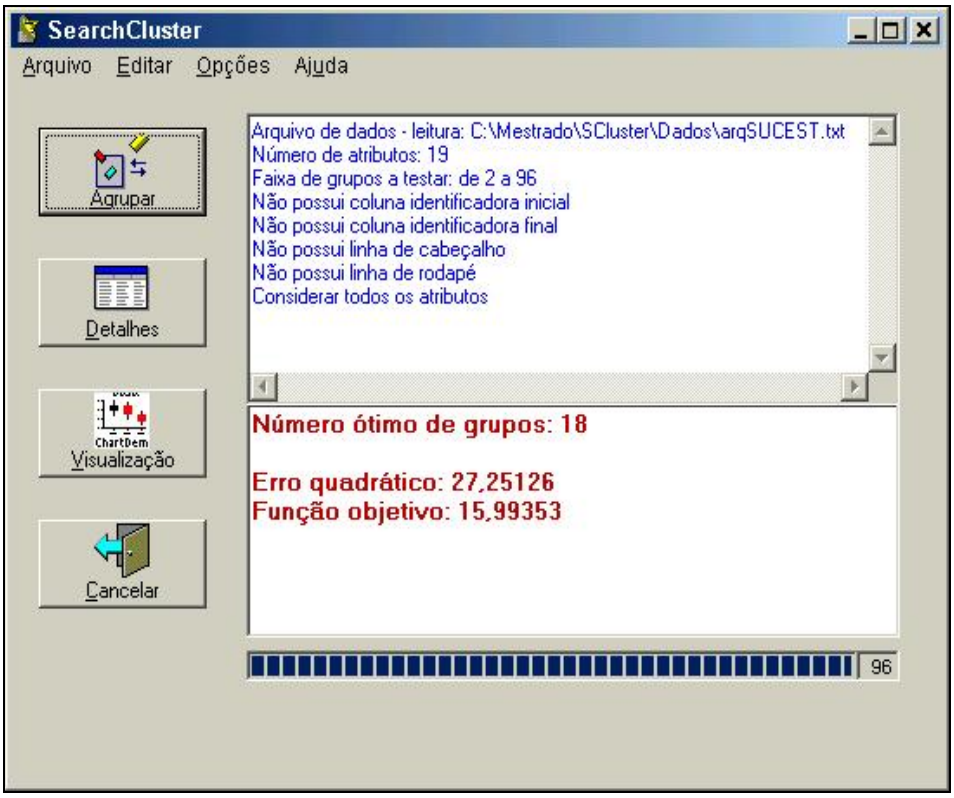


Figura 5.1: tela de resultados do processamento da base de DNA de cana de açúcar.

Detalhes dos grupos, úteis para uma melhor interpretação, como número de objetos e valor do erro podem ser vistos na Figura 5.2.

O resultado obtido é bastante razoável, tendo em vista que o número de grupos esperado por especialistas da área seria de 20.

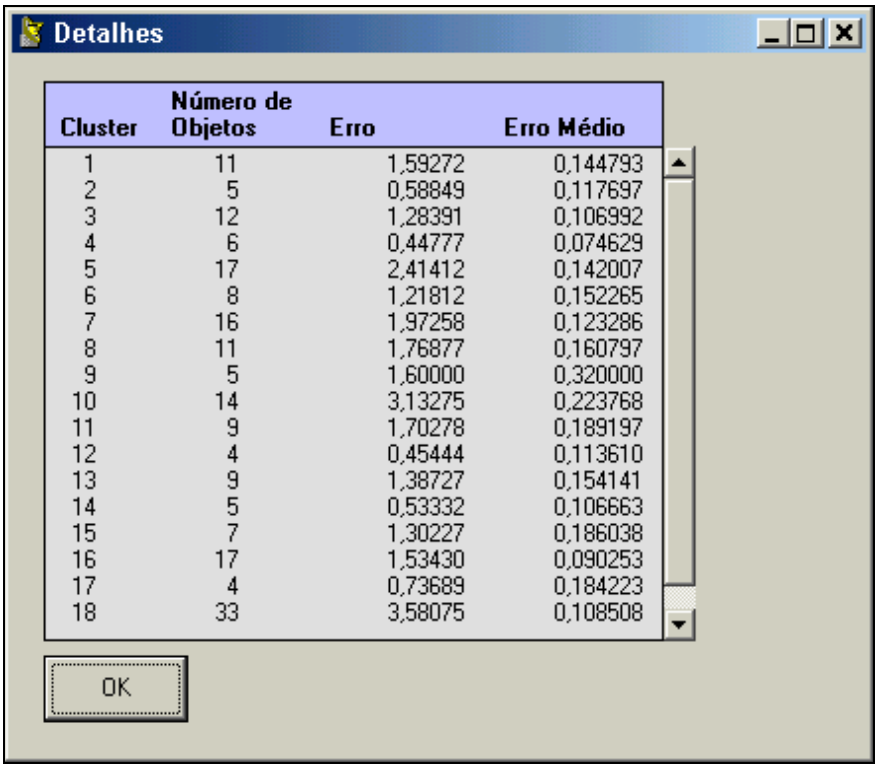


Figura 5.2: detalhes dos grupos formados pelo *SearchCluster*.

A Figura 5.3 mostra a representação dos objetos e o centro dos grupos em que eles se dividem, num plano visual de duas dimensões.

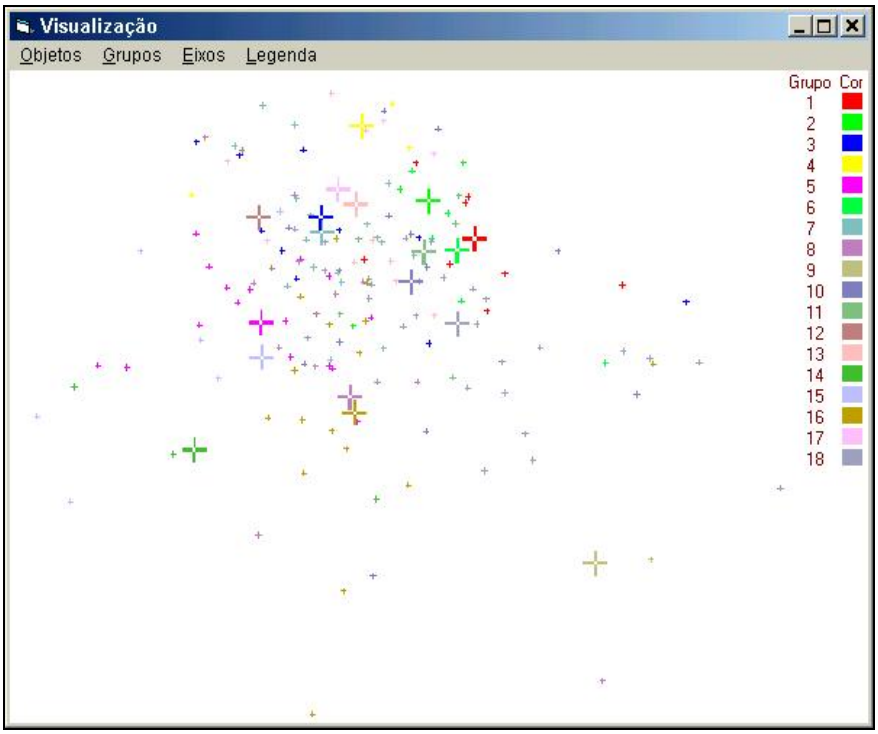


Figura 5.3: visualização dos objetos e centro dos grupos.

A segmentação da base de cana de açúcar empregando o método clássico do Índice PBM apontou, como melhor resultado, a divisão em dois grandes

agrupamentos, bastante diverso dos resultados atingidos pelo *SearchCluster* e pelos testes com outros softwares, mostrados a seguir, o que pode ser creditado à tendência que o Índice PBM possui de apontar como ótimas as segmentações formadas pelos menores números de grupos. Todavia, os gráficos exibidos na Figura 5.4 e na Figura 5.5 apresentam coincidências interessantes, apesar de inicialmente exibirem resultados completamente díspares: um máximo local pode ser observado na formação de 18 grupos, a partir de 20 grupos as curvas começam a coincidir, após o 24º grupo há uma brusca ruptura no padrão e uma estabilização nos patamares mais inferiores dos resultados.

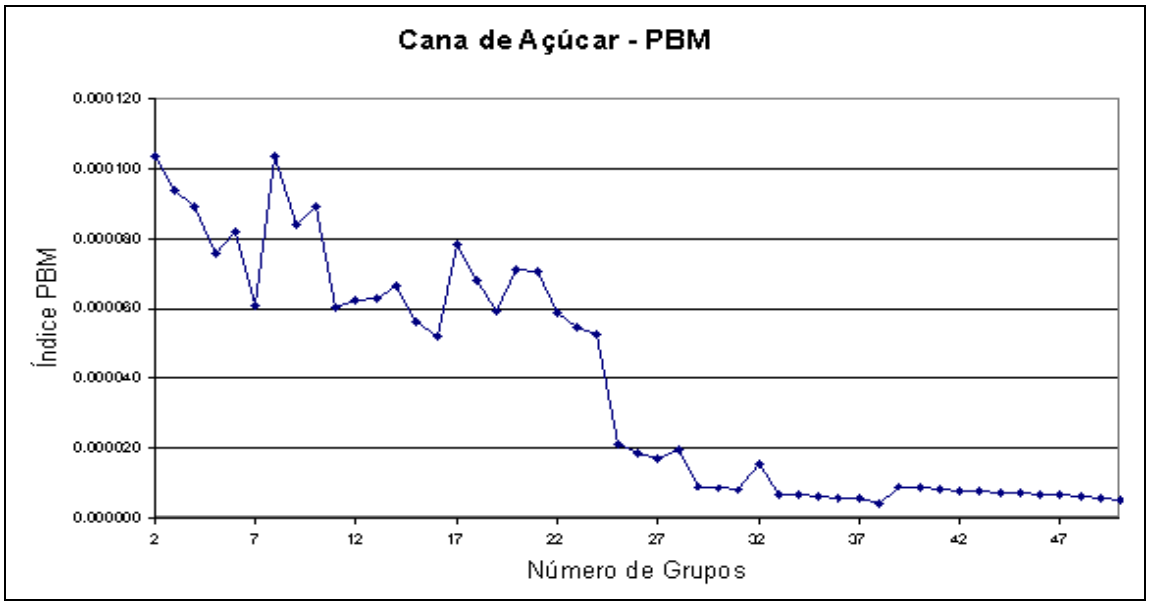


Figura 5.4: evolução do Índice PBM em relação à quantidade de grupos formados.

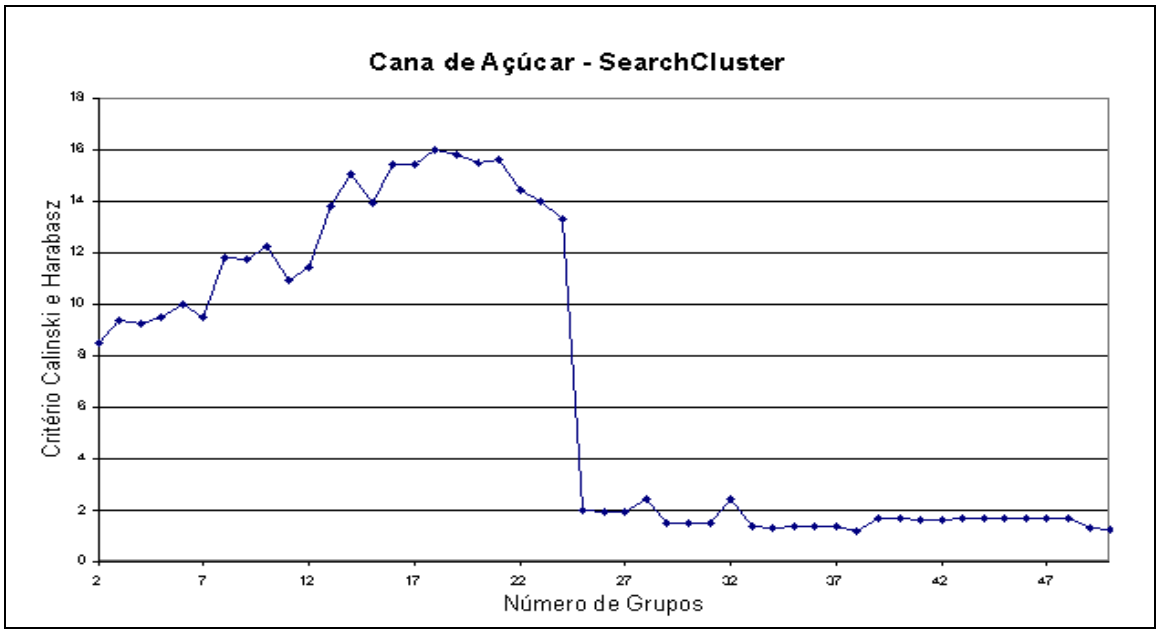


Figura 5.5: evolução do Critério *Calinski* e *Harabasz* em relação à quantidade de grupos formados.

5.1.2 Resultados do *gCLUTO* sobre a base de cana de açúcar

Para empregar o *gCLUTO* sobre qualquer base é necessário informar em quantos grupos deseja-se dividir o conjunto de dados. Analisando os resultados obtidos pelos quatro métodos de segmentação disponíveis no software, o que aparentemente apresentou a melhor segmentação da base de cana de açúcar foi o método direto. Neste caso, a visualização de montanha apresentou a maior distância entre os pico, picos com boa altura, inclusive atingindo tons de vermelho, o que indica baixa divergência interna. As telas com os resultados em modo texto e visualização de montanha podem ser observadas na Figura 5.6 e na Figura 5.7, respectivamente.

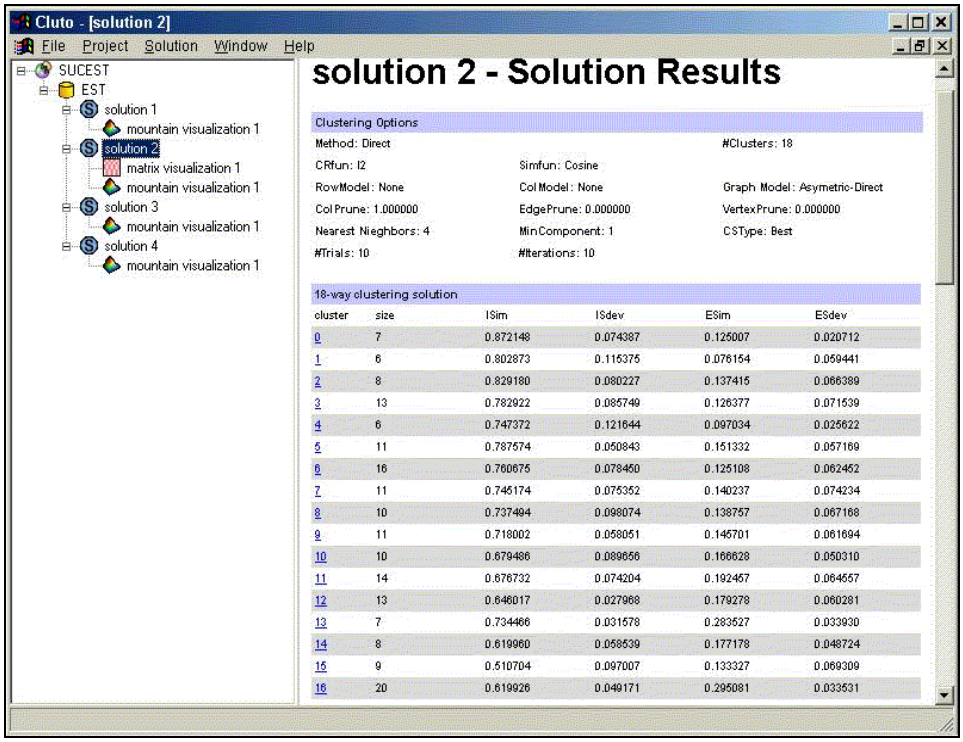


Figura 5.6: resultados em modo texto da segmentação pelo *gCLUTO*.

Comparando-se a Figura 5.3 e a Figura 5.7 é possível notar que a segmentação obtida em ambos os softwares apresenta a formação de um elevado número de grupos bastante próximos entre si e dois ou três bastante destacados.

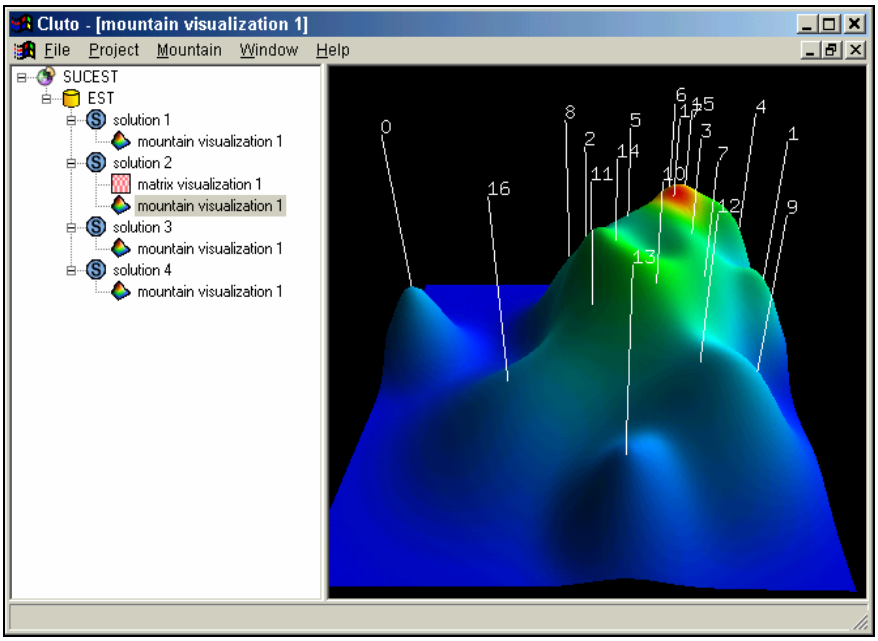


Figura 5.7: visualização de montanha.

5.1.3 Resultados do *Enterprise Miner* sobre a base de cana de açúcar

A segmentação realizada pelo *Enterprise Miner* sobre a base de cana de açúcar não apresentou um resultado que possa ser considerado bom, pois a descoberta de 5 grupos, conforme pode ser visto na Figura 5.8, fica muito aquém do número de grupos preconizado para esta base, que é de 20.

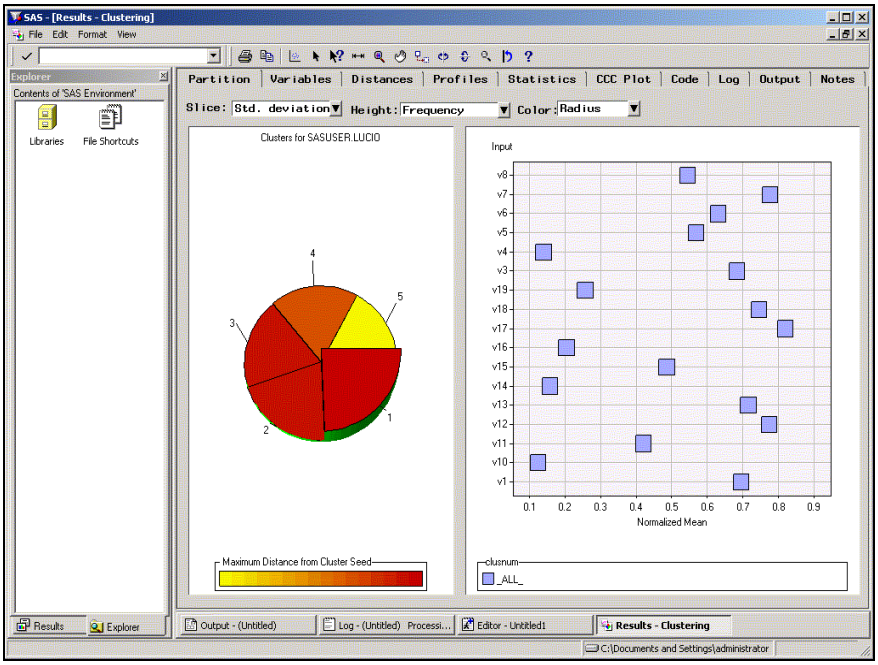


Figura 5.8: segmentação da base de cana de açúcar obtida pelo *Enterprise Miner*.

O gráfico que apresenta a proximidade dos grupos, visto na Figura 5.9, deixa perceber que dois dos grupos formados divergem bastante dos outros três que são muito próximos entre si. Este fato já havia sido notado na visualização de objetos e centro dos grupos do *SearchCluster*. Provavelmente, os três grupos que apresentam-se mais próximos sejam, de fato, diversos grupos menores considerados como únicos pelo *Enterprise Miner*.

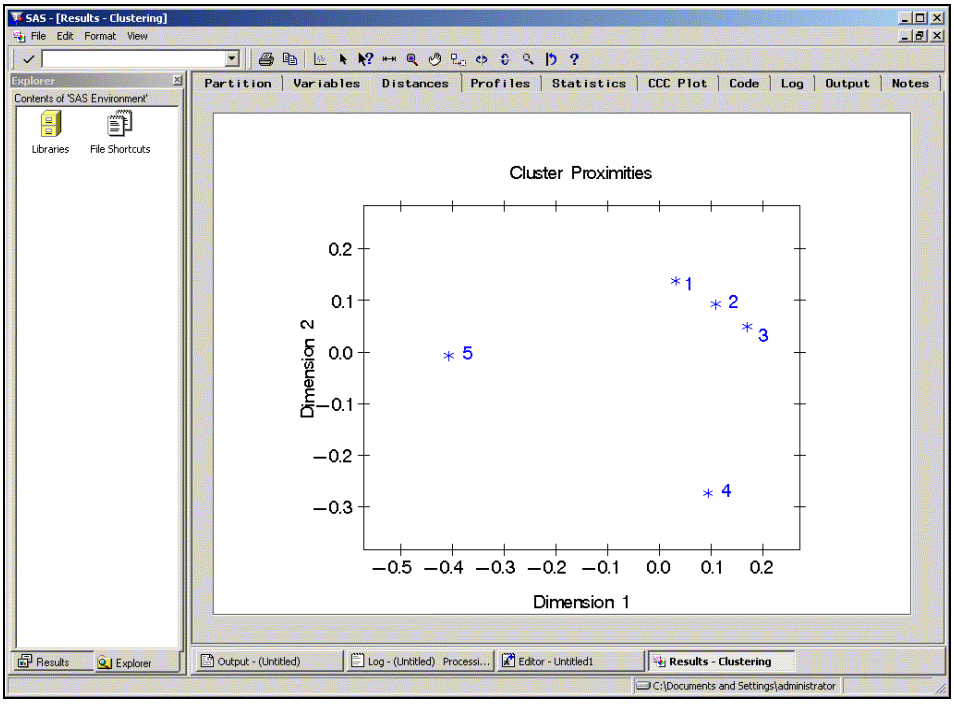


Figura 5.9: gráfico de proximidade dos grupos da base de cana de açúcar.

O gráfico que apresenta os valores atingidos pelo critério para determinação da melhor segmentação, o CCC – *cubic clustering criterion*, empregado pelo *Enterprise Miner*, apresenta diversas mudanças no padrão de comportamento do critério em determinadas faixas de grupos testados, iniciando com um declínio do valor obtido, seguido por um aumento gradativo, uma relativa estabilização e, posteriormente, uma nova queda nos valores, conforme pode ser observado na Figura 5.10, mas não apresenta qualquer indicação de que a segmentação ideal deveria estar em torno de 20 grupos.

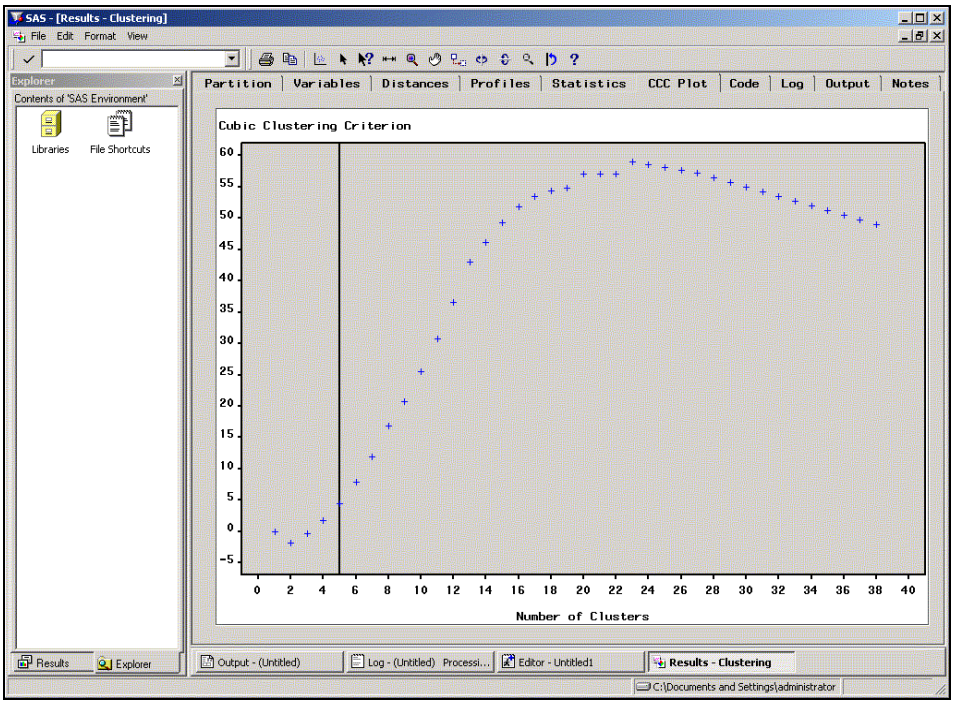


Figura 5.10: gráfico do CCC para as segmentações da base de cana de açúcar.

5.1.4 Resultados do *Mineset* sobre a base de cana de açúcar

O *Mineset* não faz, verdadeiramente, uma livre descoberta do número ideal de grupos em que a base pode ser dividida. A segmentação é influenciada pelo valor dos parâmetros inicialmente passados pelo utilizador do software. Assim, em um dos vários testes realizados com o *Mineset*, foi realizada uma segmentação em 16 grupos, que pode ser considerado razoável para fins de análise do comportamento da base de cana de açúcar.

Dentre os grupos formados, três deles podem ser vistos parcialmente na Figura 5.11, que contempla a 14ª e a 9ª coluna da base. Podem ser facilmente observados os pontos máximos e mínimos e que os valores dos atributos estão muito próximos do mínimo.

O software da *Silicon Graphics* oferece um relatório texto onde podem ser obtidas outras informações a cerca do processo de segmentação, como o índice de adequação de cada segmentação realizada, o número ótimo de grupos e as distâncias dos grupos formados no melhor agrupamento, como mostra a Figura 5.12.

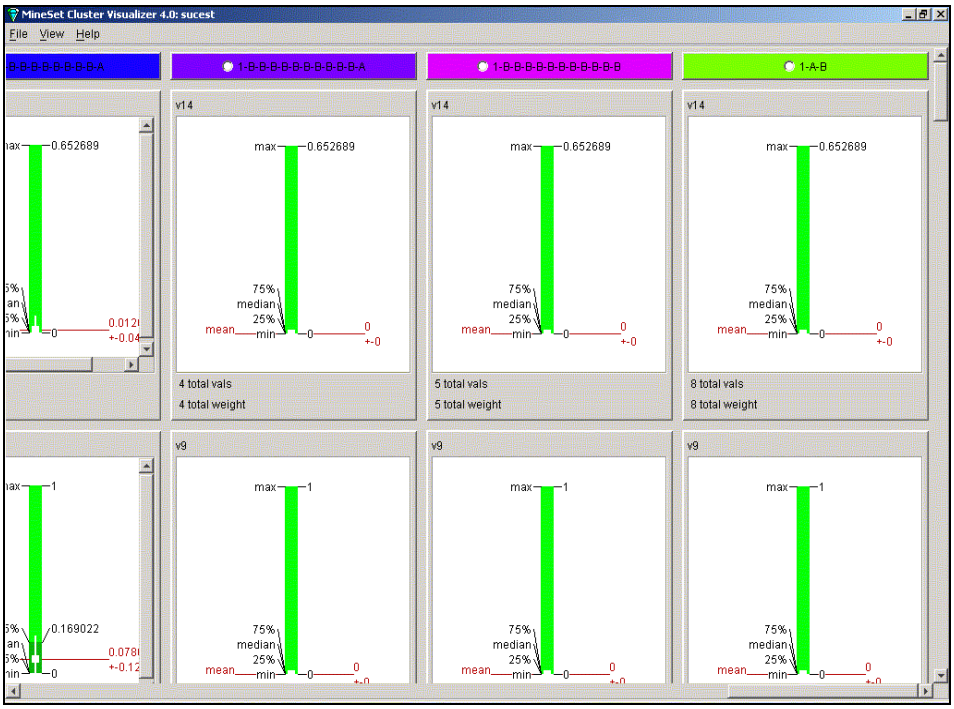


Figura 5.11: visualização estatística da segmentação da base de cana de açúcar.

```
Fitness = 0.6519                    Result of clustering:
1 clusters: fitness = 0.6518735     ------------------------------
2 clusters: fitness = 0.5827084
3 clusters: fitness = 0.5668141     Overall root-mean-squared distance from record
4 clusters: fitness = 0.5491585     to centroid: 0.3948 +- 0.1608
5 clusters: fitness = 0.5287259     RMS distance from records to each centroid:
6 clusters: fitness = 0.5171782       Cluster 1-A-A: 0.1711 +- 0.1147
7 clusters: fitness = 0.5048537       Cluster 1-B-A: 0.3498 +- 0.122
8 clusters: fitness = 0.4889622       Cluster 1-B-B-A: 0.3214 +- 0.07811
9 clusters: fitness = 0.4698183       Cluster 1-B-B-B-A: 0.3353 +- 0.05832
10 clusters: fitness = 0.4594396      Cluster 1-B-B-B-B-A-A: 0.3583 +- 0.06868
11 clusters: fitness = 0.4432538      Cluster 1-B-B-B-B-B-A: 0.3799 +- 0.09091
12 clusters: fitness = 0.4326563      Cluster 1-B-B-B-B-A-B-A-A: 0.4494 +- 0.1979
13 clusters: fitness = 0.4201383      Cluster 1-B-B-B-B-A-B-B: 0.5328 +- 0.07941
14 clusters: fitness = 0.4047414      Cluster 1-B-B-B-B-A-B-A-B: 0.4112 +- 0.2758
15 clusters: fitness = 0.3806461      Cluster 1-B-B-B-B-B-B-A: 0.5797 +- 0.1917
16 clusters: fitness = 0.3607603      Cluster 1-B-B-B-B-B-B-B-A: 0.3762 +- 0.08184
17 clusters: fitness = 0.349436       Cluster 1-B-B-B-B-B-B-B-B-A: 0.5547 +- 0.2374
18 clusters: fitness = 0.3391202      Cluster 1-B-B-B-B-B-B-B-B-B-A: 0.3601 +-
19 clusters: fitness = 0.3357459    0.08791
20 clusters: fitness = 0.3259635      Cluster 1-B-B-B-B-B-B-B-B-B-B-A: 0.2604 +-
21 clusters: fitness = 0.3210674    0.1504
22 clusters: fitness = 0.3082029      Cluster 1-B-B-B-B-B-B-B-B-B-B-B: 0.1441 +-
23 clusters: fitness = 0.3029447    0.09667
24 clusters: fitness = 0.2990027      Cluster 1-A-B: 0.3919 +- 0.09641
25 clusters: fitness = 0.2931139    done.
Chose 16 clusters (choice point =
0.3648658)
```

Figura 5.12: relatório texto do processo de segmentação da base de cana de açúcar.

5.1.5 Resultados do *Intelligent Miner* sobre a base de cana de açúcar

O software *Intelligent Miner* da IBM dispõe de dois métodos que fazem busca automática pela melhor segmentação: o "*Cluster demographic*", que emprega diretamente o Critério de *Condorcet* para avaliação da qualidade dos agrupamentos formados dentro da faixa delimitada expressamente pelo usuário, através do

parâmetro “número máximo de grupos”, e o método “*Cluster neural*”.que determina a melhor segmentação entre todos os agrupamentos formados até um número máximo de grupos estabelecido internamente pela combinação dos parâmetros de entrada.

A base de cana de açúcar foi segmentada primeiramente através do método “*Cluster demographic*”, o qual atingiu 21 como número ótimo de grupos, como pode ser visto na Figura 5.13, que representa parte do relatório gerado pelo método, cuja avaliação de qualidade é feita com o emprego do Critério de *Condorcet*, cujo valor não corresponde necessariamente à media do Critério encontrado para cada grupo, como também pode ser apreendido através da análise do relatório. A quantidade de grupos encontrada é mais próxima do esperado e também assemelha-se aos resultados obtidos nos testes com outras ferramentas, inclusive o *SearchCluster.* Porém, a distribuição dos objetos pelos grupos é muito discrepante, tendo apenas um grupo cerca de 72% do total de objetos, o que não condiz com a realidade.

Cluster_demographic

Result created: 10/04/04 16:07:52

Result File	:	C:\TEMP\WX5A48QZ.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	20
Maximum Number of Clusters	:	23
Improvement Over Last Pass	:	11.00
Similarity Threshold	:	0.68
Mining Run Outputs		
Number of Passes Performed	:	2
Number of Clusters	:	21
Improvement Over Last Pass	:	0.00
Global Condorcet Value	:	0.7954
Parallel Parameters: Applicable For Results Created On Parallel Systems Only		
Records Processed Before Merging	:	10000
NStripe	:	1
NSpan	:	1440

Cluster Characteristics :

Id	Cluster Size		Condorcet	Id	Cluster Size		Condorcet
	Absolute	Relative(%)	Value		Absolute	Relative(%)	Value
0	139	72.02	0.7954	11	2	1.04	0.8389
1	2	1.04	0.8162	12	1	0.52	0.9620
2	6	3.11	0.7631	13	2	1.04	0.8526
3	6	3.11	0.7525	14	1	0.52	0.9590
4	7	3.63	0.7685	15	2	1.04	0.8590
5	3	1.55	0.8423	16	3	1.55	0.7788
6	2	1.04	0.8336	17	3	1.55	0.8513
7	4	2.07	0.8118	18	2	1.04	0.8425
8	2	1.04	0.8903	19	2	1.04	0.8288
9	2	1.04	0.8596	20	1	0.52	0.9653
10	1	0.52	0.9597				

Figura 5.13: resumo do “*Cluster demographic*” para base de cana de açúcar.

A Figura 5.14 apresenta parte do relatório gráfico, onde podem ser vistos detalhes da segmentação pelo método “*Cluster demographic*”, como importância relativa dos grupos e contribuição das variáveis na formação do grupo.

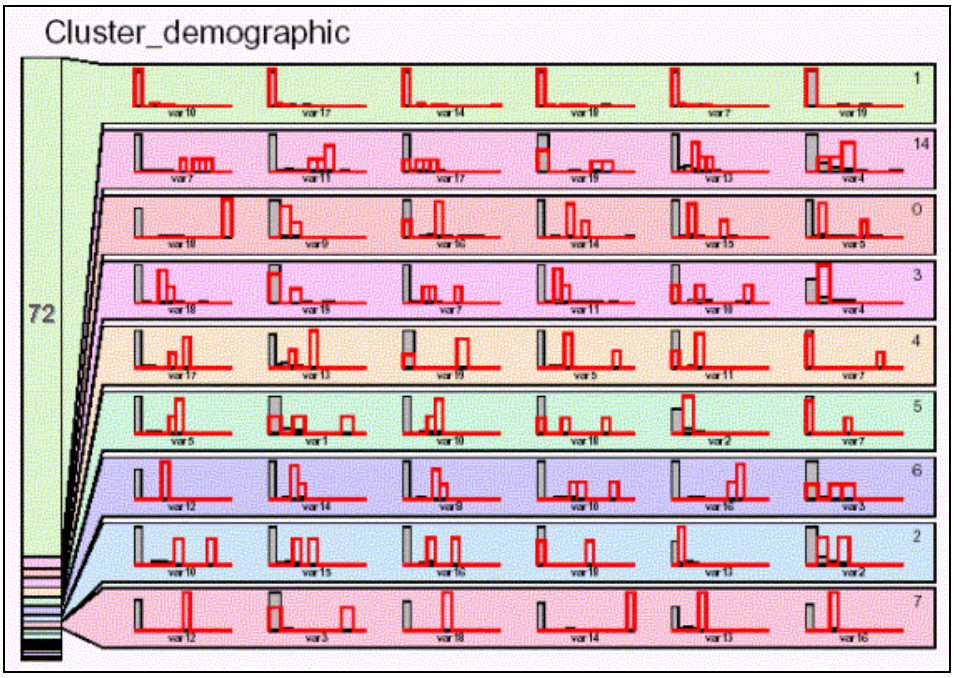


Figura 5.14: gráfico do "*Cluster demographic*" para a base de cana de açúcar.

Em seguida, a base foi segmentada com o emprego do método "*Cluster neural*", cujo melhor resultado, 18 grupos, exatamente o encontrado pelo *SearchCluster*, está representado na Figura 5.15. Nesta segmentação, o número de objetos em cada grupo é bastante satisfatório, sendo, inclusive, bastante coincidente com a distribuição realizada pelo *SearchCluster*.

Cluster_neural

Result created: 10/04/04 15:56:09

Result File	:	C:\TEMP\9ZL6TGEU.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	10
Maximum Number of Clusters	:	9
Mining Run Outputs		
Number of Passes Performed	:	10
Number of Clusters	:	18
Deviation	:	0.17

Cluster Characteristics :

Id	Cluster Size		Id	Cluster Size	
	Absolute	Relative(%)		Absolute	Relative(%)
0	35	18.13	9	6	3.11
1	11	5.70	10	5	2.59
2	9	4.66	11	12	6.22
3	13	6.74	12	11	5.70
4	8	4.15	13	7	3.63
5	9	4.66	14	10	5.18
6	9	4.66	15	14	7.25
7	5	2.59	16	7	3.63
8	7	3.63	17	15	7.77

Figura 5.15: parte do relatório do "*Cluster neural*" para base de cana de açúcar.

Da mesma forma que no método “*Cluster demographic*”, o método “*Cluster neural*” também fornece um gráfico com detalhes da segmentação, como pode ser visto na Figura 5.16.

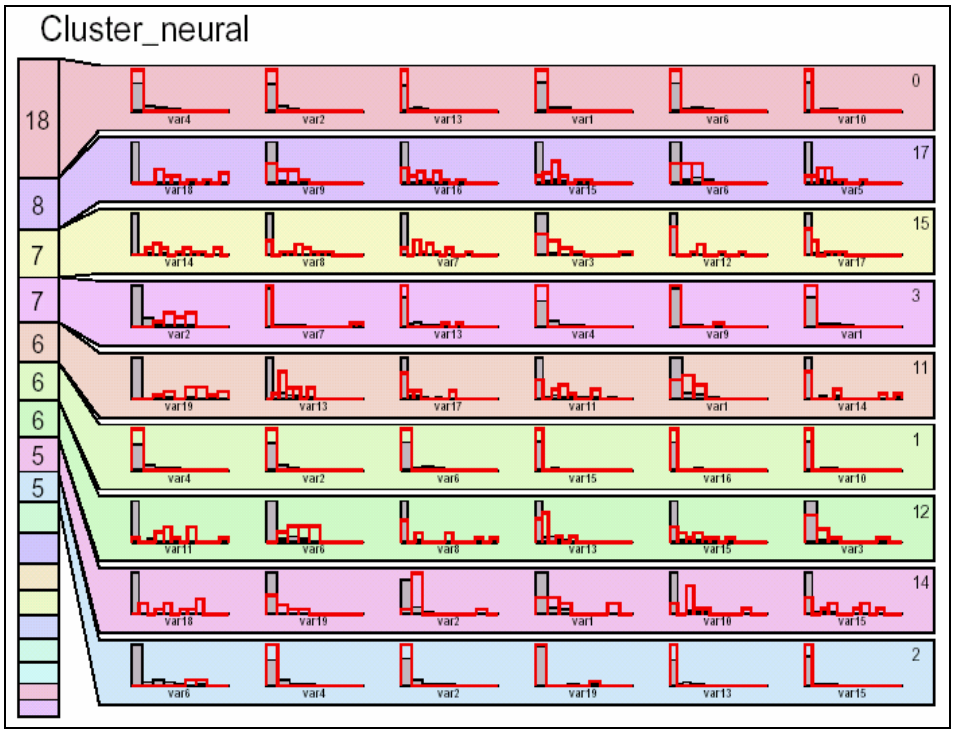


Figura 5.16: gráfico do “*Cluster neural*” para a base de cana de açúcar.

5.2 Segundo caso – base de dados de automóveis

Esta base de dados refere-se às informações a respeito de cerca de 400 automóveis (objetos) fabricados em três continentes (América, Europa e Ásia). As colunas (atributos) consideradas para cada automóvel, são, pela ordem: potência do motor em *horsepower* (cavalo força), ano de fabricação, número de cilindros do motor, cilindradas em *cubicinches*, nome do fabricante, peso em libras, aceleração em segundos para atingir a velocidade de 60 km/h, consumo de combustível em mpg (milhas por galão), nome do local de origem.

Os atributos expressos em texto exigem conversão para um tipo numérico, desta forma, foi empregada a tabela 5.1 de conversão dos atributos desse tipo para códigos inteiros arbitrários obedecendo simplesmente uma numeração seqüencial dos atributos ordenados alfabeticamente.

Tabela 5.1 - Conversão de atributos texto da base de automóveis.

Nome origem		Nome fabricante			
USA	1	*amc*	1	*mercedes*	16
Europa	2	*audi*	2	*mercury*	17
Japão	3	*bmw*	3	*nissan*	18
		buick	4	*oldsmobile*	19
		cadillac	5	*opel*	20
		capri	6	*peugeot*	21
		chevrolet	7	*plymouth*	22
		chrysler	8	*pontiac*	23
		datsun	9	*renault*	24
		dodge	10	*saab*	25
		fiat	11	*subaru*	26
		ford	12	*toyota*	27
		hi	13	*triumph*	28
		honda	14	*volvo*	29
		mazda	15	*volkswagen*	30

5.2.1 Resultados do *SearchCluster* sobre a base de automóveis

A base de dados de automóveis, com 392 objetos, não se presta a um estudo supervisionado, pois não apresenta informações sobre o número de grupos esperado como resultado do processo de segmentação. Portanto, primeiramente foi visualizada a disposição dos objetos em um plano bidimensional, conforme pode ser visto na Figura 5.17 e o dendrograma dos objetos, apresentado na Figura 5.18.

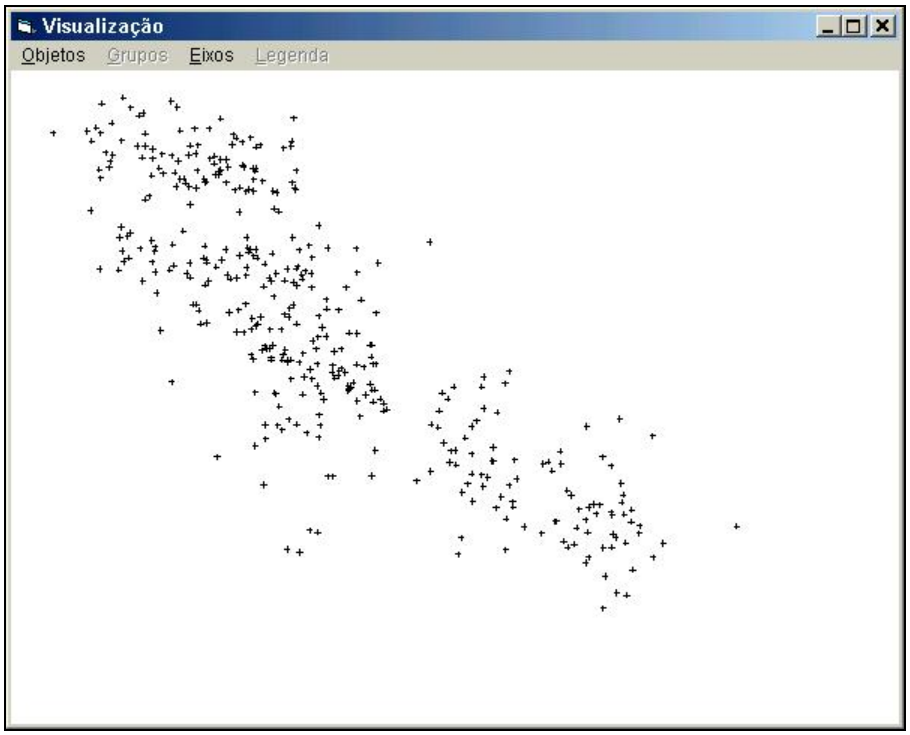


Figura 5.17: visualização dos objetos em gráfico de coordenadas estrela.

A distribuição espacial dos dados apresenta uma forma alongada, onde podem ser notados dois grupos levemente destacados, tendência também apresentada no dendrograma, mas não propicia conclusões mais avançadas, pois o formato e proximidade dos grupos não permitem leituras definitivas. Dessa forma, outras execuções do software, com diferentes combinações de parâmetros se fazem necessárias para obtenção de conclusões com maior nível de segurança.

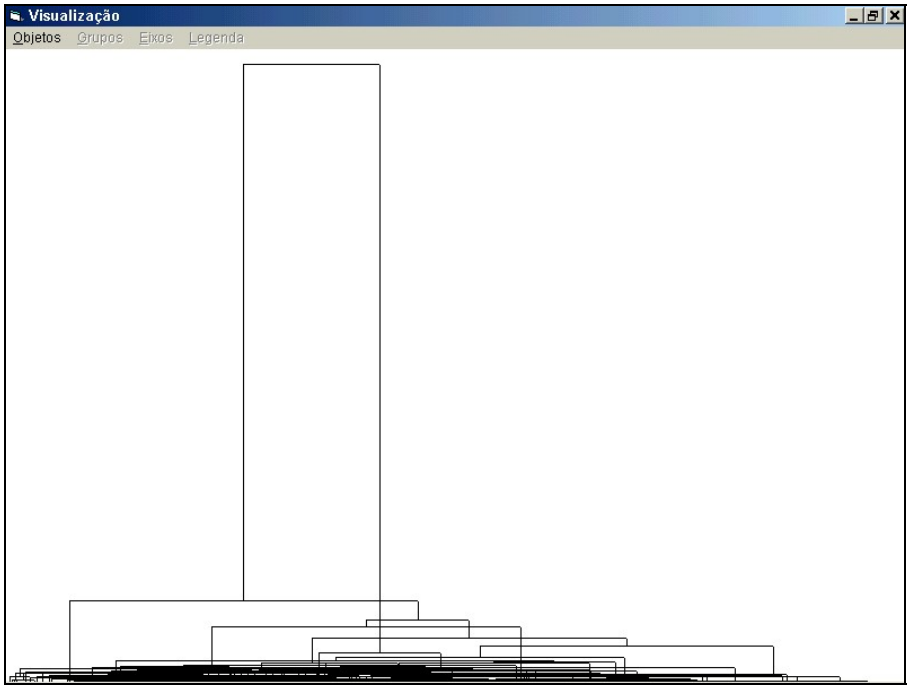


Figura 5.18: dendrograma da base de automóveis.

A base de dados foi, em seguida, submetida ao processo de segmentação do *SearchCluster*, que originou, então, a formação de dois grupos, como apresentado na Figura 5.19 e na Figura 5.20.

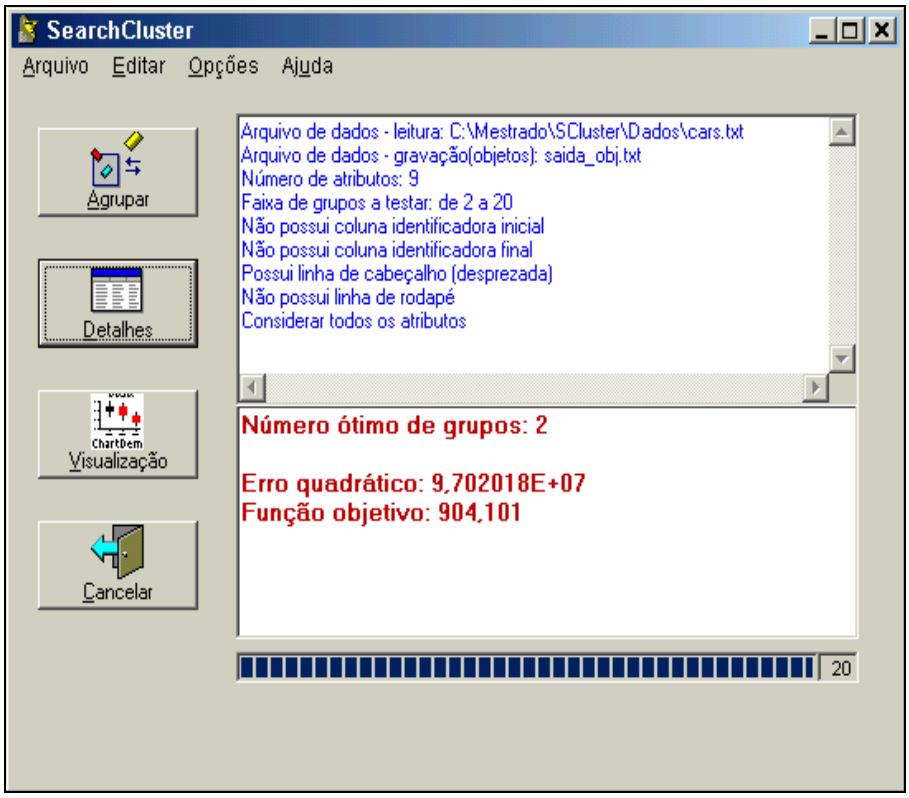


Figura 5.19: tela de resultados da segmentação da base de automóveis.

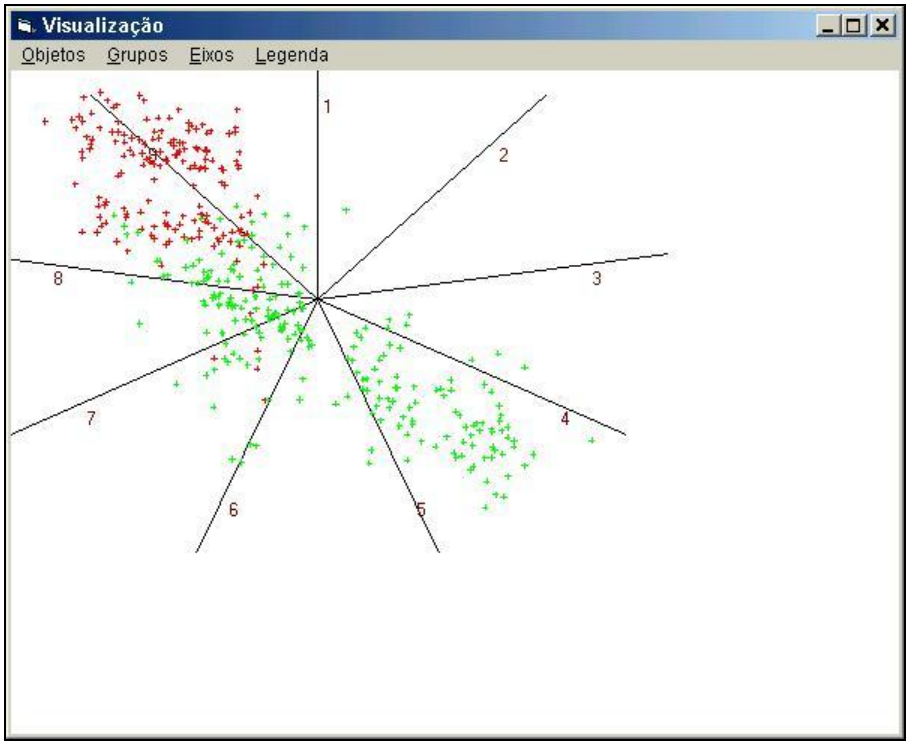


Figura 5.20: visualização dos grupos de objetos e dos eixos dimensionais.

A divisão da base em dois grupos provavelmente está ligada à origem de seus objetos. Ao ser analisado o arquivo de objetos produzido na saída do software, pode-se contar 159 automóveis norte-americanos no grupo 1 contra 86 no grupo 2; dos 68 automóveis europeus, apenas 6 foram alocados no grupo 1; a totalidade dos automóveis fabricados no Japão foi alocada no grupo 2.

Para a base de automóveis também foram feitas comparações entre o *SearchCluster* e a implementação do Índice PBM, cujos resultados das diversas segmentações obtidas, podem ser visualizados na Figura 5.21 e na Figura 5.22.

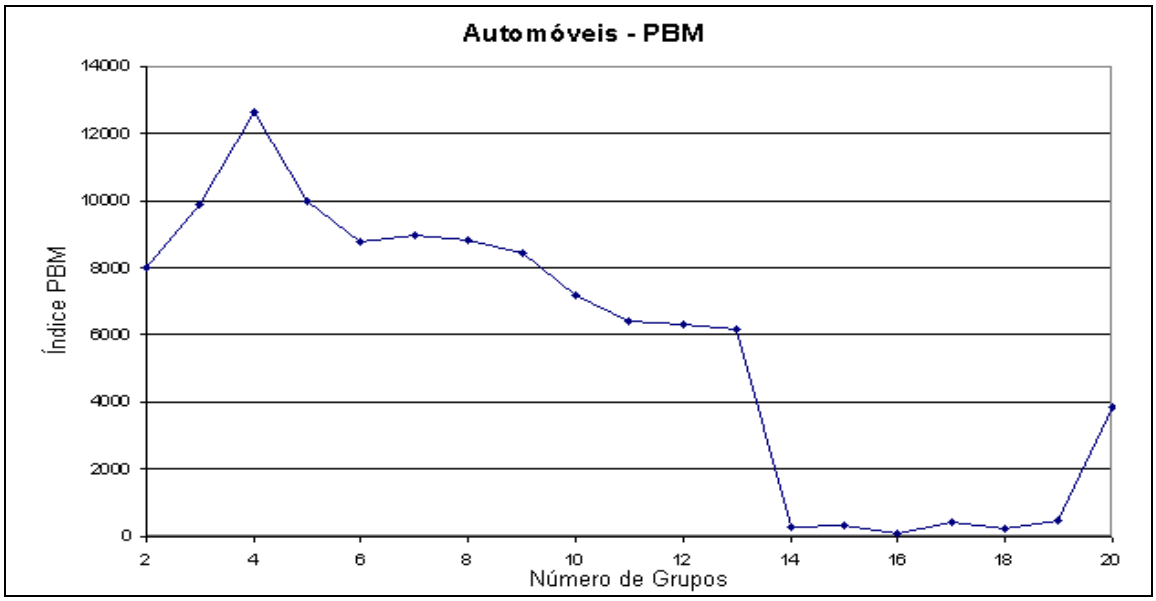


Figura 5.21: evolução do Índice PBM para a base de automóveis completa.

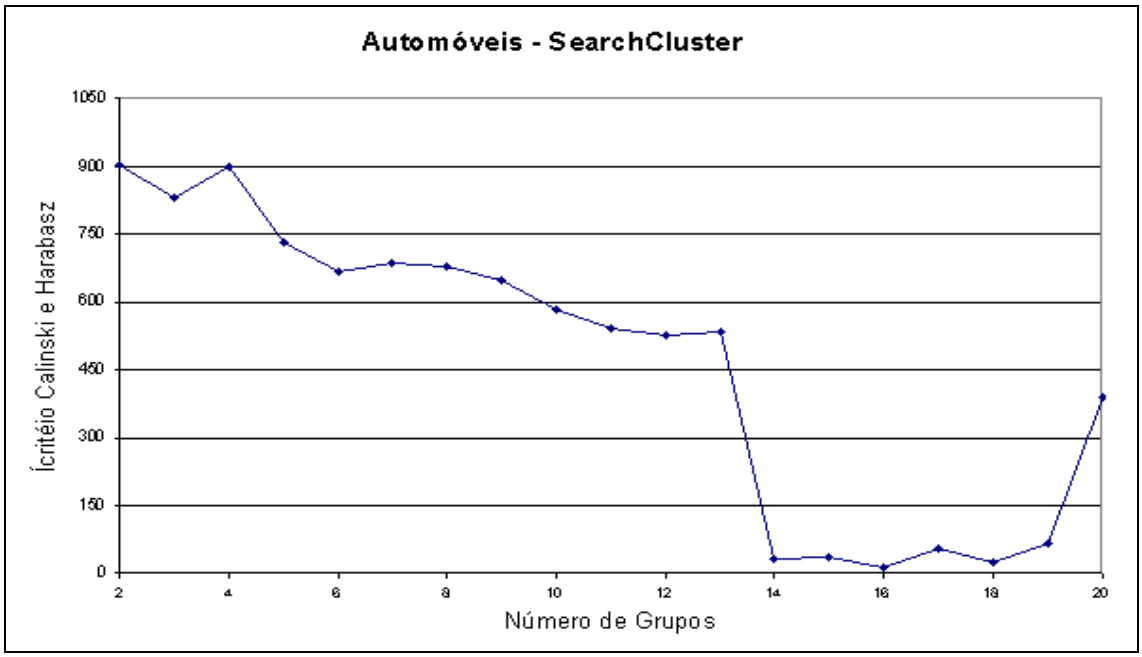


Figura 5.22: evolução do Critério *CalinsKi* e *Harabasz* para a base de automóveis completa.

O método clássico do Índice PBM apresentou como melhor segmentação para a base completa o número de 4 grupos, contra os 2 apontados pelo *SearchCluster*, mas, apesar das tendências contrárias nas primeiras segmentações, a partir de 4 grupos as curvas passam a ter um perfil muito coincidente, o que indica equivalência entre os dois indicadores utilizados.

Prosseguindo na análise, iniciou-se a supressão de alguns atributos que, devido às suas características categóricas, poderiam estar encobrindo alguma tendência mais elucidativa dos grupos formados, procurando-se, desta forma, diminuir o número de variáveis de forma que sua seleção contemple tanto a sua relevância como seu poder de discriminação em face ao problema em estudo. Primeiramente foi desconsiderado apenas o atributo identificador da origem dos automóveis. O resultado da nova segmentação redundou em dois grupos, com os valores do erro quadrático e da função objetivo perseguida pelo software apresentando ínfima alteração, conforme demonstrado na Figura 5.23, o que indica ser desprezível a influência do atributo origem dos automóveis.

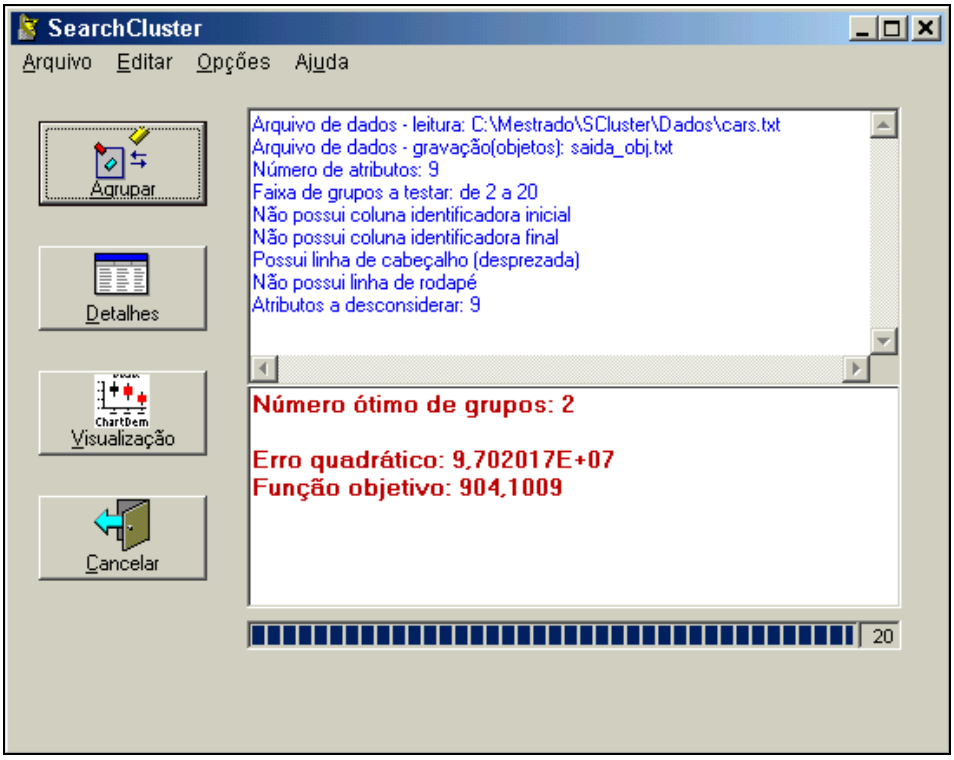


Figura 5.23: resultados da segmentação desprezando o atributo origem.

Os grupos formados, ainda, continuaram contando com a mesma quantidade de objetos, o que reforça o sentimento com relação ao atributo origem. Essas quantidades podem ser vistas na Figura 5.24, onde também pode ser observado o altíssimo valor do erro nos dois grupos, indicador de segmentação de baixa qualidade.

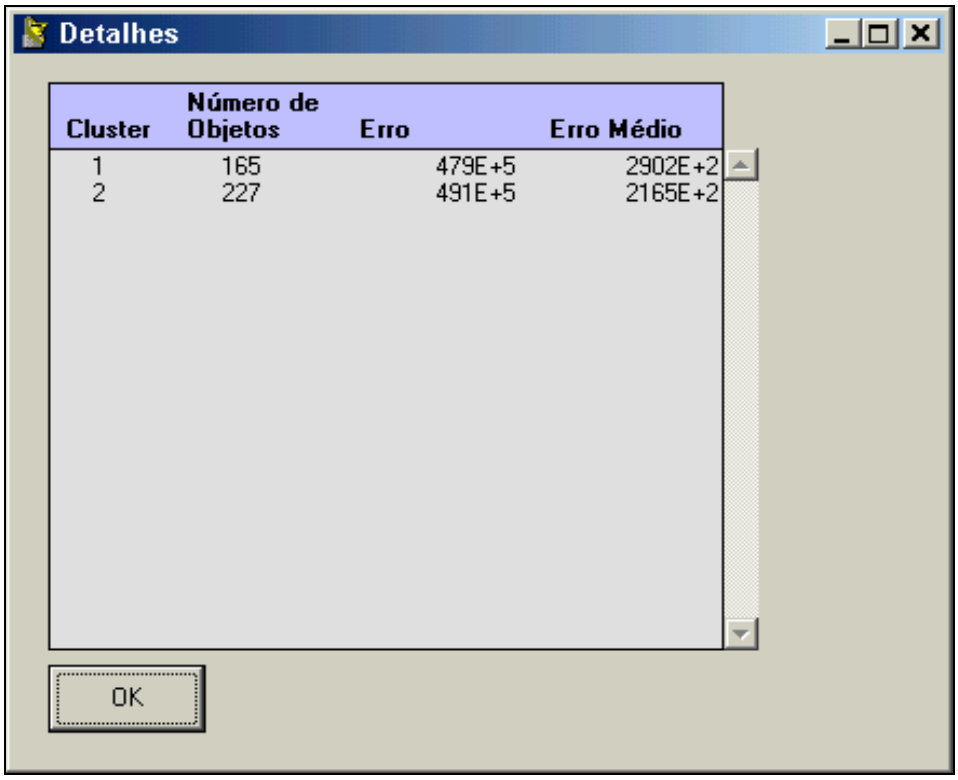


Figura 5.24: quantidades de objetos e erros dos grupos formados.

Continuando com o teste, foi suprimida também a coluna referente ao atributo nome do fabricante. O resultado da segmentação pode ser visto na Figura 5.25.

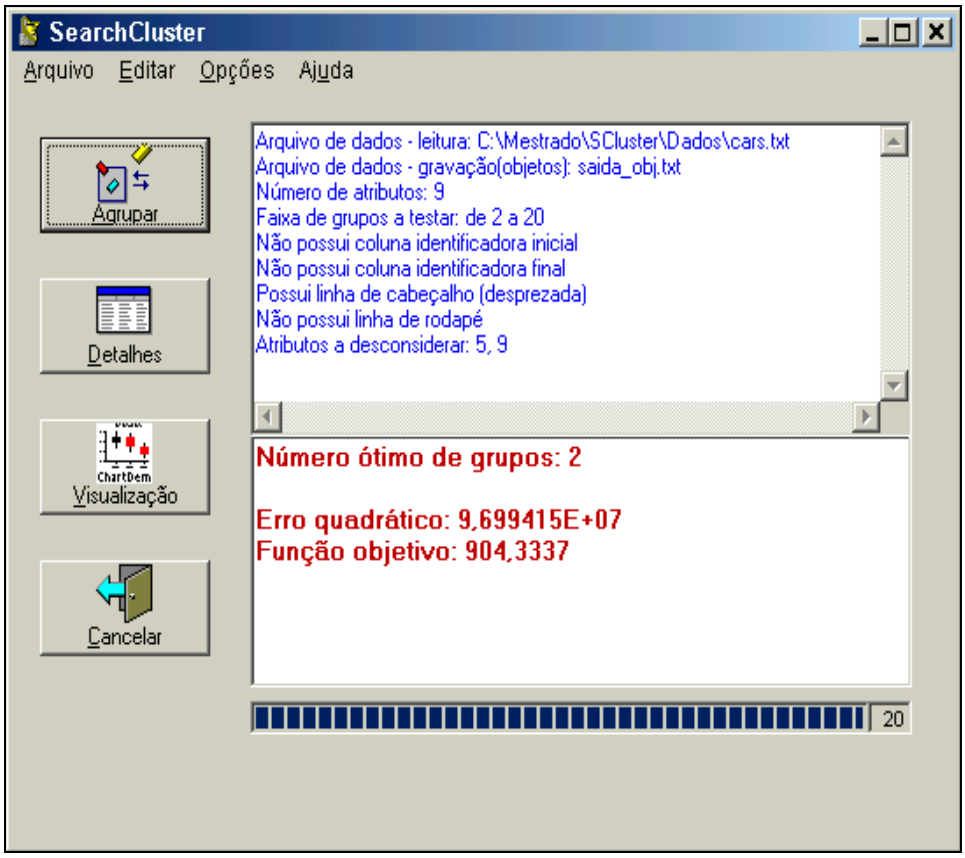


Figura 5.25: resultados da segmentação desprezando origem e fabricante.

Novamente, os resultados atingidos são muito parecidos em relação ao erro e função objetivo, inclusive os grupos formados mantêm as quantidades, 165 e 227, já obtidas na segmentação anterior, o que indica a influência quase nula desses atributos na segmentação da base. No entanto, a visualização dos objetos apresenta interessantes mudanças: o conjunto aproxima-se de uma forma mais arredondada, com os objetos dispostos em torno de um ponto central e refletindo uma tendência de divisão em três grupos, conforme pode ser observado na Figura 5.26.

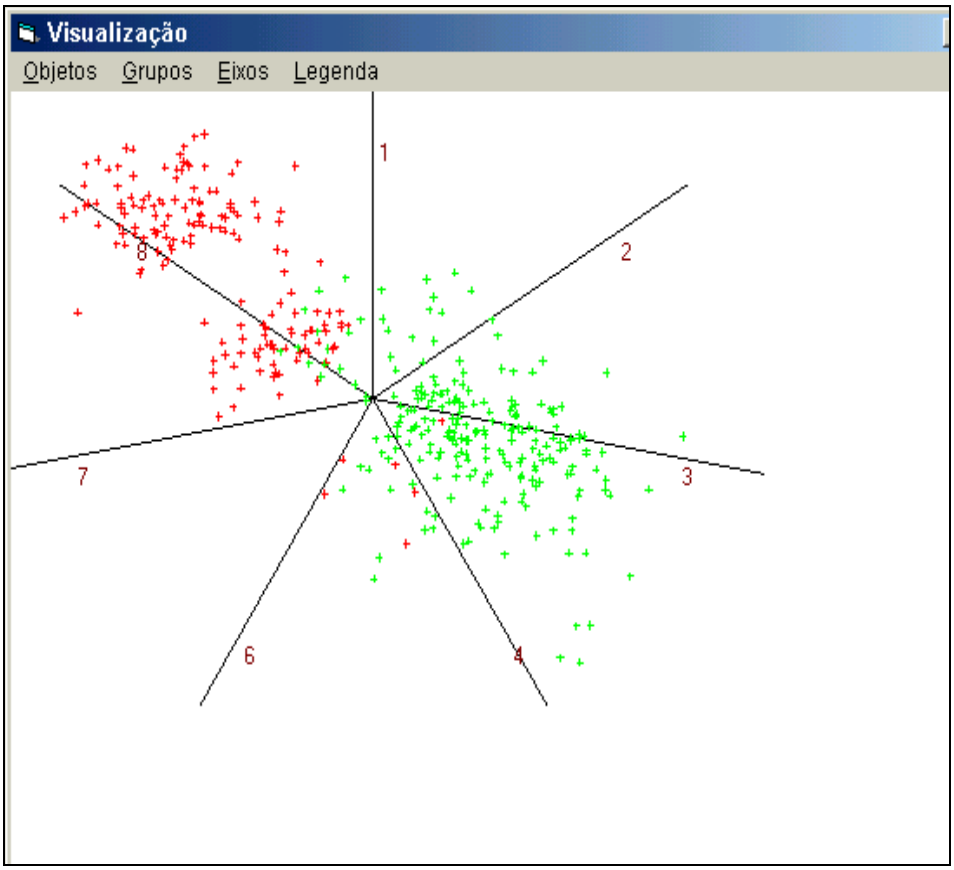


Figura 5.26: visualização dos objetos com supressão de dois atributos.

No último passo, foram adicionadas à lista de atributos desconsiderados na análise as colunas que não dizem respeito a características específicas do motor dos automóveis: ano de fabricação e peso. O resultado do teste realizado neste passo está apresentado na Figura 5.27. Neste processamento, a base foi segmentada em três grupos, apresentando, em relação aos testes anteriores, um erro quadrático geral muito menor e melhor resultado final da função objetivo.

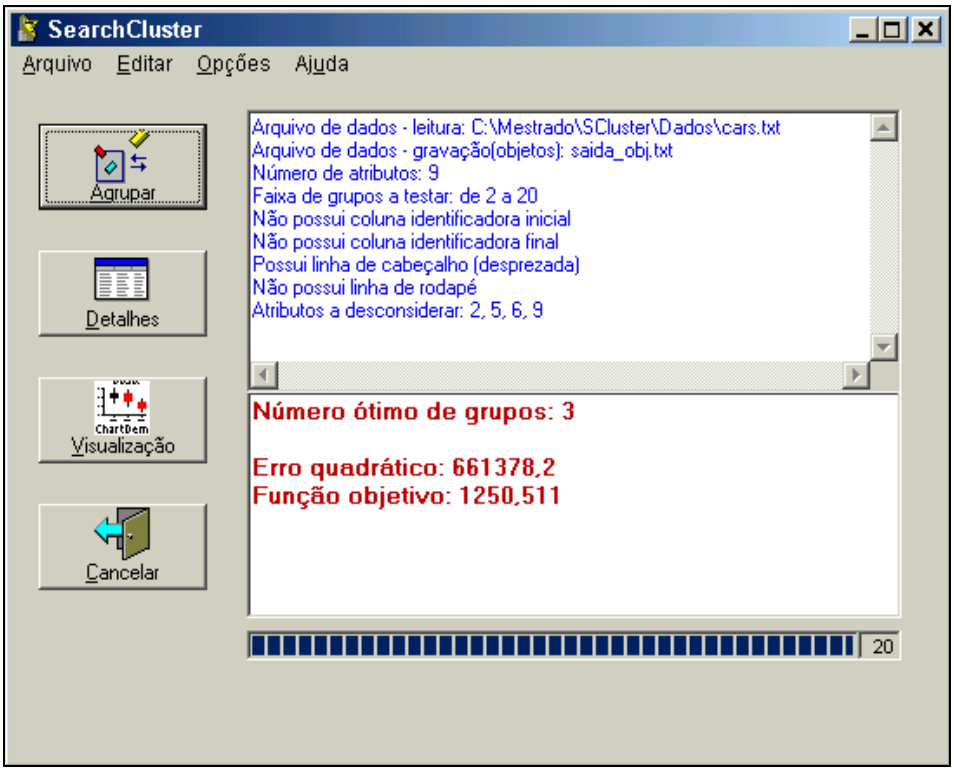


Figura 5.27: resultados da segmentação apenas com atributos de motor.

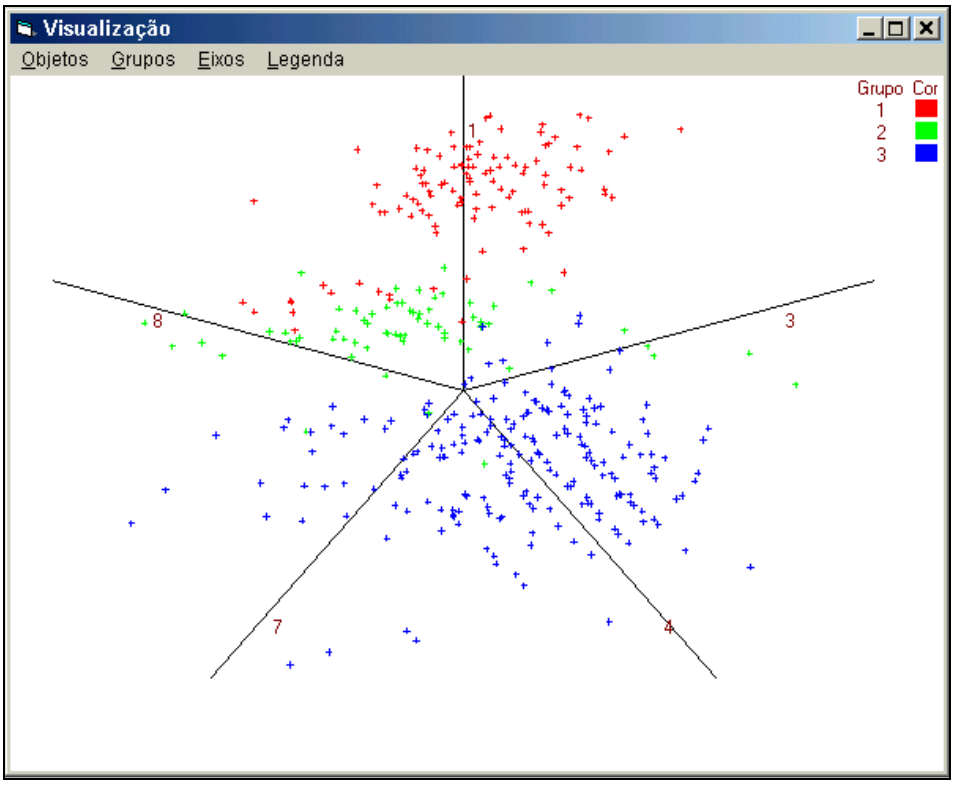


Figura 5.28: visualização dos objetos divididos em três grupos.

Embora tenham sido criados três grupos, o que poderia remeter a uma divisão por área geográfica (EUA, Europa e Japão) realizando contagens no novo arquivo de saída, percebe-se que os grupos formados mantêm tendências

prenunciadas já na primeira segmentação, conforme pode ser percebido pela observação da Figura 5.28. Dos automóveis fabricados nos EUA, um conjunto significativo mantém-se à parte dos demais: 114 objetos formam o primeiro grupo, sem ocorrências de automóveis de outra região; 68 objetos formam o segundo grupo, com apenas 2 objetos do Japão, 4 da Europa e 62 dos EUA. O terceiro grupo formado, contém uma distribuição mais equilibrada entre os objetos: 69 automóveis dos EUA, 64 da Europa e 77 do Japão.

5.2.2 Resultados do *gCLUTO* sobre a base de automóveis

Inicialmente, foi adotado o método aglomerativo do *gCLUTO* para dividir a base de automóveis em dois grupos. Os resultados, como as quantidades de 132 e 260 objetos em respectivamente por grupo e outras informações estatísticas, estão apresentados na Figura 5.29.

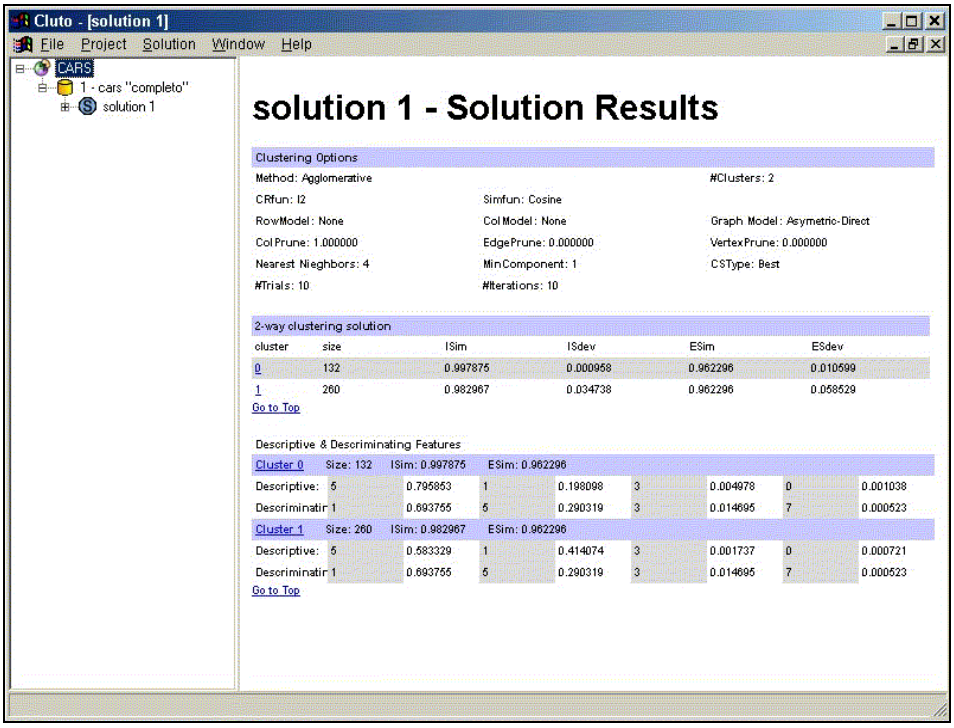


Figura 5.29: resultados em modo texto da segmentação pelo *gCLUTO*.

A segmentação realizada pelo *gCLUTO* chega a um resultado que pode ser considerado bastante coerente com as características conhecidas da base: os dois picos apresentam-se nitidamente separados; o conjunto de automóveis dos EUA possui 255 automóveis e o pico que apresenta baixa divergência interna refere-se

justamente ao grupo que contém 260 objetos; o pico que contém 132 objetos apresenta alta divergência interna, compatível com uma distribuição de objetos quase igualitária entre automóveis europeus e japoneses. Estas conclusões podem ser obtidas pela análise da Figura 5.30

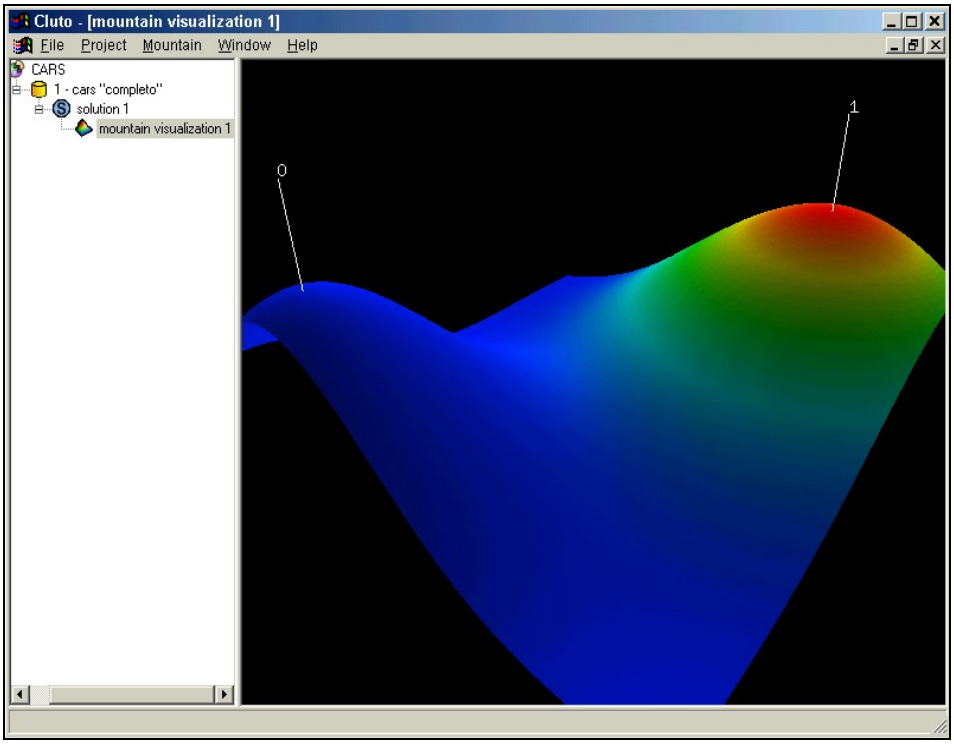


Figura 5.30: visualização de montanha da segmentação da base de automóveis.

Como já foi demonstrado, as colunas que não dizem respeito diretamente às características dos motores dos automóveis não tem influência relevante no processo de segmentação, passou-se diretamente ao processamento de uma base de dados onde foram suprimidas as colunas origem, nome do fabricante, ano de fabricação e peso do automóvel. Embora a visualização de montanha com os dois grupos formados, vista na Figura 5.31, não apresente alterações significativas em relação à anterior, os resultados obtidos pela segmentação, mostrados na Figura 5.32, apresentam uma interessante coincidência entre as quantidades de objetos dos grupos formados nesta execução do *gCLUTO* e àquelas apresentadas pelas segmentações da base completa e sem os atributos categóricos realizadas pelo *SearchCluster*, conforme visto no estudo apresentado no item anterior.

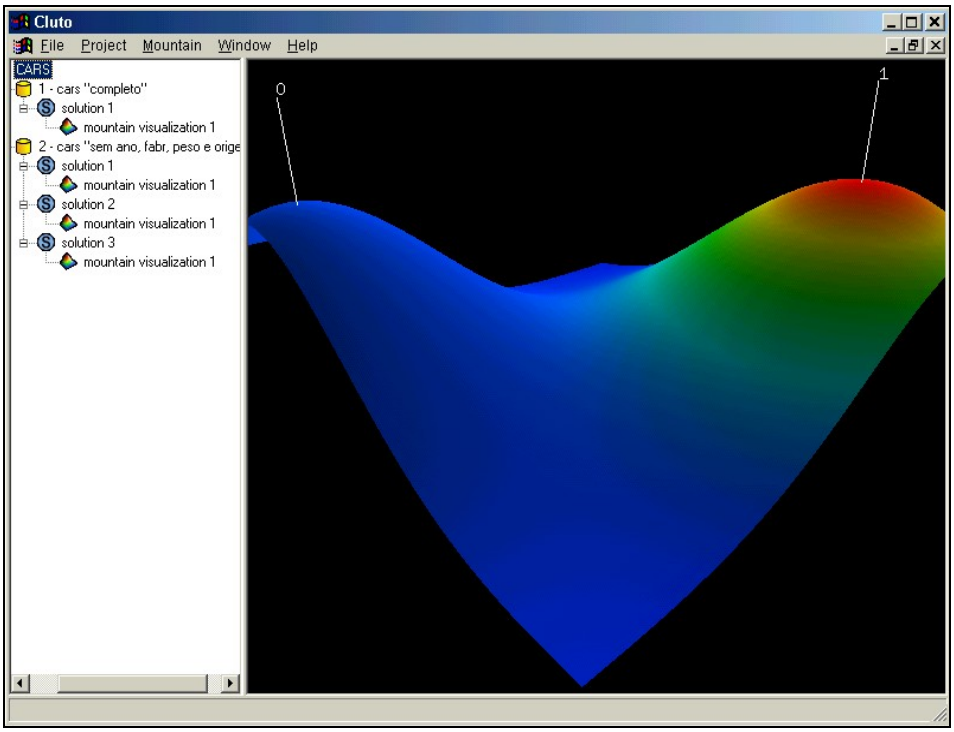


Figura 5.31: visualização da base reduzida em dois grupos.

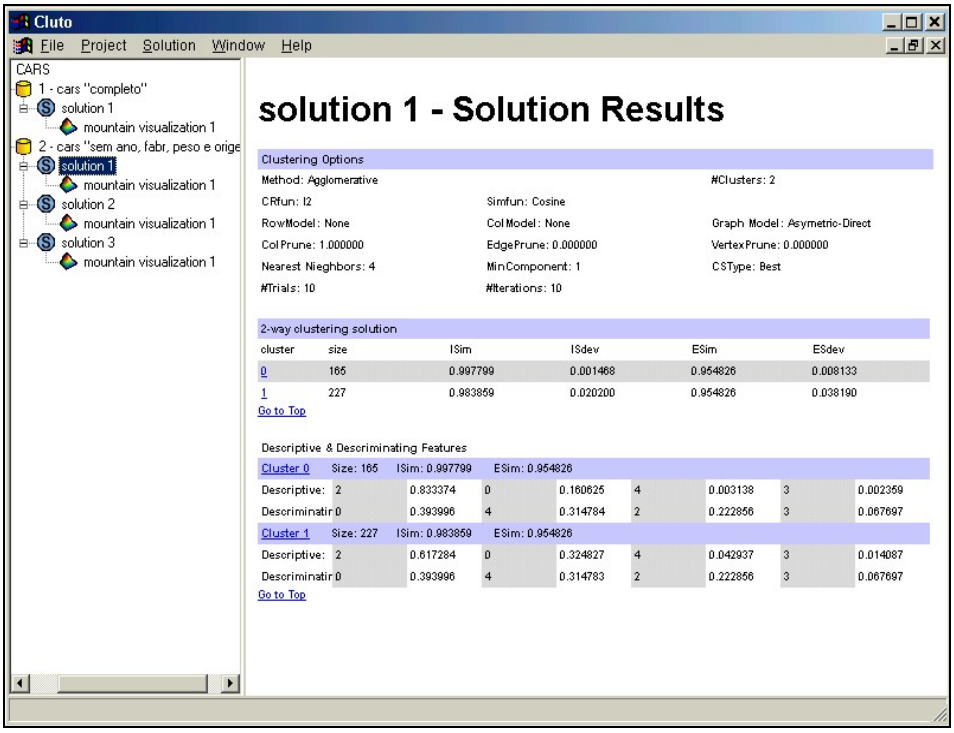


Figura 5.32: resultados da segmentação da base reduzida em dois grupos.

No terceiro teste da base de automóveis com o *gCLUTO*, a base foi dividida em três grupos segundo dois métodos, o aglomerativo e o *Repeated Bisection*. Com o primeiro método os resultados não apresentaram alterações de maior interesse. O segundo método, porém, apresentou alguma coincidência com os resultados apurados pelo *SearchCluster*: um grupo com exatos 114 objetos, apesar de outros dois,

divergentes, com 183 e 95 objetos respectivamente, conforme apresentado na Figura 5.33.

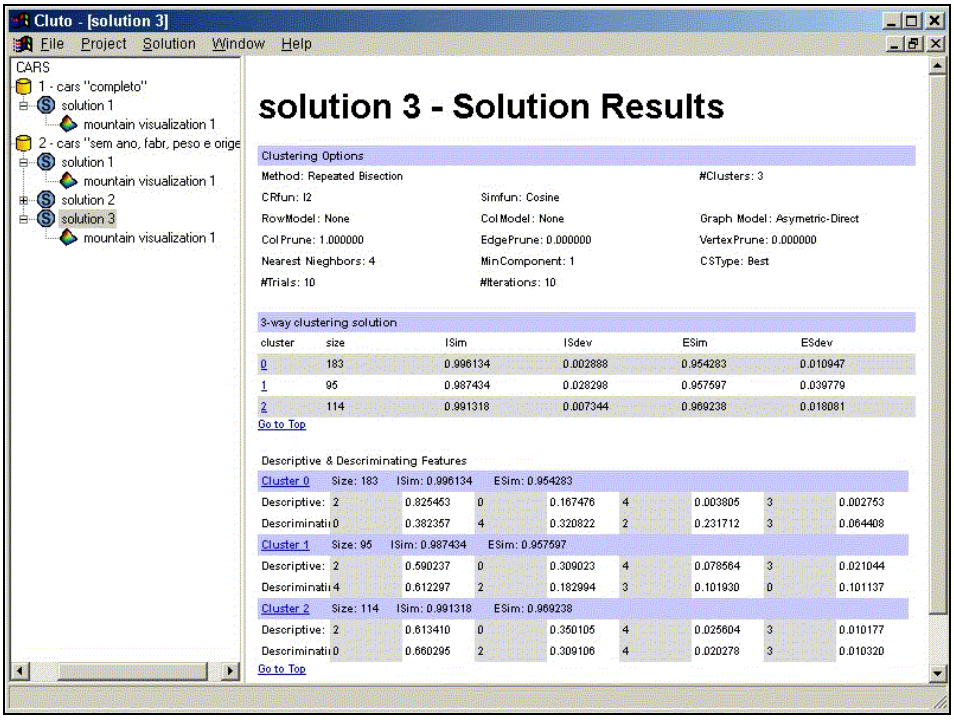


Figura 5.33: resultados da segmentação pelo método *Repeated Bisection*.

A visualização de montanha formada pela base de automóveis segmentada em ter grupos pelo método *Repeated Bisection*, como pode ser vista na Figura 5.34, reforça a idéia de coincidência entre os testes: o menor grupo, com apenas 95 objetos, apresenta a menor divergência interna, provavelmente contendo apenas automóveis fabricados nos EUA; o pico referente ao grupo de 114 objetos apresenta uma coloração que atinge o verde, denotando uma divergência interna média, compatível com a distribuição dos objetos apresentada pelo gráfico de coordenadas estrela da Figura 5.28, onde pode ser visto que a maioria dos objetos do grupo 1 dispõe-se relativamente próximos, mas alguns estão bastante distantes do centro; o pico referente ao grupo com 183 objetos apresenta alta divergência interna, provavelmente por contar com automóveis das três origens diferentes.

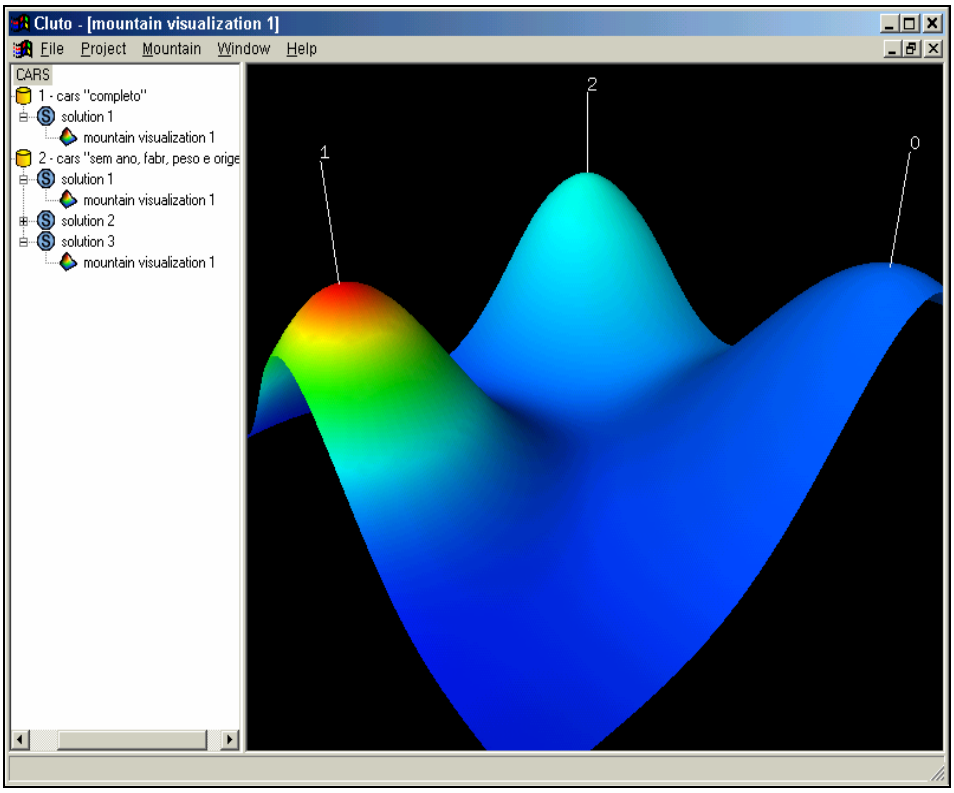


Figura 5.34: visualização da segmentação em três grupos.

5.2.3 Resultados do *Enterprise Miner* sobre a base de automóveis

Analisando a base de automóveis completa, ou seja, levando em conta todos os seus atributos originais, com o *Enterprise Miner*, software que não exige a prévia informação do número de grupos esperado, fazendo a descoberta da melhor formação pelo emprego das técnicas de segmentação nele embutidas, chega-se à divisão do conjunto de dados em dois grupos, um razoavelmente maior que o outro, em coincidência com o resultado obtido pela aplicação do *SearchCluster*.

A Figura 5.35, que apresenta a visualização dos resultados obtidos pelo *Enterprise Miner*, contém um gráfico de pizza que permite perceber a proporção dos grupos formados e um gráfico de dispersão com as médias dos atributos da base.

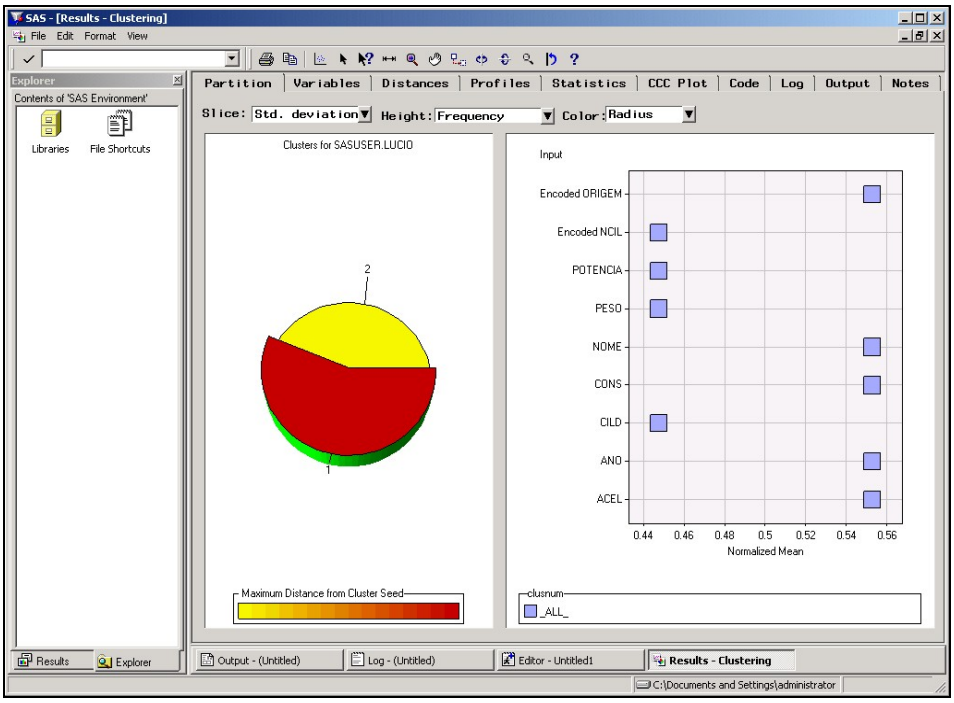


Figura 5.35: segmentação da base de automóveis obtida pelo *Enterprise Miner*.

O *Enterprise Miner* apresenta um relatório com informações que descrevem características básicas dos atributos componentes da base analisada, como tipo e importância relativa do atributo no processo de segmentação. A forma e conteúdo de tal relatório podem ser observados na Figura 5.36.

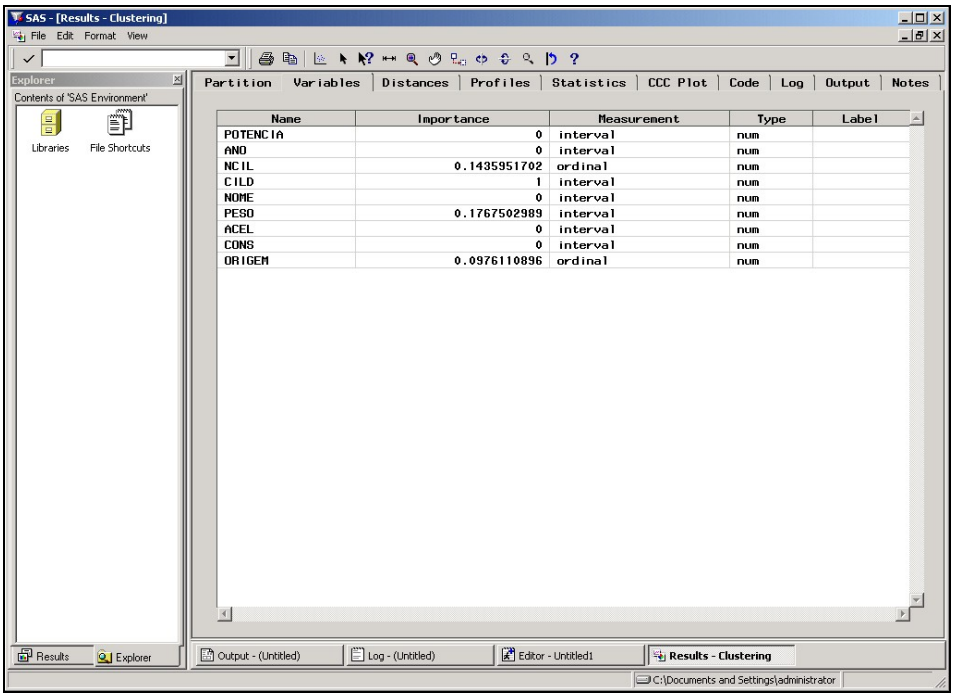


Figura 5.36: informações descritivas dos atributos da base de automóveis.

O gráfico do CCC pode ser visto na Figura 5.37. Nele é possível verificar a evolução do critério em função do número de grupos testados pelo *software*, em busca

do melhor agrupamento. Note-se que há um salto no valor do critério obtido para dois grupos e outra quebra de continuidade na passagem de 22 para 24 grupos. O valor do critério, em função do número de grupos formados, aponta para 2 como melhor agrupamento.

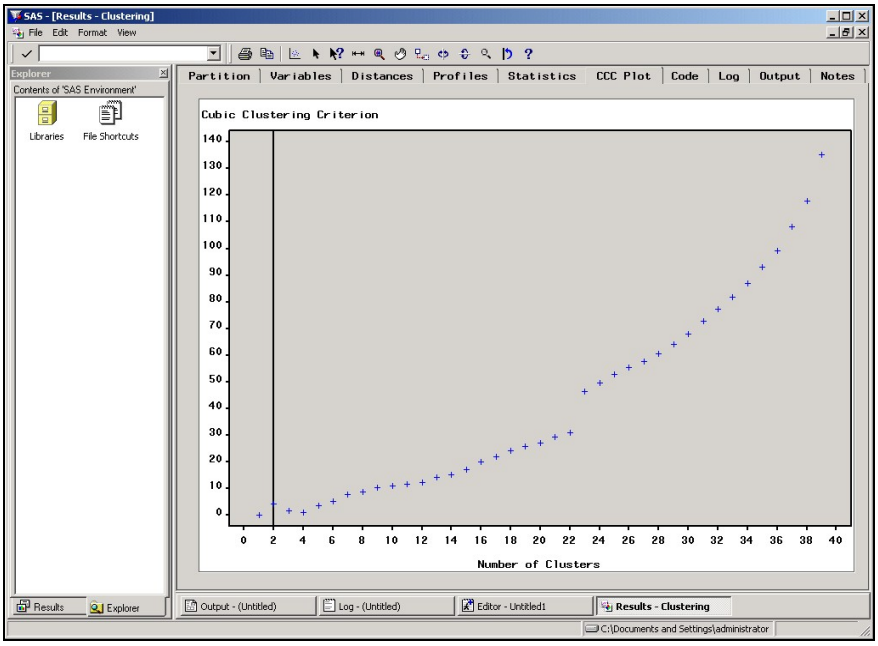


Figura 5.37: gráfico do CCC das segmentações da base de automóveis.

Na Figura 5.38 é apresentado um gráfico de árvore que permite visualizar com as regras criadas e adotadas pelo software, em função dos atributos, na formação dos grupos obtidos na melhor segmentação realizada. O gráfico também dá a noção exata da contribuição de cada atributo no processo de agrupamento.

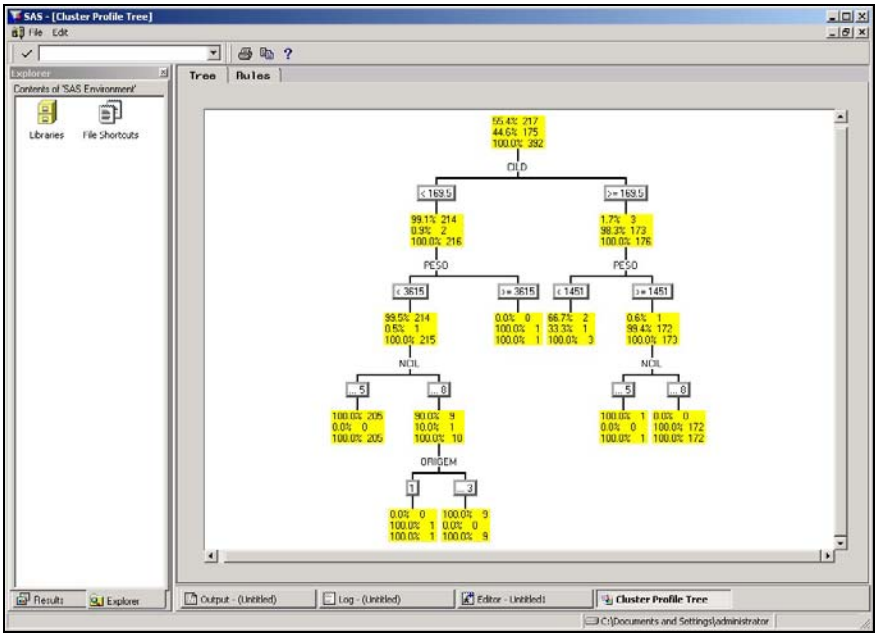


Figura 5.38: gráfico de árvore com as regras utilizadas para a segmentação.

Com o intuito de convalidar os resultados obtidos pelo *SearchCluster*, foram suprimidos alguns atributos, reduzindo a base ao mesmo conjunto testado anteriormente.

Com a retirada das colunas relativas à origem e ao peso do automóvel, classificados entre os atributos de maior importância para a formação dos grupos, nome do fabricante e ano de fabricação, a base passa a contar apenas com atributos que refletem características do motor dos automóveis e, novamente, o resultado obtido pela segmentação é bastante semelhante ao que foi apresentado pelo *SearchCluster*, conforme demonstrado na Figura 5.39.

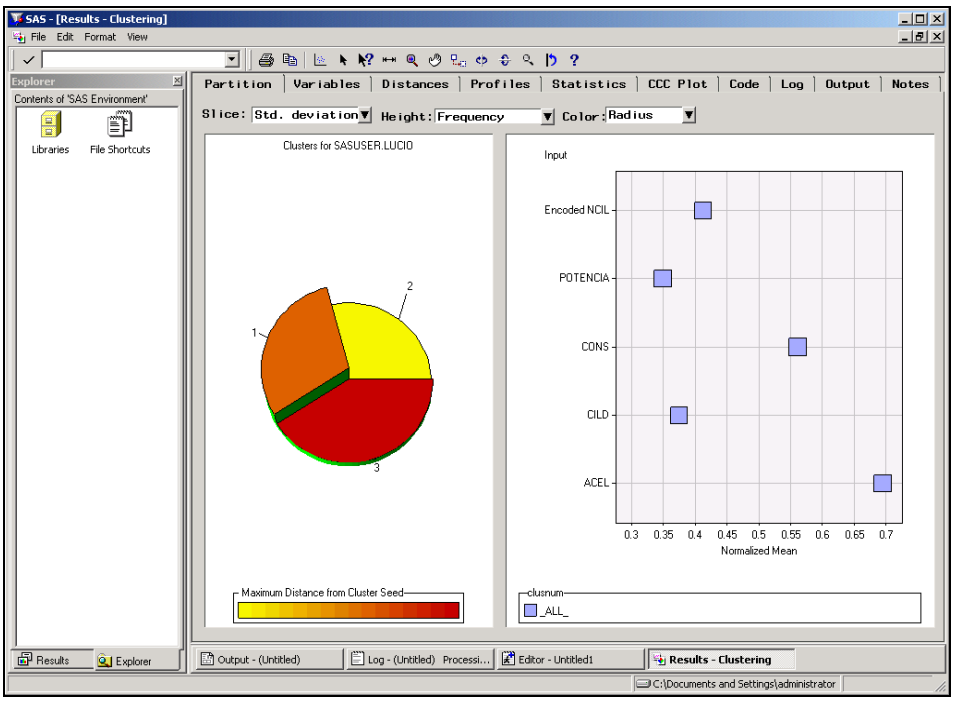


Figura 5.39: resultados da nova segmentação para base com 5 atributos.

Ao observar o relatório descritivo dos atributos da base reduzida, apresentado na Figura 5.40, é possível notar que as colunas remanescentes revestem-se de uma importância melhor distribuída, permanecendo apenas um atributo, a aceleração com importância 0, bastante diferente da configuração anterior onde mais de metade dos atributos, 5 exatamente, apresentavam importância nula para formação dos grupos.

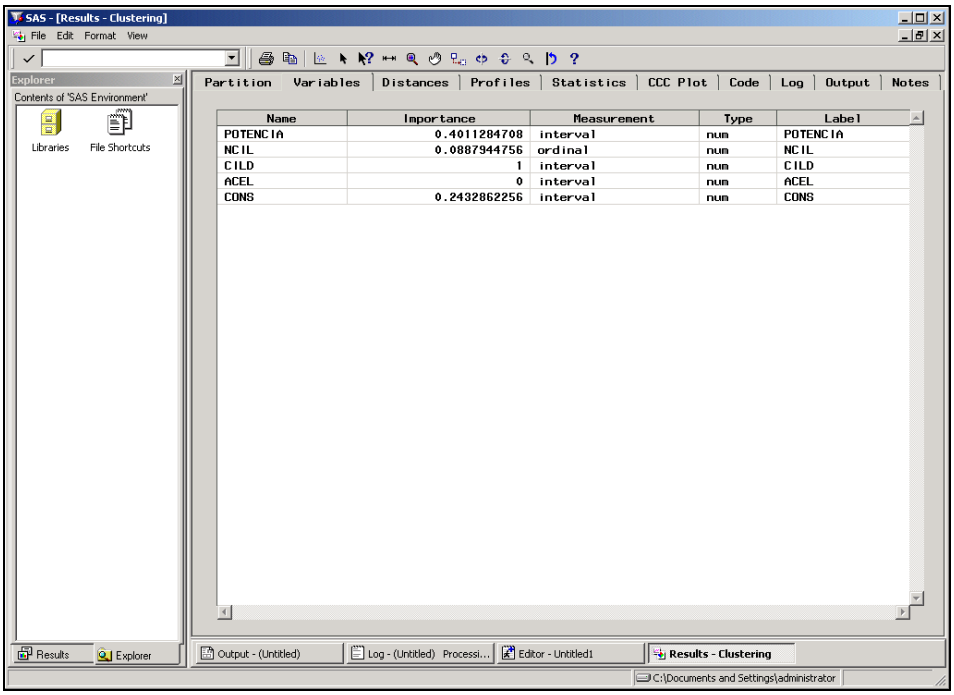


Figura 5.40: informações descritivas dos atributos da base reduzida.

O gráfico de apresentação do CCC continua mostrando um salto significativo, agora em três grupos, e uma pequena ruptura de 7 para 8 grupos, como pode ser visto na Figura 5.41.

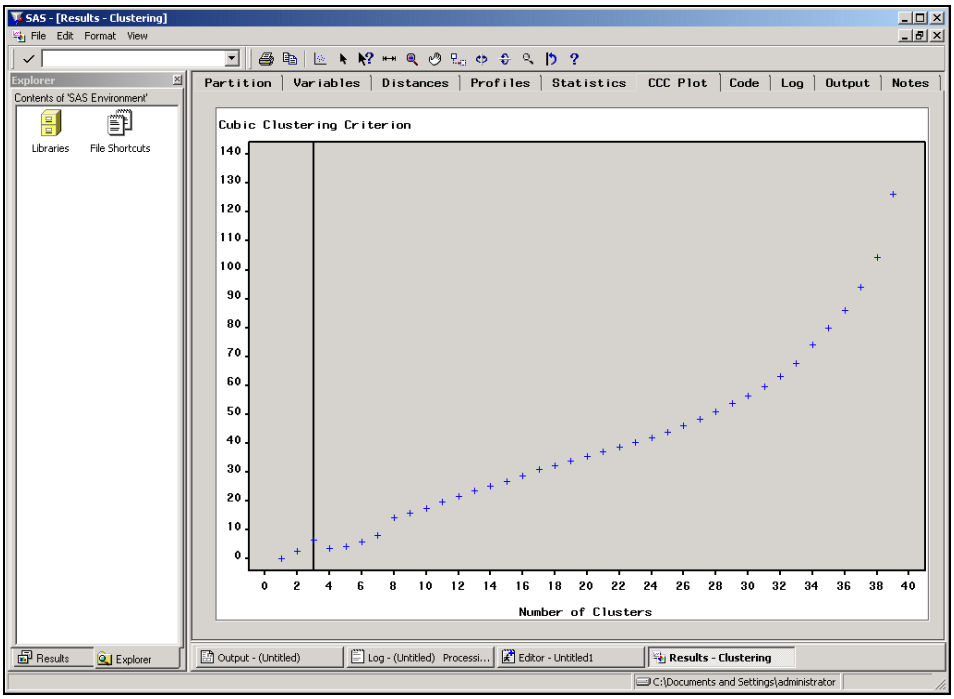


Figura 5.41: gráfico demonstrativo da evolução do CCC para a base reduzida.

O gráfico de árvore, para a base reduzida, torna-se mais complexo, como pode ser observado na Figura 5.42, pois a maioria dos atributos passou a ter

importância significativa na construção dos grupos e aparecem em diversos níveis na composição das regras do agrupamento.

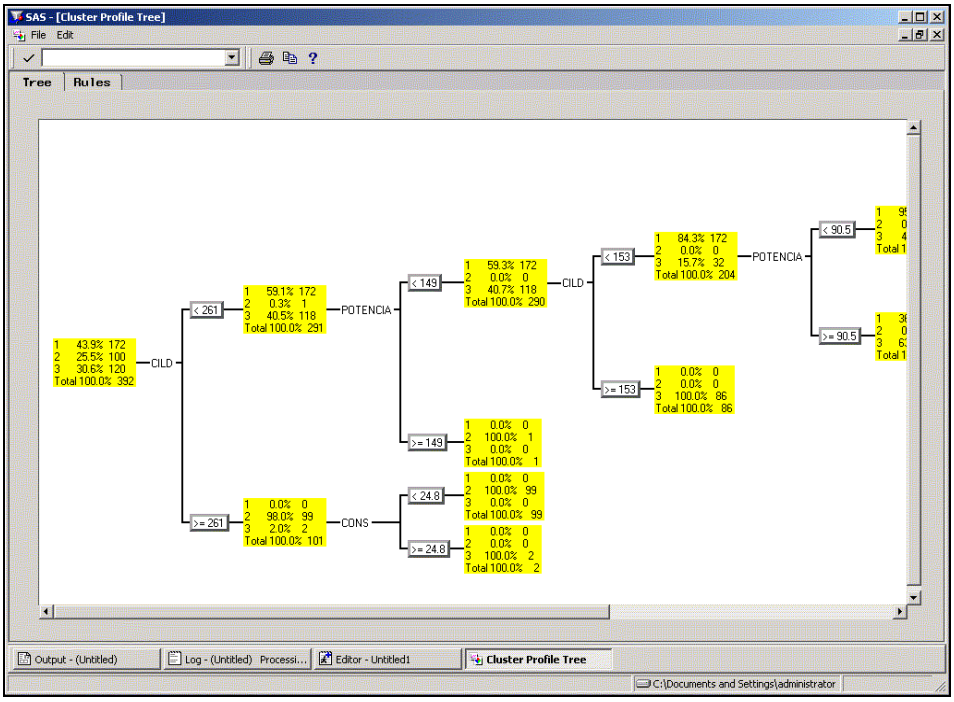


Figura 5.42: gráfico de árvore em forma horizontal para a base reduzida.

5.2.4 Resultados do *Mineset* sobre a base de automóveis

A segmentação da base de dados de automóveis, com todos os atributos iniciais, realizada pelo *Mineset*, aproximou-se bastante dos testes com o *SearchCluster*: experimentando de um a dez grupos, a melhor formação conseguida foi composta de dois grupos, com 218 e 174 objetos respectivamente. A Figura 5.43 apresenta a saída, em forma de relatório, com os resultados do primeiro processamento.

```
Fitness = 0.5336                          Result of clustering:
1 clusters: fitness = 0.5336299           ----------------------------------------
2 clusters: fitness = 0.2802967
3 clusters: fitness = 0.196839            Overall root-mean-squared distance from
4 clusters: fitness = 0.1735968           record to centroid: 0.3097 +- 0.1318
5 clusters: fitness = 0.1597568           RMS distance from records to each centroid:
6 clusters: fitness = 0.1488458             Cluster 1-A: 0.2389 +- 0.1071
7 clusters: fitness = 0.1428551             Cluster 1-B: 0.3745 +- 0.1154
8 clusters: fitness = 0.1388395           done.
9 clusters: fitness = 0.1345022
10 clusters: fitness = 0.1290762
Chose 2 clusters (choice point = 0.3313531)
```

Figura 5.43: relatório texto do processo de segmentação da base de automóveis.

Na Figura 5.44, onde mais uma vez pode-se ver que foram dois os grupos formados, 1-A e 1-B, também é possível observar, por exemplo, o comportamento da variável cilindradas, dentro do conjunto total e em cada grupo formado.

O valor máximo da variável cilindradas atinge o valor de 455 *cubicinches* e o valor mínimo 11, nos dois grupos. Para a população total, 25% dos automóveis estão entre 267 e 455 *cubicinches*, 50% entre 105 e 267 e os 25% restantes de 11 a 105. A distribuição no primeiro grupo, 1-A, não é tão equilibrada, com metade dos objetos do grupo numa estreita faixa que vai de 91 a 131 *cubicinches*. O segundo grupo, 1-B, com 75% acima de 232 *cubicinches*, dá a noção de ser composto majoritariamente por carros norte-americanos, notadamente possuidores das maiores cilindradas.

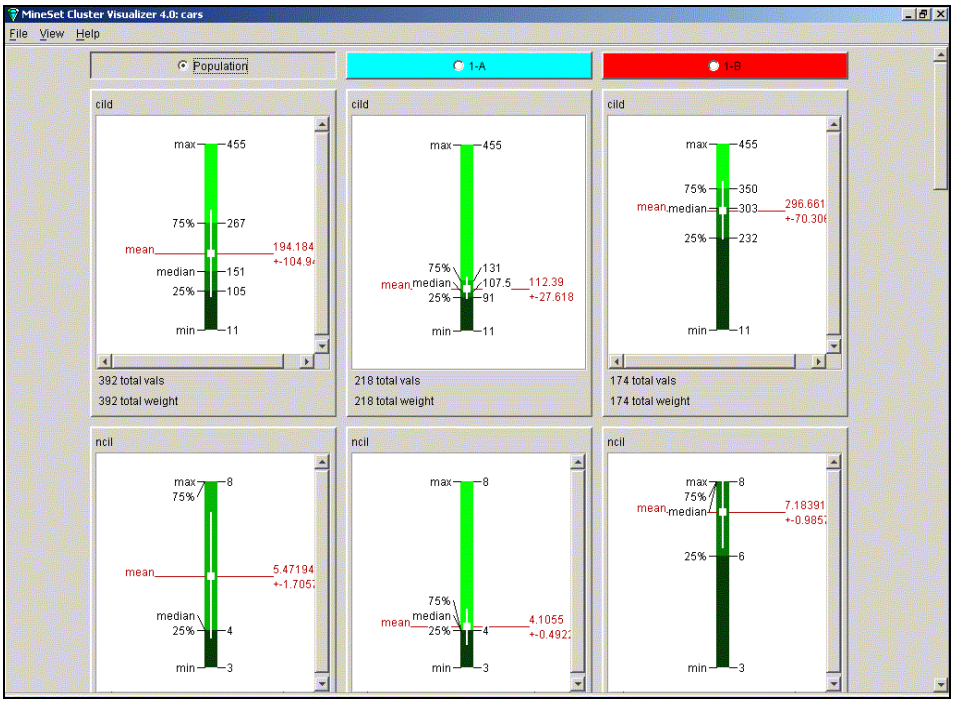


Figura 5.44: visualização estatística da segmentação da base de automóveis.

Nos testes com o *Mineset*, o passo seguinte constituiu-se na retirada dos 4 atributos não inerentes ao motor dos automóveis. Após a segmentação, a divisão em dois grupos permanece, não atingindo a formação esperada de três grupos, mas o grupo que deveria referir-se ao grupo de automóveis americanos aumenta, como pode ser notado na Figura 5.45, provavelmente por ter recebido automóveis com valores de cilindrada mais altos, vindos do outro grupo, onde predominam automóveis europeus e japoneses.

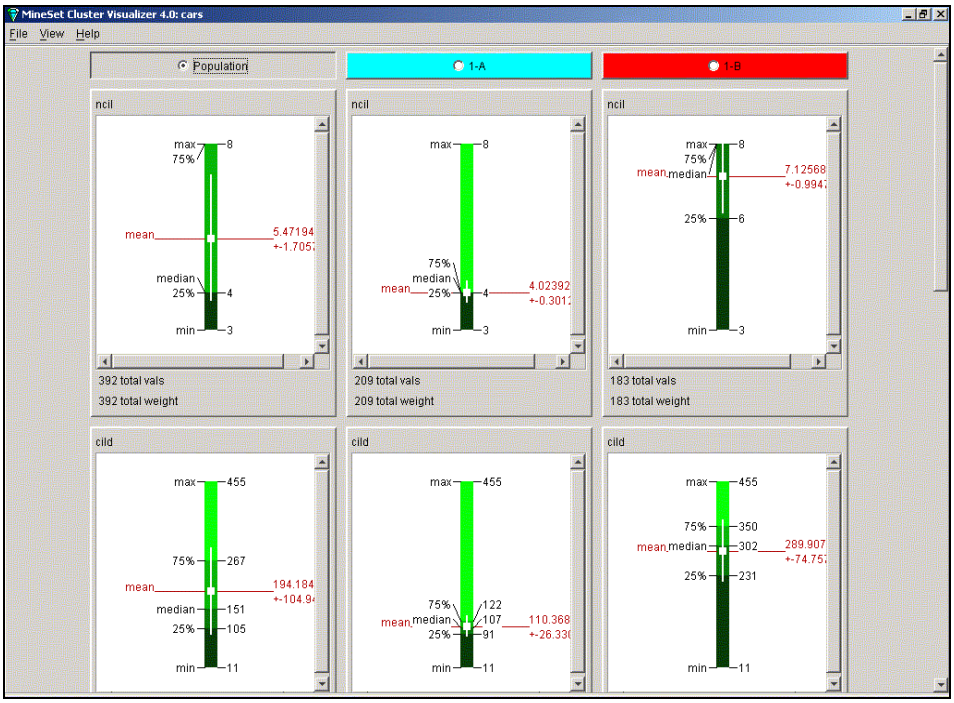
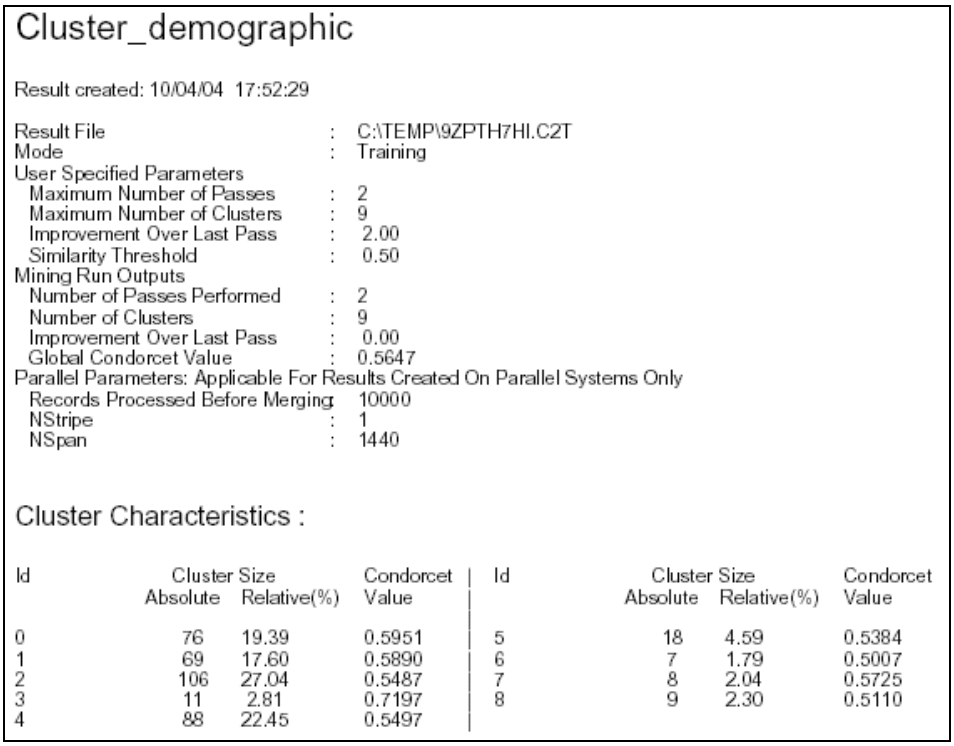


Figura 5.45: visualização estatística da segmentação da base reduzida.

5.2.5 Resultados do *Intelligent Miner* sobre a base de automóveis

Os resultados apresentados pelo *Intelligent Miner para a* segmentação da base de dados de automóveis, com o emprego dos dois métodos, não acompanham a mesma linha dos outros testes realizados. Adotando, inicialmente, os valores padrão para os parâmetros de entrada do método demográfico, a base completa, com todos os seus atributos originais, foram obtidos 9 grupos, mesmo valor do parâmetro "número máximo de grupos", como pode ser observado na Figura 5.46, donde se conclui que o software tende a encontrar como número ótimo de grupos um valor muito acima do esperado para a base de automóveis. Este fato provavelmente deve ter sido ocasionado pelo agrupamento de automóveis de uma mesma região em grupos menores, influenciados pelos atributos referentes à potência, consumo e cilindradas, cujas amplitudes de 46 a 230, de 0,1 a 44 e de 11 a 455 respectivamente, são bastante elevadas para o conjunto considerado.



Cluster_demographic

Result created: 10/04/04 17:52:29

Result File	:	C:\TEMP\9ZPTH7HI.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	2
Maximum Number of Clusters	:	9
Improvement Over Last Pass	:	2.00
Similarity Threshold	:	0.50
Mining Run Outputs		
Number of Passes Performed	:	2
Number of Clusters	:	9
Improvement Over Last Pass	:	0.00
Global Condorcet Value	:	0.5647
Parallel Parameters: Applicable For Results Created On Parallel Systems Only		
Records Processed Before Merging		10000
NStripe	:	1
NSpan	:	1440

Cluster Characteristics :

Id	Cluster Size Absolute	Relative(%)	Condorcet Value	Id	Cluster Size Absolute	Relative(%)	Condorcet Value
0	76	19.39	0.5951	5	18	4.59	0.5384
1	69	17.60	0.5890	6	7	1.79	0.5007
2	106	27.04	0.5487	7	8	2.04	0.5725
3	11	2.81	0.7197	8	9	2.30	0.5110
4	88	22.45	0.5497				

Figura 5.46: resumo da segmentação da base de automóveis com o "*Cluster demographic*".

No recurso de visualização gráfica representado em parte pela Figura 5.47, pode-se observar à contribuição dos primeiros atributos na formação dos grupos mais numerosos, além do percentual, valor à esquerda, representativo de cada grupo, indicador à direita, obtido pela relação entre o número de objetos do grupo em relação à totalidade de objetos da base e os histogramas relativos a contribuição dos atributos.

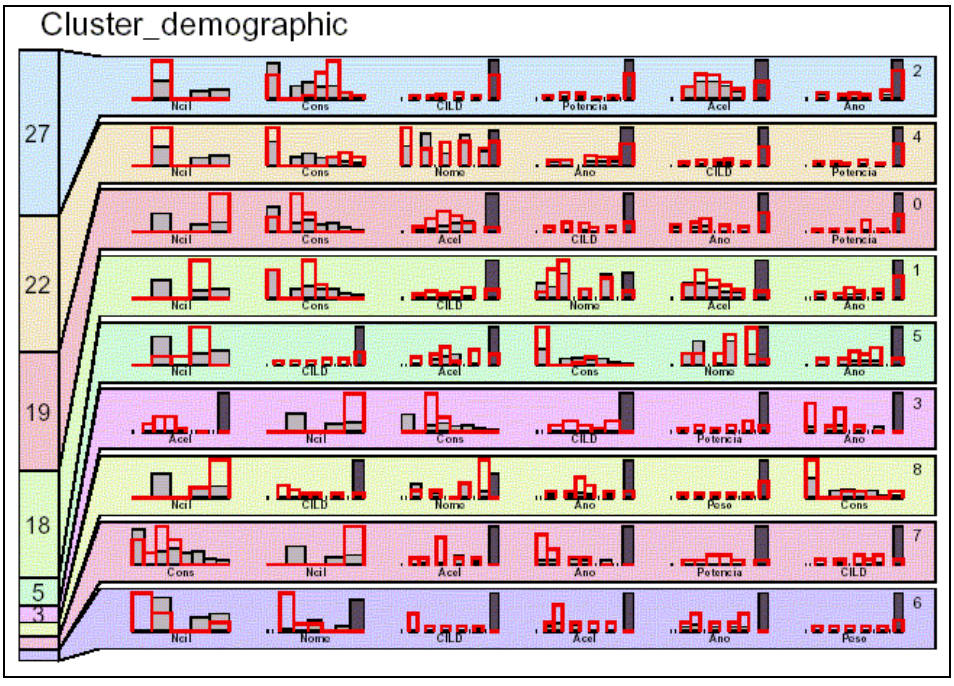


Figura 5.47: gráfico da segmentação com o "*Cluster demographic*".

O relatório da segmentação, que pode ser visto em parte na Figura 5.48, produzido pelo método "*Cluster neural*" para a base de automóveis completa, também apresenta o número ótimo de 9 grupos, coincidente com o resultado do método "C*luster demographic*" e, provavelmente, pelos mesmos motivos. Apesar de ser mostrado, "*Maximum Number of Clusters*" não é um parâmetro de entrada que pode ser informado diretamente pelo usuário. Assim, o aplicativo não se limita a este valor, podendo apontar valores além daquele como ótimo.

Nesta segmentação realizada, a composição dos grupos apresenta quantidades mais equilibradas embora não uniformes, o que não chega a surpreender, pois os automóveis que compõem a base, mesmo dentro de um mesmo grupo, não agregam características muito regulares.

Cluster_neural

Result created: 10/04/04 17:55:55

Result File	:	C:\TEMP\JAEL7LMO.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	5
Maximum Number of Clusters	:	9
Mining Run Outputs		
Number of Passes Performed	:	5
Number of Clusters	:	9
Deviation	:	0.07

Cluster Characteristics :

Id	Cluster Size		Id	Cluster Size	
	Absolute	Relative(%)		Absolute	Relative(%)
0	55	14.03	5	31	7.91
1	50	12.76	6	74	18.88
2	61	15.56	7	31	7.91
3	27	6.89	8	38	9.69
4	25	6.38			

Figura 5.48: resumo da segmentação da base de automóveis com o "*Cluster neural*".

O resumos da segmentação em forma de gráfico, apresentado na Figura 5.49, permite perceber a relevante contribuição dos atributos referentes à potência, consumo e número de cilindros, que podem ter sido determinantes para tal formação de grupos

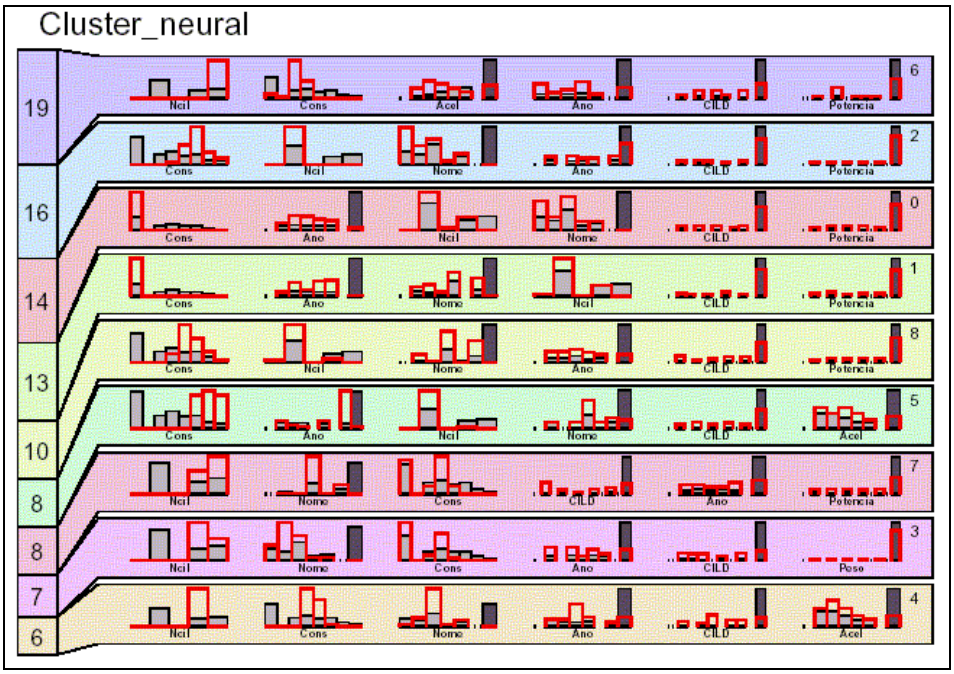


Figura 5.49: gráfico da segmentação com o "*Cluster neural*".

Como nos testes com os demais aplicativos, as colunas que não dizem respeito diretamente ao motor dos automóveis foram retiradas e a base, assim reduzida, foi novamente segmentada através do emprego dos dois métodos disponíveis.

Cluster_demographic

Result created: 11/05/04 17:49:50

Result File	:	C:\TEMP\S1O26J2R.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	2
Maximum Number of Clusters	:	20
Improvement Over Last Pass	:	2.00
Similarity Threshold	:	0.50
Mining Run Outputs		
Number of Passes Performed	:	2
Number of Clusters	:	10
Improvement Over Last Pass	:	0.00
Global Condorcet Value	:	0.6454
Parallel Parameters: Applicable For Results Created On Parallel Systems Only		
Records Processed Before Merging	:	10000
NStripe	:	1
NSpan	:	1440

Cluster Characteristics :

Id	Cluster Size		Condorcet	Id	Cluster Size		Condorcet
	Absolute	Relative(%)	Value		Absolute	Relative(%)	Value
0	83	21.17	0.6698	5	10	2.55	0.7365
1	11	2.81	0.8019	6	4	1.02	0.7997
2	72	18.37	0.6879	7	2	0.51	0.7403
3	7	1.79	0.6245	8	4	1.02	0.7490
4	195	49.74	0.6343	9	4	1.02	0.8369

Figura 5.50: resumo da segmentação com o "*Cluster demographic*" para a base de automóveis reduzida.

Primeiramente, com o método "*Cluster demographic*", o número ótimo de grupos encontrado foi 10, conforme mostrado na Figura 5.50, aumentando sensivelmente o valor do Critério de *Condorcet*. Apesar de não ser o número esperado de grupos, comparando-se este resultado com o obtido pelo *SearchCluster* para a mesma base, nota-se alguns interessantes pontos de convergência: um grupo com cerca de 50% de todos os objetos e outros dois grupos que dividem todos ou a grande maioria dos objetos remanescentes. São encontrados também grupos com 2 e 4 objetos, números iguais aos automóveis de origem Japonesa e Européia encontrados em meio aos automóveis originários dos EUA compondo um grupo único, formado pelo *SearchCluster* .

Com a redução da base de 9 para 5 atributos, é possível visualizar o gráfico de histogramas completo, como apresentado na Figura 5.51.

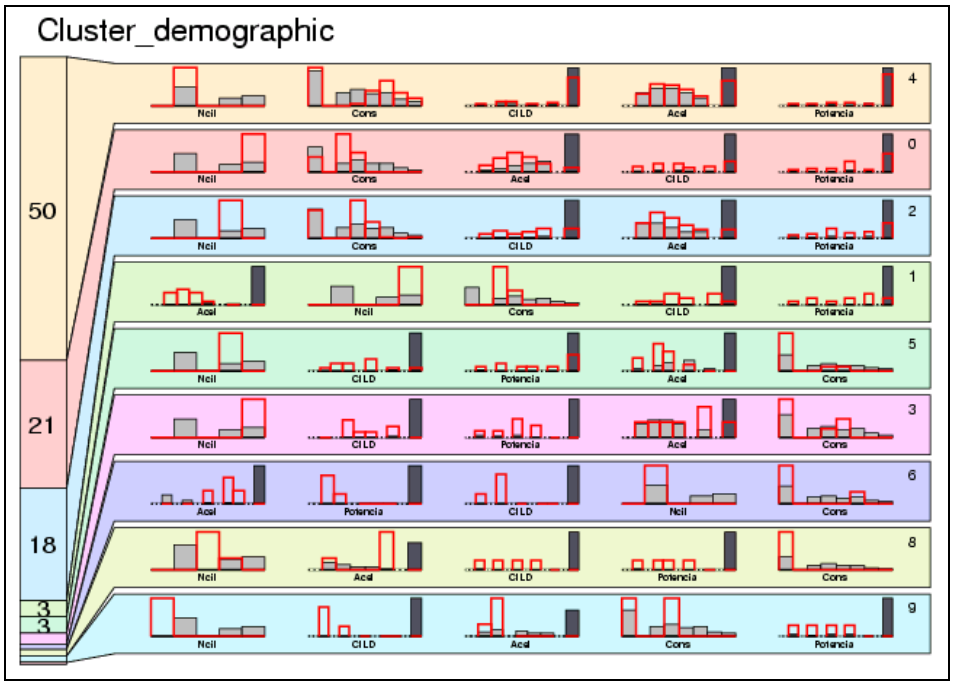


Figura 5.51: gráfico da segmentação com o "*Cluster demographic*" para a base de automóveis reduzida.

O segundo teste com a base de automóveis reduzida foi realizado com o emprego do método "*Cluster neural*". O relatório, mostrado em parte na Figura 5.52, e o gráfico, mostrado na Figura 5.53, apresentam como resultado um número de 9 grupos. Porém, há um defeito nestas saídas, pois o grupo 4 não apresenta nenhum objeto, devendo, assim, ser considerado como o resultado real uma segmentação em 8 grupos.

De qualquer forma, esta segmentação não se apresenta muito eficiente, pois entre os grupos formados não há um grupo que concentre um grande número de objetos, como naturalmente formariam os automóveis de origem norte-americana, nem grupos que, pelo seu número de objetos, levem a uma suposição razoável sobre a sua composição.

Cluster_neural

Result created: 11/05/04 17:53:11

Result File	:	C:\TEMP\MVWC45RX.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	5
Maximum Number of Clusters	:	9
Mining Run Outputs		
Number of Passes Performed	:	5
Number of Clusters	:	9
Deviation	:	0.02

Cluster Characteristics :

Id	Cluster Size Absolute	Relative(%)	Id	Cluster Size Absolute	Relative(%)
0	87	22.19	5	30	7.65
1	36	9.18	6	80	20.41
2	50	12.76	7	38	9.69
3	3	0.77	8	68	17.35
4	0	0.00			

Figura 5.52: resumo da segmentação com o "*Cluster neural*" para a base de automóveis reduzida.

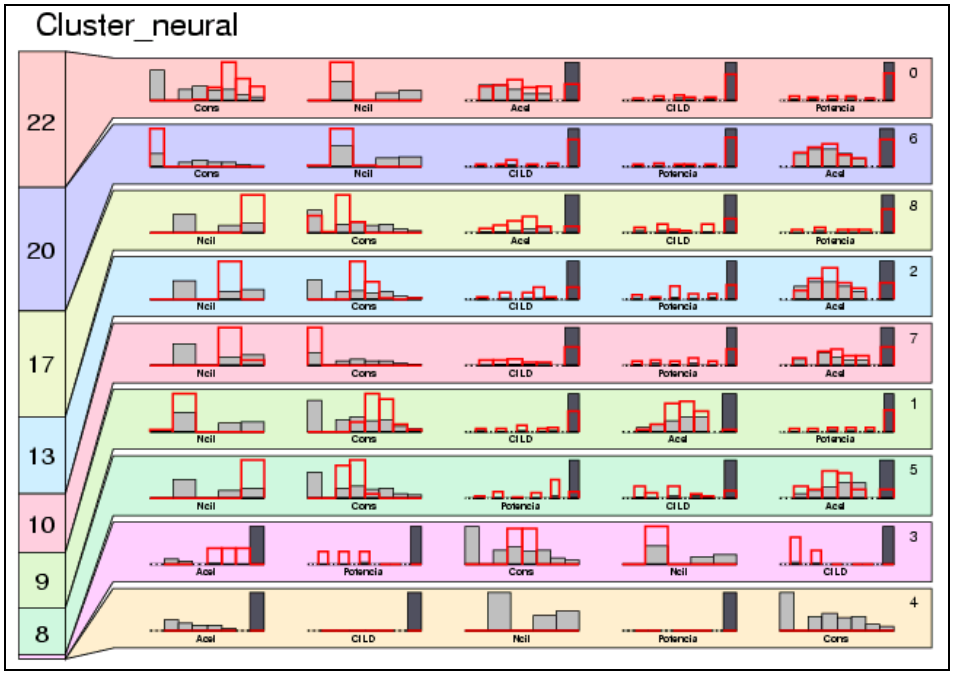


Figura 5.53: gráfico da segmentação com o "*Cluster neural*" para a base de automóveis reduzida.

5.3 Terceiro caso – base de dados de clientes “*churn*”

A base utilizada neste caso é de uma companhia de telecomunicações dos EUA, contendo 21 atributos relativos a 5000 clientes com padrão de consumo e relacionamento com a empresa considerado normal e clientes que cancelaram algum serviço da empresa ou reduziram expressivamente sua utilização, considerados, assim, clientes “*churn*”. As informações contidas nesta base, e conseqüentemente a sua análise, são de extrema valia para a empresa, pois através delas pode-se detectar padrões de comportamento que identifiquem os clientes propensos a trocarem de prestadora de serviços, possibilitando que a empresa possa tomar alguma medida que evite a concretização do fato. Como um dos atributos da base é justamente um identificador de clientes “*churn*”, o estudo pode ser supervisionado, sabendo-se que a base divide-se em dois grupos com 707 clientes “*churn*” e 4293 clientes não “*churn*”, respectivamente.

Os 21 atributos que compõem a base de clientes “*churn*”, na mesma seqüência em que figuram na base, são: sigla do estado onde o cliente possui a conta, número de meses que o cliente utiliza os serviços da concessionária, código de área especificado, número de telefone, indicador de existência de plano com pacote de preços especiais para chamadas internacionais, indicador de existência de plano com pacote de preços especiais para clientes com correio de voz, média do número de mensagens de correio de voz por dia, média do número de minutos de utilização de serviços durante o horário comercial, média do número de chamadas efetuadas durante o horário comercial, custo médio de utilização de serviços durante o horário comercial, média do número de minutos de utilização de serviços entre o horário comercial e o horário noturno, média do número de chamadas efetuadas entre o horário comercial e o horário noturno, custo médio de utilização de serviços entre o horário comercial e o horário noturno, média do número de minutos de utilização de serviços durante o horário noturno, média do número de chamadas efetuadas durante o horário noturno, custo médio de utilização de serviços durante o horário noturno, média do número de minutos de utilização com taxa internacional, média do número de chamadas internacionais, custo médio de utilização com taxa internacional, número de chamadas feitas pelo cliente, nos últimos seis meses, para o serviço de suporte ao cliente da concessionária, indicador de troca de concessionária de serviços nos últimos seis meses (“*churned*”).

Para que sejam efetuados estudos sobre os clientes desta base, primeiramente é necessário substituir os atributos meramente indicadores, tipo texto – yes/no (sim/não), por valores numéricos inteiros – 1/0, respectivamente e a sigla dos estados por números inteiros arbitrários, iniciados em 1 e sendo atribuídos seqüencialmente às siglas, à medida que estas vão ocorrendo.

A conversão dos atributos da base de clientes "*churn*" do tipo texto para numérico inteiro foi realizada de acordo com a tabela 5.2.

TABELA 5.2 – Conversão de atributos texto da base de clientes "churn"

Atributo							Valor original	Valor convertido
indicador de existência de plano chamadas internacionais							YES	1
							NO	0
indicador de existência de plano correio de voz							YES	1
							NO	0
indicador de troca de concessionária							YES	1
							NO	0
sigla do estado	AR	1	OR	18	UT	35		
	WI	2	CT	19	NY	36		
	ME	3	CO	20	LA	37		
	NC	4	AL	21	GA	38		
	MN	5	KY	22	MO	39		
	OH	6	MI	23	IA	40		
	DE	7	DC	24	HI	41		
	IL	8	WV	25	RI	42		
	NV	9	AK	26	SC	43		
	IN	10	MS	27	TX	44		
	MT	11	WY	28	FL	45		
	TN	12	MA	29	WA	46		
	VT	13	OK	30	MD	47		
	VA	14	NJ	31	PA	48		
	ID	15	ND	32	NH	49		
	KS	16	NM	33	AZ	50		
	SD	17	NE	34	CA	51		

5.3.1 Resultados do *SearchCluster* sobre a base de clientes "*churn*"

Inicialmente, o software *SearchCluster* foi empregado para uma visualização dos objetos da base de clientes "*churn*" em um plano bidimensional, através do gráfico de coordenadas estrela, visto na Figura 5.54.

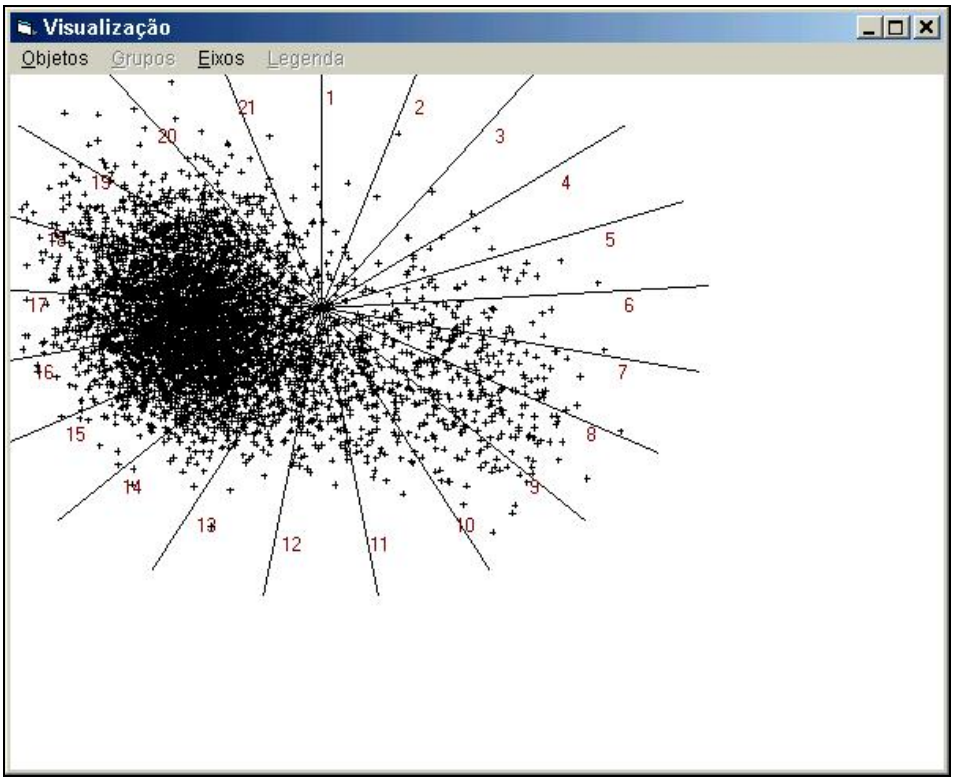


Figura 5.54: visualização da base de clientes “*churn*” em um plano bidimensional.

A visualização da base como se apresenta, embora possa dar a idéia de que dois grupos podem ser formados, não permite nenhuma conclusão definitiva sobre o comportamento de seus objetos, inclusive indicando que, da maneira como está, a análise será muito dificultada, devido à proximidade dos objetos. Como a natureza da base é mesmo uma divisão em dois grupos, o atributo “*churned*” teve seu peso bastante aumentado para criar maior discrepância entre os grupos. A segmentação realizada desta forma, na faixa de teste de 2 a 8 grupos, redundou em 100% de acerto na formação dos grupos, como mostram a Figura 5.55 e a Figura 5.56, e também na alocação dos clientes, conforme constatado na comparação dos dados do arquivo de saída, permitindo que os componentes de cada grupo sejam identificados por cores diferentes, conforme demonstrado na Figura 5.57.

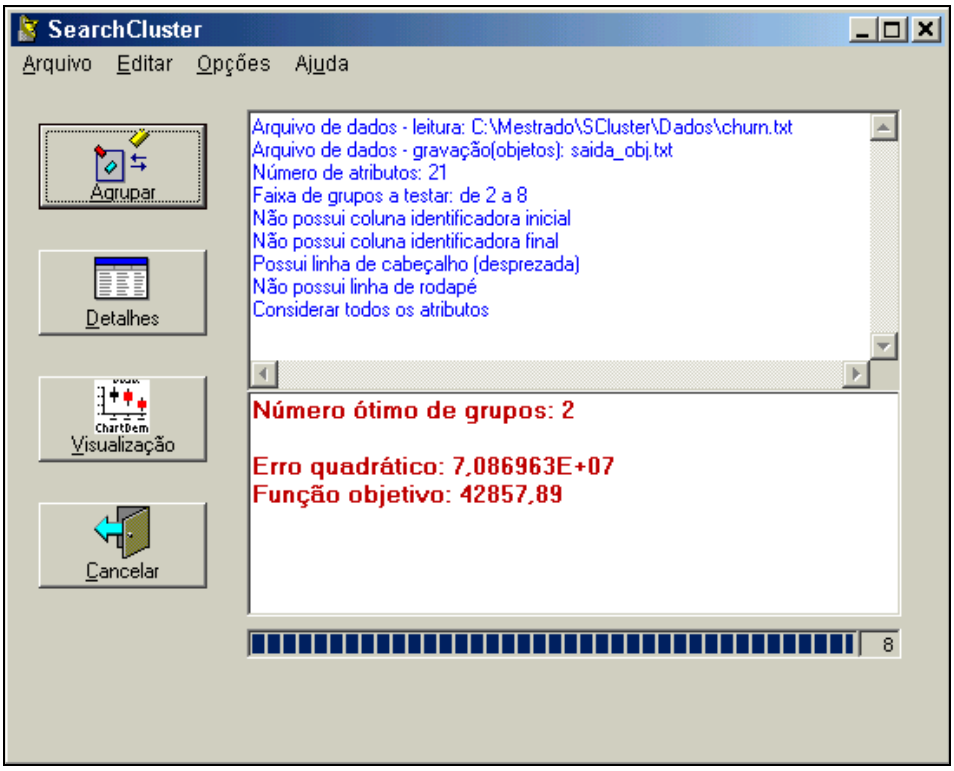


Figura 5.55: resultados da segmentação com aumento de peso do atributo "*churned*".

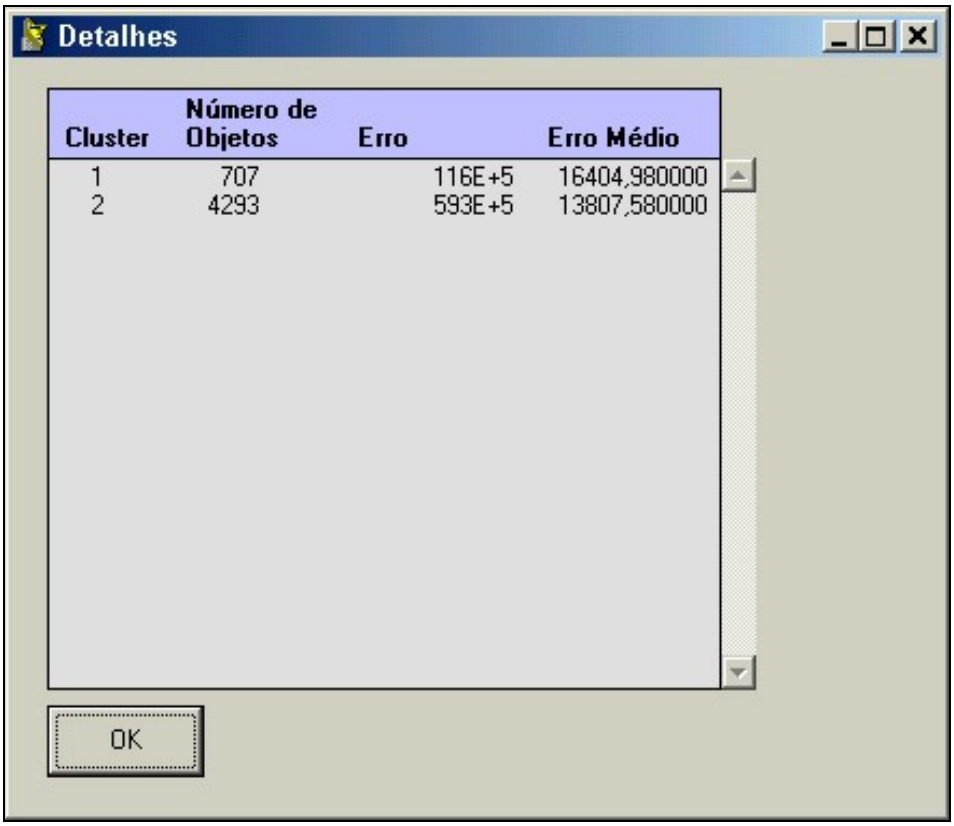


Figura 5.56: detalhes da segmentação com aumento de peso do atributo "*churned*"

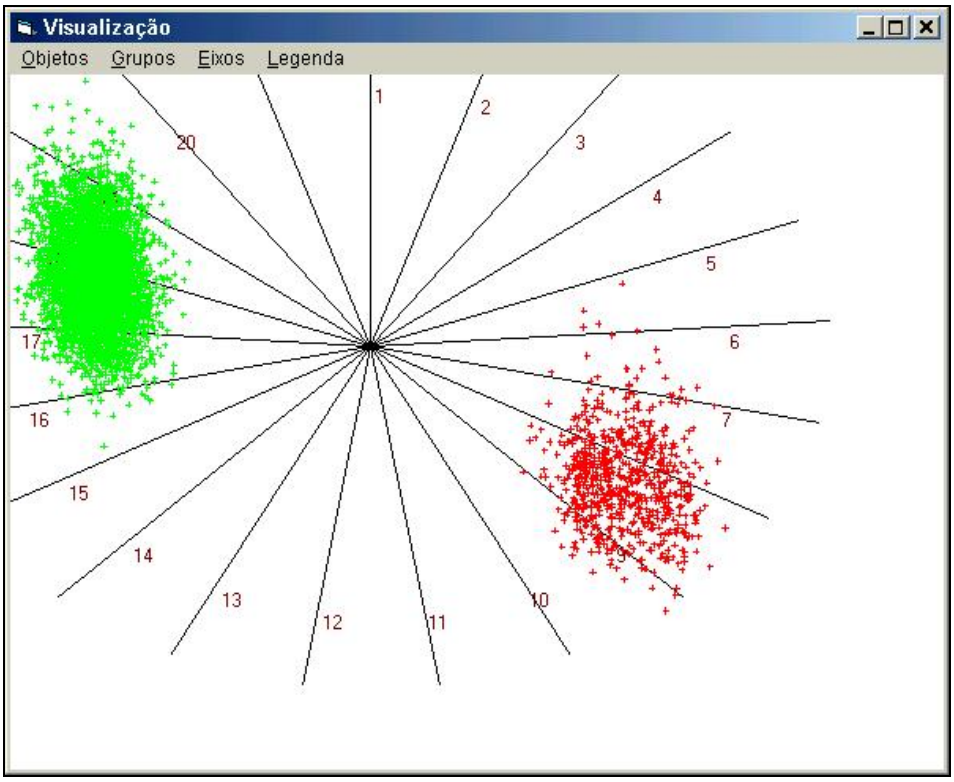


Figura 5.57: visualização dos objetos com aumento de peso do atributo "*churned*".

Uma sondagem mais detida sobre o conteúdo da base de dados, numa tentativa de tornar a sua exploração mais eficaz e conclusiva, indica a eliminação da coluna número de telefone, por ser um atributo que não possui nenhuma relação com o perfil do cliente e sua representação como um número de 7 dígitos tornar-se sem sentido. A última coluna, indicador de troca de serviço, "*churned*", passará a ser considerada como uma coluna identificadora e não será mais percebida pelo software como atributo dos clientes, já que pode ser vista apenas como uma coluna de supervisão do resultado obtido pela segmentação. Os atributos referentes a estado e código de área também serão desconsiderados nos próximos passos.

Mesmo com a desconsideração das colunas descritas acima, a visualização da base, como mostra a Figura 5.58, apresenta todos os objetos muito próximos do centro da base, formando um único grupo Este resultado, provavelmente, ainda se deve a colunas fortemente relacionadas que diminuem a divergência entre os grupos.

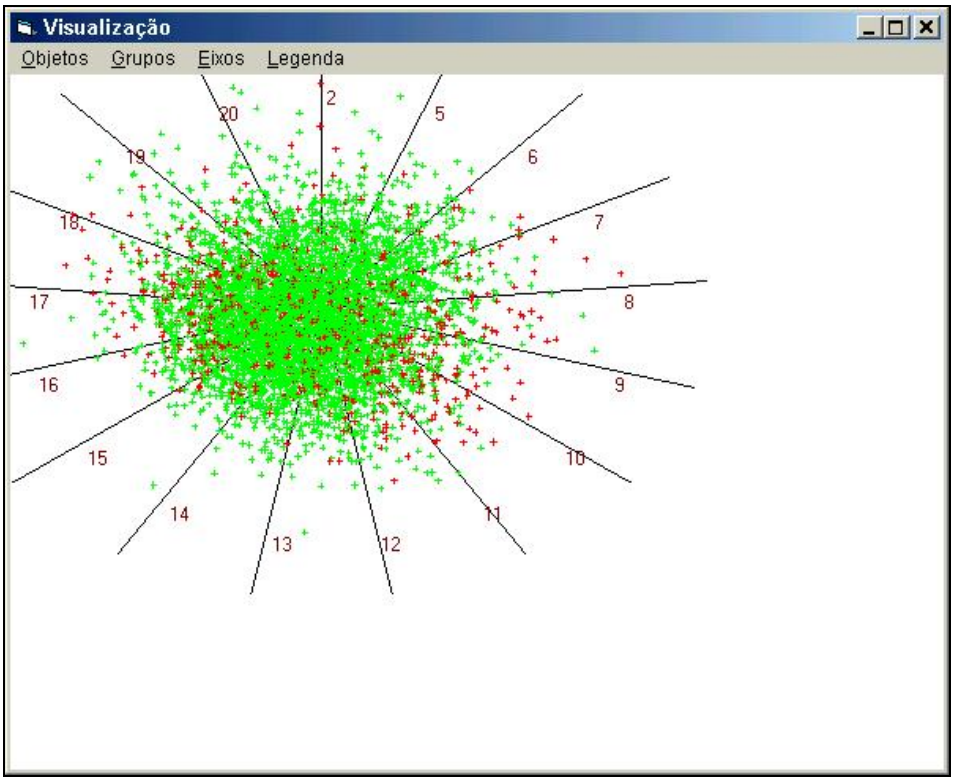


Figura 5.58: visualização dos objetos sem os atributos 1, 3, 4 e 21.

Por ainda manter resultados pouco conclusivos, a base foi revista com maior grau de acuidade. Assim, grupos de colunas que podem ter alta correlação entre si passaram a contar com apenas uma delas, sem prejuízo da representação dos dados e simplificando o processo de análise. Os grupos referentes às chamadas em horário comercial, chamadas noturnas, chamadas em horário intermediário e chamadas com taxa internacional foram simplificados de três para apenas uma coluna cada grupo, passando a ser considerado apenas o atributo referente à média do número de minutos de utilização, o qual está forte e diretamente relacionado aos atributos custo total das chamadas e média do número de chamadas efetuadas em cada classe de tarifa. A visualização, vista na Figura 5.59, apresenta um tímido princípio de separação dos objetos em dois grupos.

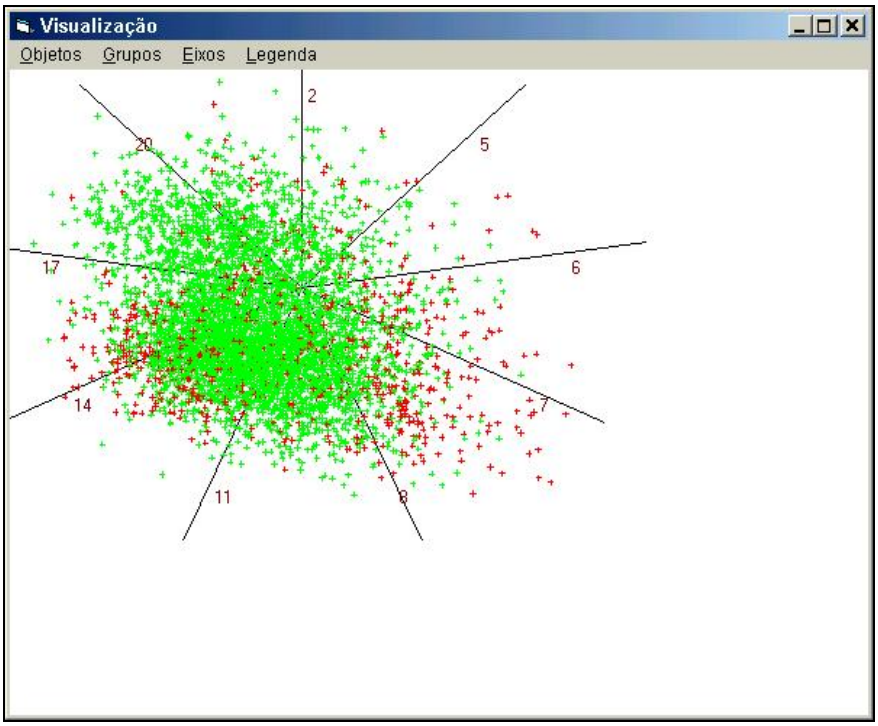


Figura 5.59: visualização dos objetos com redução máxima de colunas.

Traçando-se dois gráficos, um com a evolução do Índice PBM em relação ao número de grupos, como visto na Figura 5.60, e outro com a evolução do Critério de *Calinski* e *Harabasz*, como visto na Figura 5.61, pode-se notar que apesar do primeiro ter uma redução de valores mais precipitada nas segmentações iniciais em relação ao segundo, ambos apontam o mesmo número de grupos, 2, como ótimo e possuem, a partir de 4 grupos, perfis com diversas coincidências

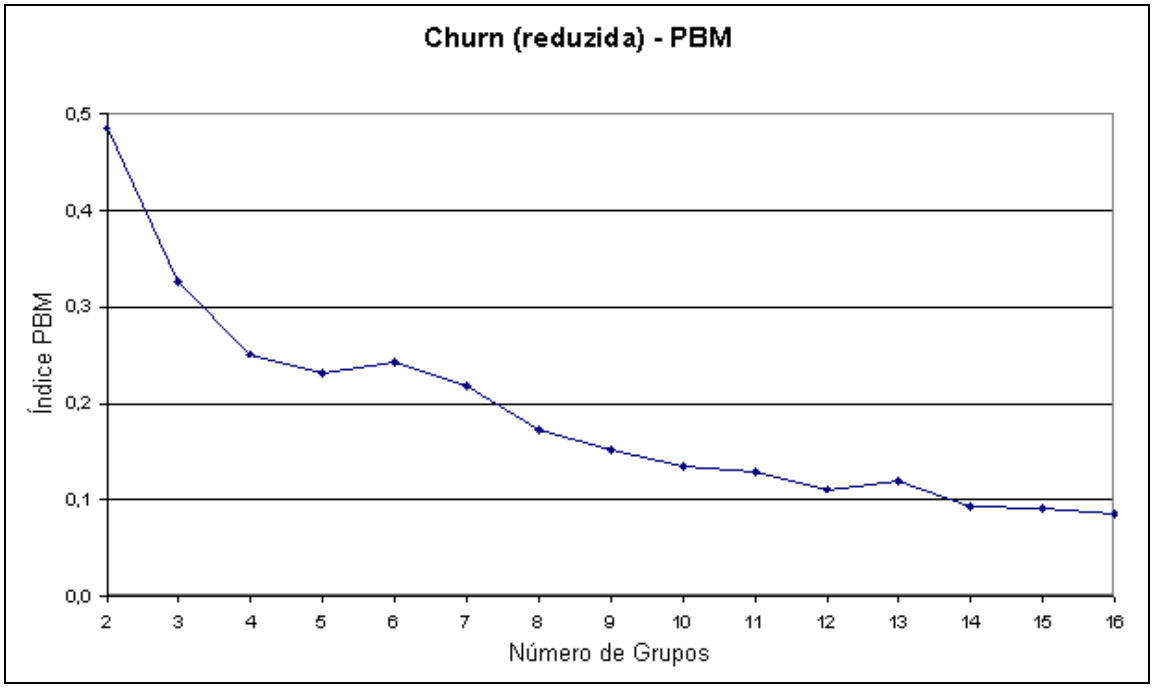


Figura 5.60: evolução do Índice PBM para a base de clientes "*churn*" reduzida.

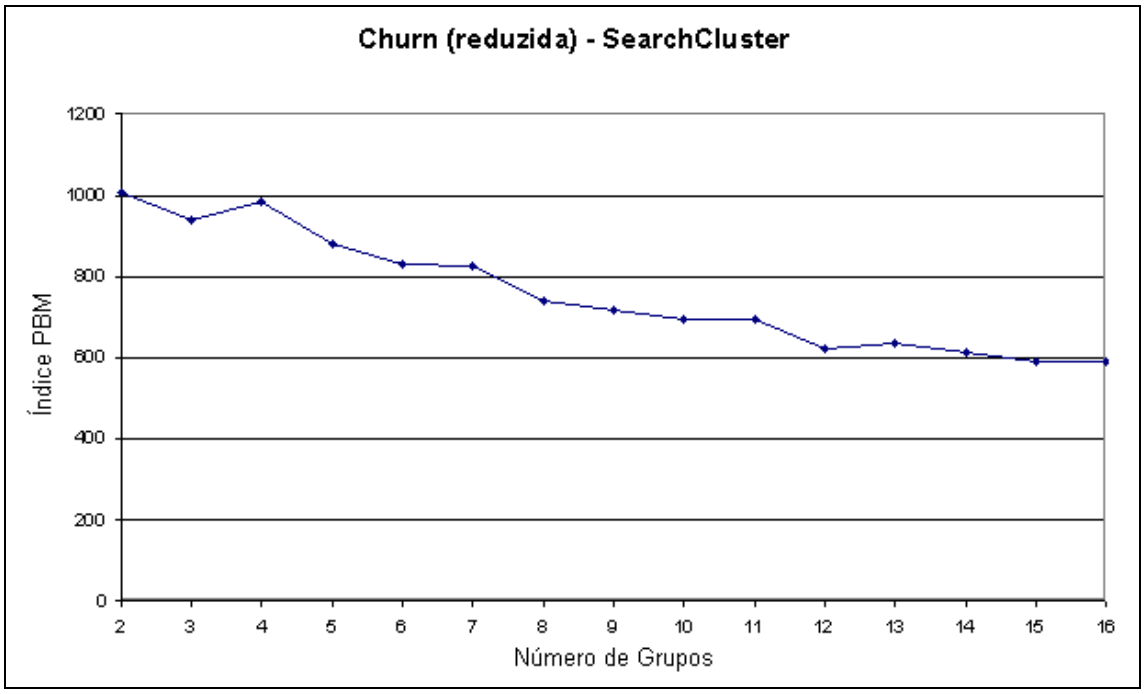


Figura 5.61: evolução do Critério *Calinski* e *Harabasz* para a base de clientes "*churn*" reduzida.

A fim de testar a influência dos atributos remanescentes na decisão dos clientes trocarem de prestadora de serviços, o peso dos atributos existência de plano internacional, existência de correio de voz e número de chamadas ao serviço de suporte foram aumentados um a um, obtendo-se as visualizações da Figura 5.62.

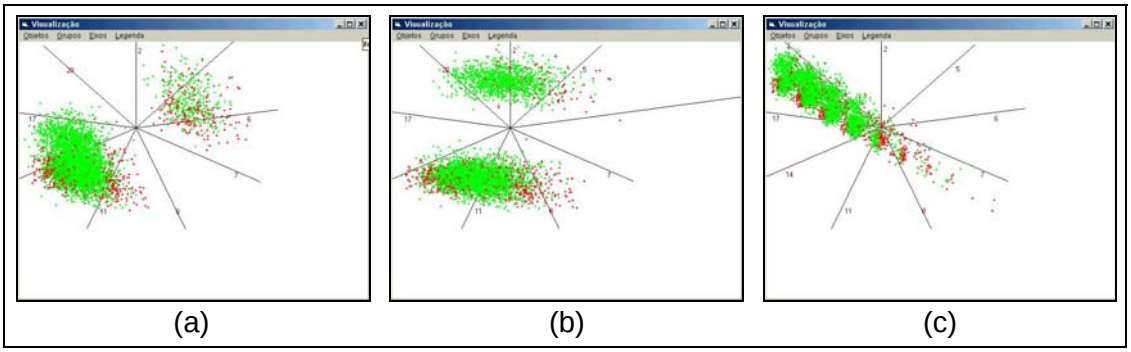


Figura 5.62: visualização dos objetos com aumento de peso de algumas colunas. (a) aumento de peso do atributo plano internacional de colunas; (b) aumento de peso do atributo correio de voz; (c) aumento de peso do atributo chamadas ao suporte.

Observando-se a Figura 5.62.a e a Figura 5.62.b, vê-se a nítida formação de dois grupos. Embora as quantidades de objetos nos grupos e as formas destes pareçam muito semelhantes, a presença de clientes "*churn*" é proporcionalmente maior nos grupos em possuem plano internacional e correio de voz, o que indica serem estes atributos de grande importância para a manutenção de clientes, devendo merecer maior atenção por parte da companhia, pois os clientes que possuem um ou

ambos os planos são mais suscetíveis de trocar de prestadora de serviços. O atributo número de chamadas ao suporte não é decisivo para o cliente tornar-se "*churn*", pois, como pode ser visto na Figura 5.62.c, não há uma grande separação em dois grandes grupos, parecendo que há uma leve tendência à formação de vários grupos menores, mas que mantêm a proporção de clientes "*churn*" muito aproximada.

Outra conclusão interessante que pode ser tirada destes testes, é que a afirmação do método *k-means* não ser muito eficiente para segmentação de grupos de tamanhos muito diferentes faz sentido: para a base de clientes "*churn*" com 5000 objetos, onde 4293 pertencem ao grupo de clientes que permaneceram fiéis à concessionária de serviços e apenas 707 trocaram de empresa, devendo aparecer em um segunda grupo, os resultados da segmentação são geralmente de baixa qualidade.

5.3.2 Resultados do *gCLUTO* sobre a base de clientes "*churn*"

No primeiro teste com o *gCLUTO* foi analisada a base completa, com os 21 atributos disponíveis. O resultado da segmentação em dois grupos pelo *gCLUTO*, representado em detalhes na Figura 5.63, chama a atenção pela distribuição de objetos em números muito parecidos. A visualização de montanha reforça a idéia da grande similaridade entre os objetos e os grupos formados, conforme visto na Figura 5.20.

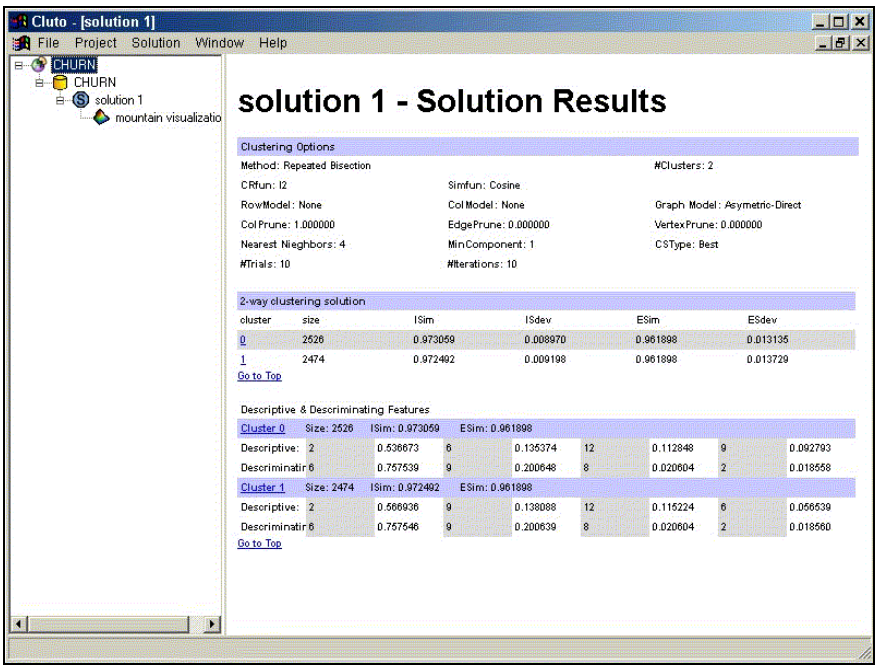


Figura 5.63: resultados da segmentação da base de clientes "*churn*" pelo *gCLUTO*.

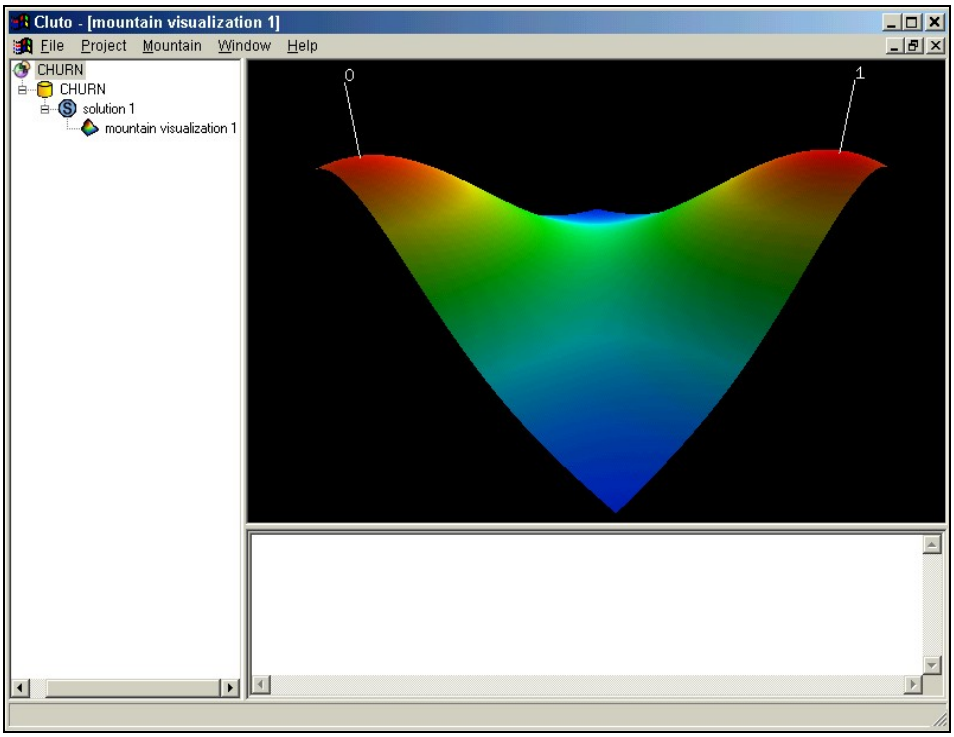


Figura 5.64: visualização de montanha da segmentação da base em dois grupos.

O segundo teste consistiu no emprego do *gCLUTO* sobre uma base reduzida, onde permaneceram apenas as colunas referentes a número de meses de utilização dos serviços, indicador de plano para chamadas internacionais, indicador de plano para correio de voz, média de mensagens de correio de voz por dia, média de minutos de utilização de serviços durante o horário comercial, média de minutos de utilização de serviços entre o horário comercial e o horário noturno, média de minutos de utilização de serviços durante o horário noturno, média de minutos de utilização com taxa internacional, número de chamadas para o serviço de suporte.

Mesmo com o corte de 12 colunas a composição dos grupos permanece muito semelhante, como apresentado na Figura 5.65 e na Figura 5.66, corroborando os testes realizados anteriormente pelo *SearchCluster*. Por limitações do *gCLUTO*, a análise da base de clientes "*churn*" não pode ser mais aprofundada, pois o *software* não faz a descoberta do melhor número de grupos para uma determinada base, não permite a identificação dos objetos que compõem o grupo ou a mudança do peso de algum atributo, como realizado no *SearchCluster*, por exemplo.

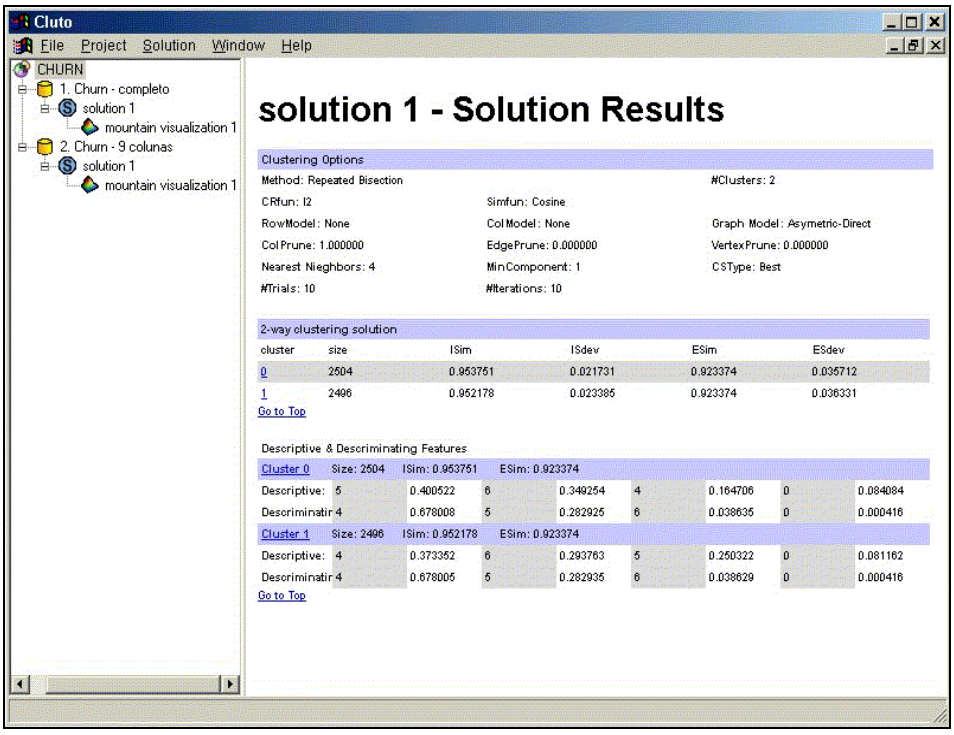


Figura 5.65: resultados da segmentação da base reduzida a 9 colunas.

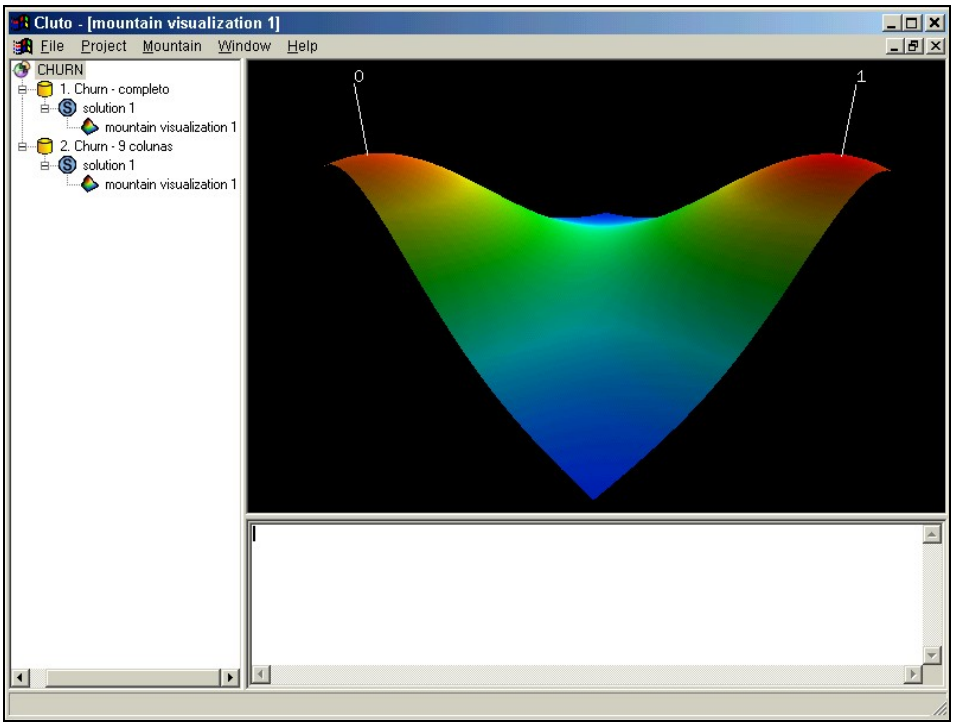


Figura 5.66: visualização de montanha da segmentação da base em dois grupos.

5.3.3 Resultados do *Enterprise Miner* sobre a base de clientes "*churn*"

O *Enterprise Miner*, empregado na segmentação da base de clientes "*churn*" com todos os seus atributos, realizou a descoberta de 5 grupos distintos dentro do

conjunto de dados, como mostra a Figura 5.67, embora existam 4 grupos muitíssimo próximos, considerados dois a dois, como pode ser visto na Figura 5.68.

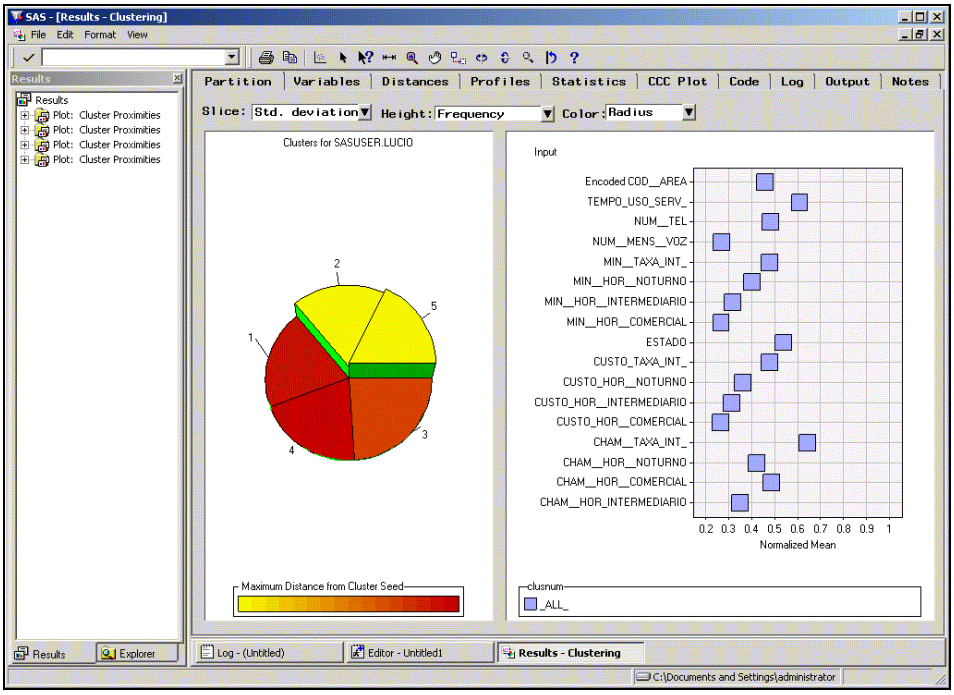


Figura 5.67: 1ª segmentação da base de clientes "*churn*" pelo *Enterprise Miner*.

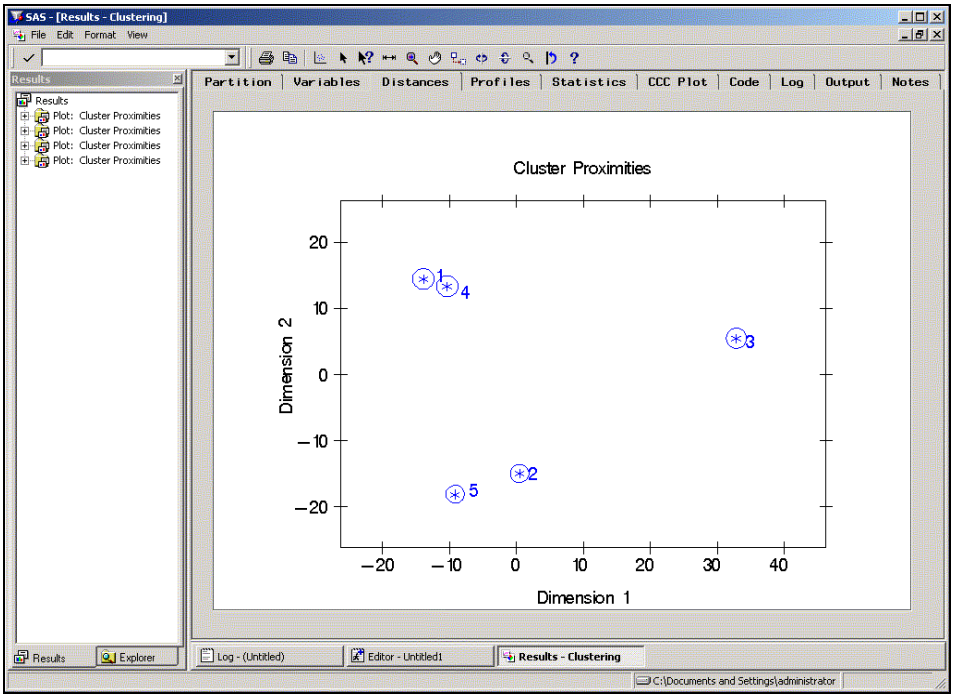


Figura 5.68: gráfico de proximidade dos grupos da base de clientes "*churn*".

O gráfico do CCC, mostrado na Figura 5.69, apresenta uma evolução relativamente uniforme ao longo dos 40 grupos testados e deixa perceber que as distribuições em 2, 3 e 4 grupos não são muito discrepantes da distribuição em 5 grupos.

Tal forma de divisão pode ser atribuída à influência de atributos categóricos que foram transformados em valores numéricos contínuos através de uma equivalência arbitrária; atributos que não expressem uma característica real do objeto, por exemplo, número de telefone é um valor dado pela empresa, independentemente do perfil do cliente; ou atributos que tenham uma relação direta e proporcional entre si, como número de chamadas, valor da chamada e média dos valores das chamadas.

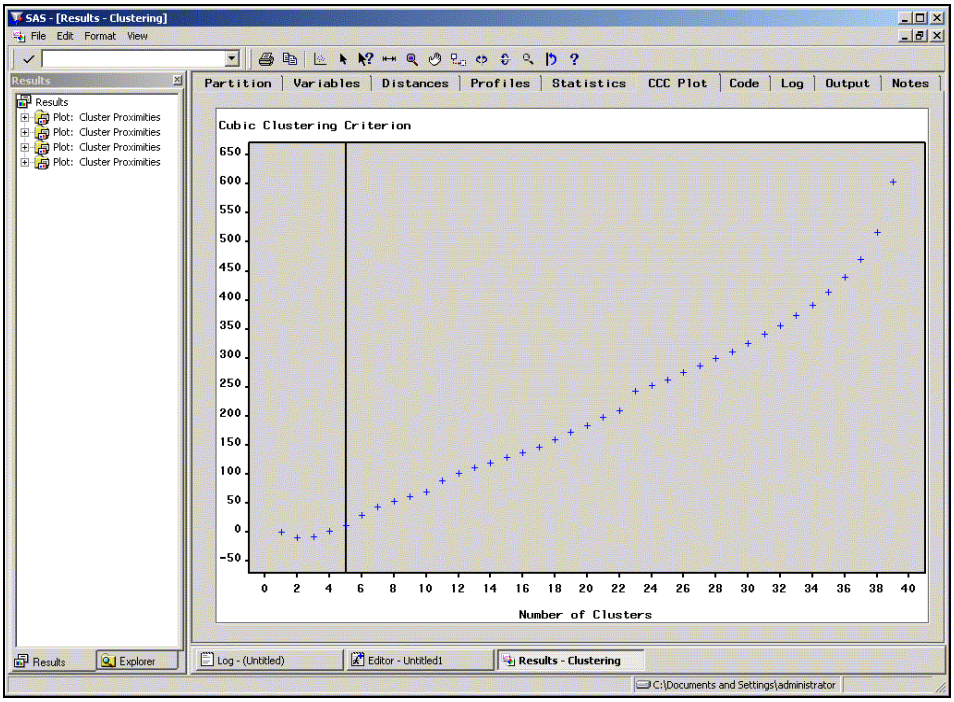


Figura 5.69: gráfico CCC da 1ª segmentação da base de clientes "*churn*".

Assim, a base foi reduzida, à semelhança do que foi realizado nos experimentos anteriores, de 21 para 9 colunas e novamente submetida ao processo de segmentação do *Enterprise Miner*, quando a base foi dividida em 6 grupos, cuja explicação pode ser o fato de existirem três faixas de horários para cobrança das chamadas, comercial, noturno e intermediário, que podem dividir-se, cada uma, em dois grupos, de acordo com os tipos de clientes, o que trocou de serviço e o que não trocou. Este resultado pode ser visto na Figura 5.71.

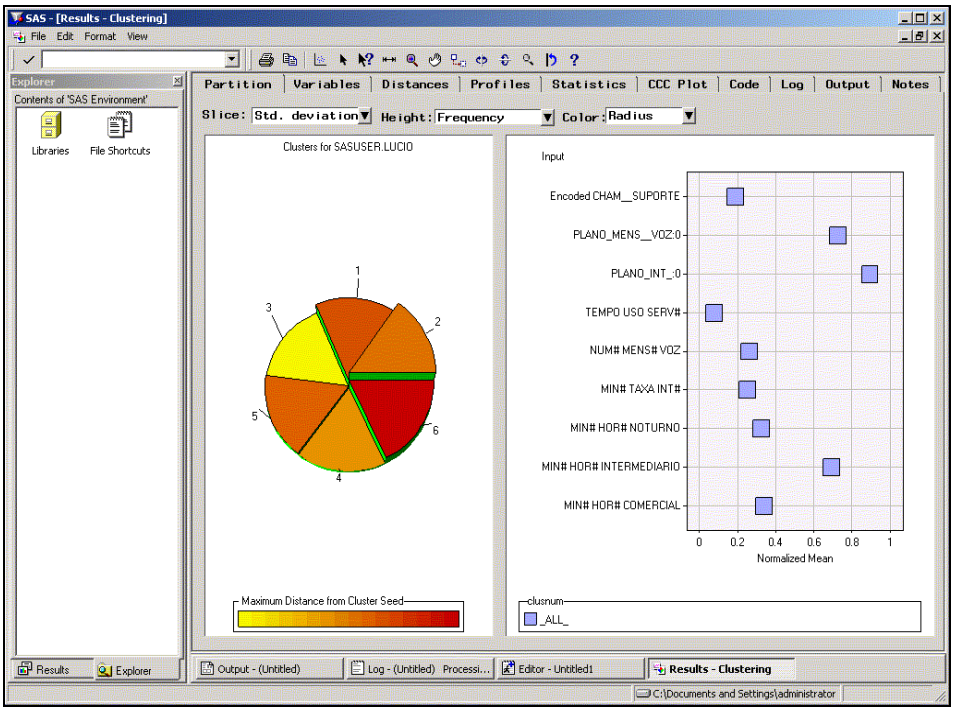


Figura 5.70: 2ª segmentação da base de clientes "*churn*" pelo *Enterprise Miner*.

Os grupos apresentam-se, agora, com maior divergência, como apresentado na Figura 5.72, provavelmente devido ao peso atribuído pelo software aos atributos que dizem respeito às faixas de horários das chamadas.

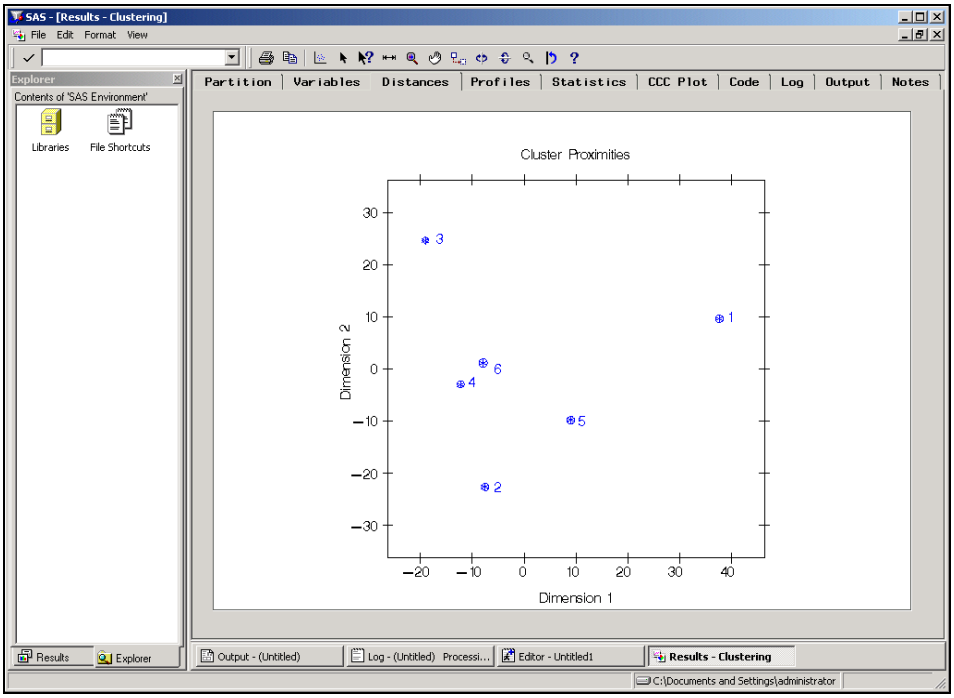


Figura 5.71: gráfico de proximidade dos grupos na 2ª segmentação da base.

O CCC mostra uma tendência à uniformidade menor agora, mas ainda com outras formações podendo ser satisfatórias. O gráfico de evolução do CCC é apresentado na Figura 5.72.

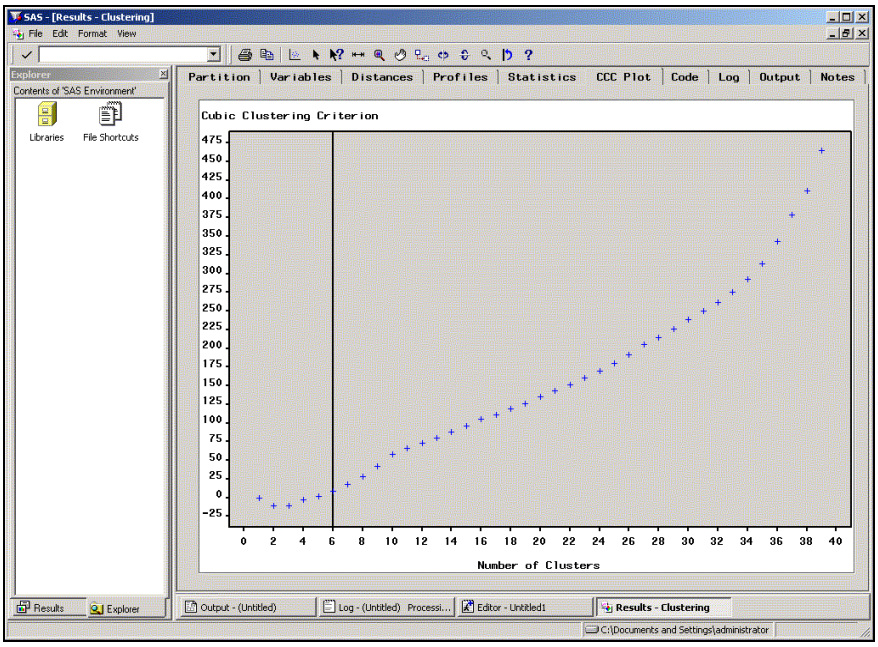


Figura 5.72: gráfico CCC da 2ª segmentação da base de clientes "*churn*".

Ainda mais um teste foi realizado, onde as segmentações a serem testadas foram limitadas a apenas dois grupos. Mesmo assim, o desempenho do processo não foi considerado de muito boa qualidade, pois os grupos não foram separados convenientemente, pois a divisão natural é um grupo com 707 clientes que trocaram de concessionária e outro grupo com os 4293 clientes que permaneceram na empresa. O melhor resultado apontado pelo *Enterprise Miner*, visto na Figura 5.73 e na Figura 5.74, formou um grupo com 1323 e outro com 3677 clientes.

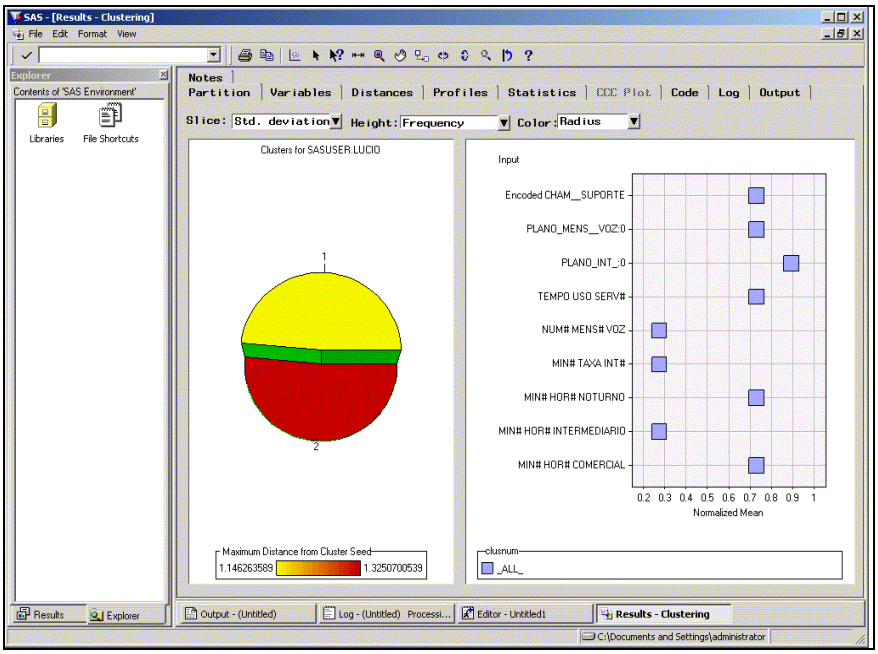


Figura 5.73: segmentação da base de clientes "*churn*" em dois grupos.

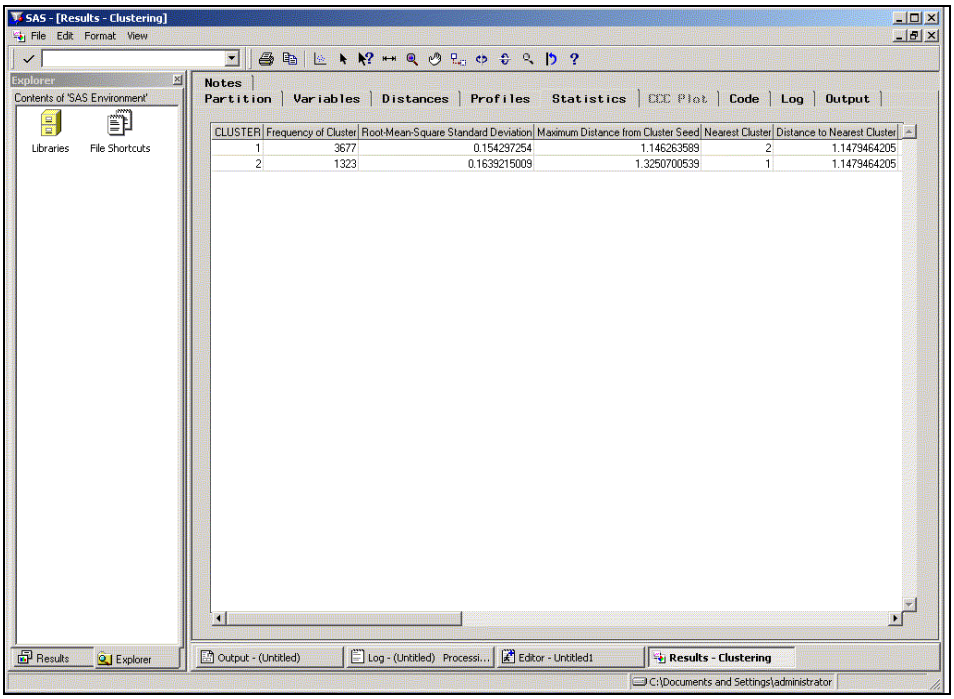


Figura 5.74: número de objetos e outros detalhes dos dois grupos formados .

5.3.4 Resultados do *Mineset* sobre a base de clientes "*churn*"

Para análise da base de clientes "*churn*" com o *Mineset*, optou-se por iniciar logo pela base reduzida a 9 colunas, ou seja, apenas com os atributos que significam ações ou características determinantes do perfil dos clientes.

Neste teste a segmentação redundou, mais uma vez, na formação de dois grupos, como mostra a Figura 5.75

```
Fitness = 0.6913                          | Result of clustering:
1 clusters: fitness = 0.6912986           | ----------------------------------------
2 clusters: fitness = 0.4232792           |
3 clusters: fitness = 0.4041667           | Overall root-mean-squared distance from
4 clusters: fitness = 0.3521153           | record to centroid: 0.4706 +- 0.2057
5 clusters: fitness = 0.3514269           | RMS distance from records to each centroid:
6 clusters: fitness = 0.3456019           |   Cluster 1-A: 0.4628 +- 0.2059
7 clusters: fitness = 0.345098            |   Cluster 1-B: 0.4916 +- 0.2033
8 clusters: fitness = 0.3448967           | done.
9 clusters: fitness = 0.3414743           |
10 clusters: fitness = 0.339739           |
Chose 2 clusters (choice point = 0.5155188)|
```

Figura 5.75: relatório texto do processo de segmentação da base de clientes "*churn*".

Embora as quantidades de objetos por grupo de 3677 e 1323, contra 4293 e 707, não sejam as esperadas, o resultado é bastante semelhante ao apresentado pelo *SearchCluster* e permite algumas conclusões como, por exemplo, a grande influência do atributo número de mensagens de voz (correio) na separação dos grupos, haja vista um grupo, identificado como 1-A, somente possuir clientes que não adotam o

plano de voz e o outro grupo, identificado como 1-B, ao contrário, ser exclusivamente constituído por clientes que utilizam tal plano, conforme é apresentado na Figura 5.76.

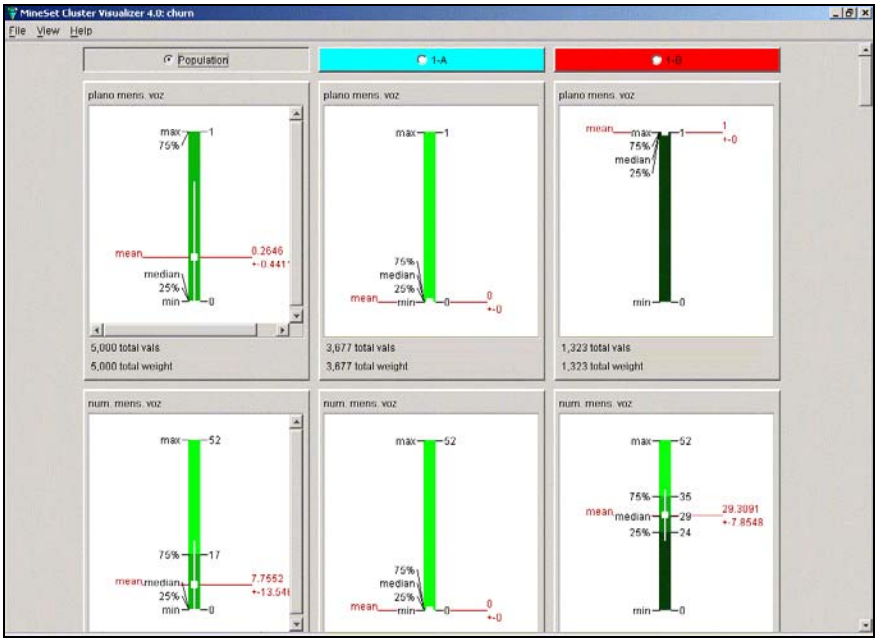


Figura 5.76: visualização estatística da segmentação da base de clientes "*churn*" reduzida.

5.3.5 Resultados do *Intelligent Miner* sobre a base de clientes "*churn*"

Cluster_demographic

Result created: 10/28/04 17:07:27

Result File	:	C:\TEMP\KO19YHLQ.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	2
Maximum Number of Clusters	:	20
Improvement Over Last Pass	:	2.00
Similarity Threshold	:	0.50
Mining Run Outputs		
Number of Passes Performed	:	2
Number of Clusters	:	20
Improvement Over Last Pass	:	0.00
Global Condorcet Value	:	0.4844
Parallel Parameters: Applicable For Results Created On Parallel Systems Only		
Records Processed Before Merging	:	10000
NStripe	:	1
NSpan	:	1440

Cluster Characteristics :

Id	Cluster Size Absolute	Cluster Size Relative(%)	Condorcet Value	Id	Cluster Size Absolute	Cluster Size Relative(%)	Condorcet Value
0	428	8.56	0.5132	10	222	4.44	0.4753
1	227	4.54	0.4765	11	158	3.16	0.3908
2	229	4.58	0.4799	12	223	4.46	0.4521
3	261	5.22	0.4777	13	398	7.96	0.5111
4	186	3.72	0.4441	14	220	4.40	0.4675
5	204	4.08	0.4619	15	205	4.10	0.4111
6	493	9.86	0.5221	16	293	5.86	0.4986
7	209	4.18	0.4389	17	287	5.74	0.4938
8	181	3.62	0.4262	18	187	3.74	0.4452
9	191	3.82	0.4302	19	198	3.96	0.4413

Figura 5.77: relatório de saída do método "*Cluster demographic*" para a base de clientes "*churn*" completa.

Inicialmente, o software *Intelligent Miner* foi empregado na segmentação da base de clientes "*churn*" completa, considerando todos os seus 21 atributos originais.

Sabendo-se que o esperado seria uma divisão da base em dois grupos, ou até em um número maior, definido pela subdivisão destes em grupos menores, percebe-se que a segmentação pelo método "*Cluster demographic*" não apresentou bons resultados. A Figura 5.77 mostra uma divisão em 20 grupos, com uma distribuição de objetos muito equitativa. E mesmo assim, os resultados que podem ser vistos não refletem o número máximo de grupos que o aplicativo pode apontar como ótimo. Testes subseqüentes apontam para um número superior a 200 grupos, o que, para esta base, é um resultado de qualidade muito inferior ao aceitável, não sendo útil para analisar o comportamento dos objetos.

A segmentação pelo método "*Cluster neural*" não apresentou resultados tão inesperados, embora a formação 13 grupos, com números de objetos não muito discrepantes, também esteja longe de ser considerada boa e a distribuição dos objetos pelos grupos não contribua para observações conclusivas. E, ainda, como pode ser vista na Figura 5.78, uma vez mais o aplicativo apresentou o defeito de apontar 15 grupos, quando dois deles estão vazios. Se o grupo não possui objetos não deveria ser contado no processo de agrupamento.

Cluster_neural

Result created: 10/28/04 17:13:39

Result File	:	C:\TEMP\MICM7DUA.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	10
Maximum Number of Clusters	:	9
Mining Run Outputs		
Number of Passes Performed	:	10
Number of Clusters	:	15
Deviation	:	0.15

Cluster Characteristics :

Id	Cluster Size Absolute	Relative(%)	Id	Cluster Size Absolute	Relative(%)
0	494	9.88	8	476	9.52
1	0	0.00	9	158	3.16
2	561	11.22	10	0	0.00
3	426	8.52	11	471	9.42
4	354	7.08	12	441	8.82
5	535	10.70	13	360	7.20
6	168	3.36	14	196	3.92
7	360	7.20			

Figura 5.78: relatório de saída do método "*Cluster neural*" para a base de clientes "*churn*" completa.

Considerando-se que a base de clientes "*churn*" completa é formada por atributos dispensáveis ao processo de segmentação, por sua pouca relevância, ou até

mesmo pelas distorções que poderiam causar, também foram realizados testes com a base reduzida para 9 colunas.

Adotando-se 50 como parâmetro de máximo de grupos, o aplicativo atingiu 43 grupos, conforme mostrado na Figura 5.79. Número muito elevado em relação ao esperado, porém aqui faz algum sentido, pois um grupo é muito maior que os demais, contendo 3.290 objetos, representando cerca de 66% da base, como pode ser facilmente notado na Figura 5.80, sendo provavelmente composto de clientes que não trocaram de concessionária. Os outros grupos, alguns unitários até, são resultados da pulverização do grupo de clientes que trocaram o serviço e de clientes fiéis, mas com características diferentes o bastante para que fossem percebidos como grupos distintos.

Cluster_demographic

Result created: 11/05/04 17:27:03

Result File	:	C:\TEMP\PK1AXLD7.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	2
Maximum Number of Clusters	:	50
Improvement Over Last Pass	:	2.00
Similarity Threshold	:	0.50
Mining Run Outputs		
Number of Passes Performed	:	2
Number of Clusters	:	43
Improvement Over Last Pass	:	0.00
Global Condorcet Value	:	0.5745
Parallel Parameters: Applicable For Results Created On Parallel Systems Only		
Records Processed Before Merging	:	10000
NStripe	:	1
NSpan	:	1440

Cluster Characteristics :

Id	Cluster Size		Condorcet	Id	Cluster Size		Condorcet
	Absolute	Relative(%)	Value		Absolute	Relative(%)	Value
0	3290	65.80	0.5747	22	24	0.48	0.6275
1	337	6.74	0.5774	23	8	0.16	0.6611
2	29	0.58	0.5608	24	11	0.22	0.6269
3	442	8.84	0.5662	25	4	0.08	0.6389
4	192	3.84	0.5729	26	7	0.14	0.6323
5	58	1.16	0.5692	27	1	0.02	0.9809
6	46	0.92	0.5638	28	1	0.02	0.9770
7	8	0.16	0.6313	29	7	0.14	0.6566
8	134	2.68	0.5669	30	3	0.06	0.7037
9	13	0.26	0.5823	31	7	0.14	0.6559
10	237	4.74	0.5623	32	2	0.04	0.7424
11	12	0.24	0.5881	33	1	0.02	0.9829
12	4	0.08	0.6494	34	2	0.04	0.7987
13	4	0.08	0.6984	35	4	0.08	0.7063
14	13	0.26	0.5926	36	2	0.04	0.7506
15	7	0.14	0.6642	37	1	0.02	0.9756
16	3	0.06	0.6900	38	4	0.08	0.7330
17	2	0.04	0.8357	39	2	0.04	0.7559
18	28	0.56	0.5600	40	2	0.04	0.7659
19	23	0.46	0.6183	41	1	0.02	0.9862
20	12	0.24	0.6114	42	3	0.06	0.7751
21	9	0.18	0.6153				

Figura 5.79: relatório de saída do método "*Cluster demographic*" para a base de clientes "*churn*" reduzida.

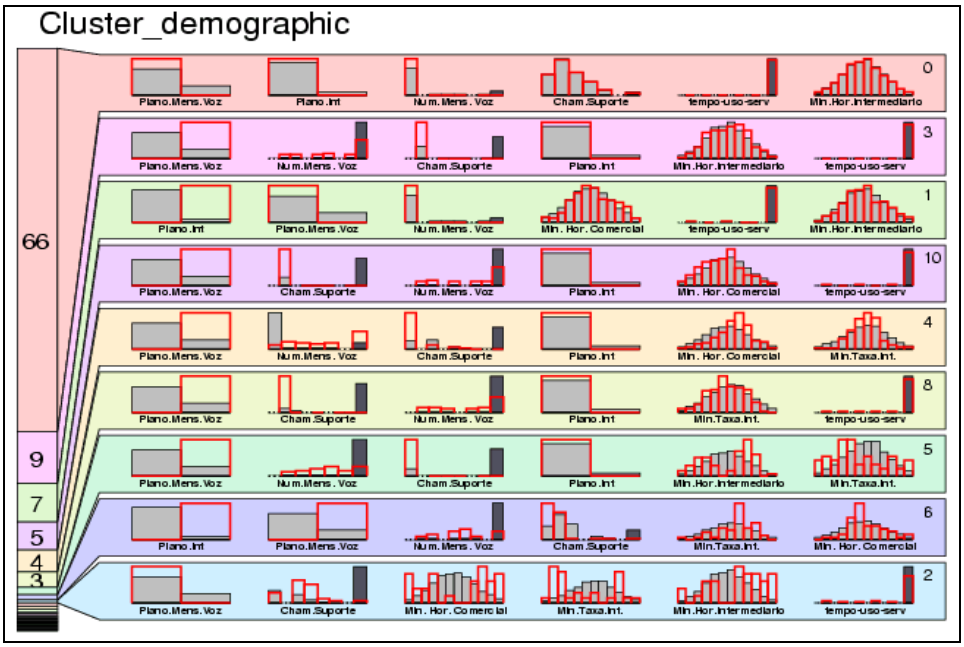


Figura 5.80: gráfico do método "*Cluster demographic*" para a base de clientes "*churn*" reduzida.

O teste seguinte, usando o método "*Cluster neural*", conforme mostrado na Figura 5.81, apresentou um resultado coincidente com o apresentado pelo mesmo método aplicado sobre a base completa: 13 grupos, considerando-se que grupos com 0 objetos não devem ser contados. A alocação dos objetos pelos grupos novamente não apresenta contribuição conclusiva para processo de análise do agrupamento.

Cluster_neural

Result created: 11/05/04 17:31:41

Result File	:	C:\TEMP\E52BHS4N.C2T
Mode	:	Training
User Specified Parameters		
Maximum Number of Passes	:	10
Maximum Number of Clusters	:	9
Mining Run Outputs		
Number of Passes Performed	:	10
Number of Clusters	:	15
Deviation	:	0.03

Cluster Characteristics :

Id	Cluster Size Absolute	Relative(%)	Id	Cluster Size Absolute	Relative(%)
0	399	7.98	8	342	6.84
1	333	6.66	9	506	10.12
2	460	9.20	10	681	13.62
3	77	1.54	11	0	0.00
4	49	0.98	12	558	11.16
5	54	1.08	13	508	10.16
6	352	7.04	14	681	13.62
7	0	0.00			

Figura 5.81: relatório de saída do método "*Cluster neural*" para a base de clientes "*churn*" reduzida.

Na observação do gráfico gerado pelo método "*Cluster neural*", visto em parte na Figura 5.82, chama a atenção a larga contribuição da opção pelo plano

internacional e correio de voz. Fato que pode ser notado igualmente no gráfico gerado pelo método “*Cluster demographic*”, visto na Figura 5.80 e já apontado desde os testes com *SearchCluster*, reafirmando a importância de tais atributos para a manutenção da fidelidade dos clientes da concessionária de serviços telefônicos.

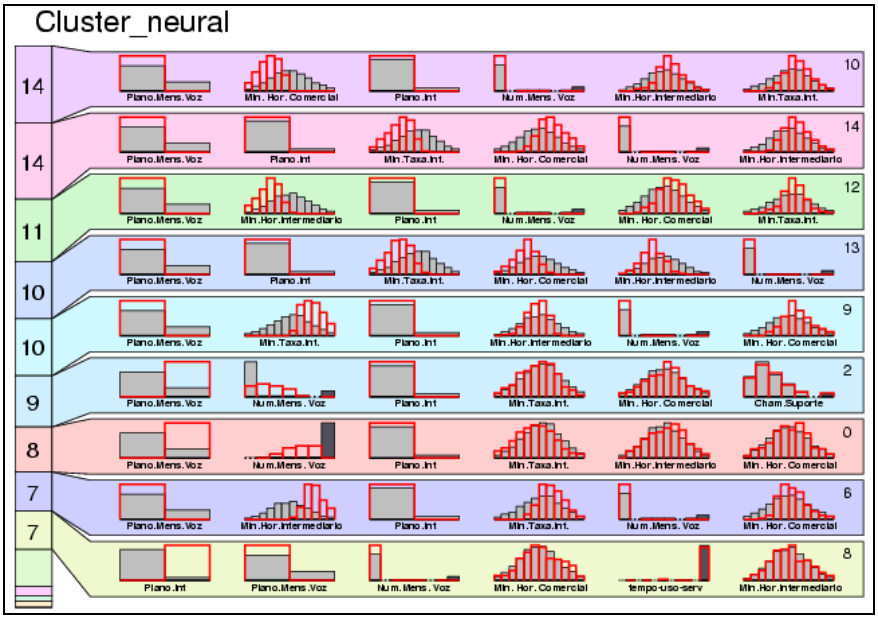


Figura 5.82: gráfico do método “*Cluster neural*” para a base de clientes “*churn*” reduzida.

CAPÍTULO 6
CONCLUSÃO

A análise de agrupamentos e a conseqüente extração de conhecimento de grandes bases de dados integram um recente ramo da atividade computacional. Diversas técnicas e softwares estão sendo estudados e desenvolvidos para auxiliar os especialistas nesta complexa, e cada vez mais importante, tarefa.

Com o intuito de comprovar a validade de algumas das técnicas que têm sido propostas para análise e agrupamento de dados, o objetivo deste trabalho foi desenvolver um aplicativo que implementasse métodos particionais, hierárquicos e visuais, aliando eficácia e funcionalidade na tarefa de mineração de dados.

Para desenvolvimento do aplicativo, intitulado *SearchCluster,* foi empregada a linguagem de programação *Microsoft Visual Basic*, a qual permite a construção de softwares com uma interface bastante amigável e intuitiva, de maneira que possam ser analisadas bases com diferentes tamanhos e conformações, desde que estejam em formato texto, o que proporciona maior rapidez na leitura dos dados de entrada. Apesar de não possuir notória habilidade em manipular recursos gráficos em alta velocidade, o *Visual Basic* apresenta excelente capacidade na leitura dos arquivos e na realização das operações aritméticas de forma rápida e precisa, compensando sobremaneira suas deficiências. O produto final, geralmente obtido, é representado por aplicativos de custo computacional compensador e com código fonte de fácil manutenção. Ainda, se recursos mais poderosos do que aqueles que podem ser desenvolvidos exclusivamente com o *Visual Basic* realmente forem necessários*,* é possível introduzir funções para cálculos e exibição de telas gráficas, entre outras, desenvolvidas em linguagens que tiram melhor proveito da interface de programação de aplicativos (API, *Application Programming Interface*) do sistema operacional *Microsoft® Windows*.

A segmentação de dados realizada pelo *SearchCluster* é realizada através da aplicação de dois métodos: um particional e um hierárquico. O que provê o software de duas opções de análise, que sendo aplicadas em conjunto, proporcionam ao usuário um apoio mais variado para sua descoberta de conhecimento.

O método particional *"K-Means"*, uma das técnicas de agrupamento implementadas, é baseado no conceito estatístico do centróide para formação de grupos cujos elementos tenham a mais alta similaridade entre si e a mais baixa similaridade em relação aos elementos dos outros grupos. Apesar deste método ser passível de críticas pela sua inaptidão em distinguir grupos com quantidades de objetos muito discrepantes e bases com formas não convexas, a sua implementação pelo *SearchCluster* mostrou-se eficiente, conseguindo produzir agrupamentos de muito boa qualidade.

Como ponto crucial deste trabalho, a determinação automática do número de grupos é uma tarefa delicada e feita por tentativas sucessivas, ou seja, para que seja determinado o número ótimo de grupos, a base é dividida inicialmente em dois grupos, depois em três e assim sucessivamente até alcançar o número máximo de grupos pretendido pelo usuário. A cada agrupamento um índice é calculado e comparado com o do anterior, sempre conservando aquele que otimizou o critério, Na implementação do *SearchCluster* foi empregado o método estatístico proposto por Calinski e Harabasz para verificar a qualidade da formação realizada. Com efeito, o sucesso das segmentações obtidas deve-se em muito à aplicação deste método, que já havia sido apontado como o melhor, entre 30 critérios de otimização de agrupamentos comparados em um estudo realizado por Milligan e Cooper, em 1985.

O outro método de segmentação disponível no *SearchCluster* é a segmentação hierárquica aglomerativa que, partindo da consideração individual dos objetos vai criando fusões sucessivas até que todos os objetos estejam em um único grupo ou a satisfação de certas condições de conclusão. Oferecida na forma de dendrogramas, a implementação desta técnica pelo método do Vizinho Mais Próximo é um recurso visual que não determina objetivamente o número ideal de grupos ou a alocação dos objetos, como as opções de saída em arquivos ou tela de resumo, como na segmentação particional, mas permite a visualização, em forma de árvore, das ligações dos objetos e subgrupos, possibilitando uma análise mais subjetiva por parte do especialista.

Ainda no âmbito dos recursos gráficos, a visualização da distribuição espacial relativa dos objetos e do centro dos grupos formados, determinados com a segmentação particional prévia, torna-se possível através do emprego da técnica de "Coordenadas Estrela", que produz excelentes resultados na transposição de objetos com n dimensões para um plano bidimensional. Este recurso visual colabora de maneira muito efetiva na análise de agrupamentos, ao dar a percepção virtual do

posicionamento dos objetos no espaço, no formato da base e dos grupos em que esta se divide. Aliando-se a isto a representação dos objetos de diferentes grupos com diferentes cores e a possibilidade de alterar o peso dos atributos, a visualização torna-se um importante auxílio ao usuário em sua tarefa de análise de dados.

Com capacidade máxima de 30 atributos, o *SearchCluster* possui uma limitação teórica de 100.000 objetos, porém este número é dependente da configuração da máquina utilizada em sua execução. Nos casos em que a base a ser estudada exceder a capacidade do ambiente, pode ser realizada uma segmentação por amostragem. Contudo, durante os testes de segmentação de diversas bases, o aplicativo desenvolvido mostrou-se satisfatório: é estável, confiável e de baixo custo computacional, proporcional ao tamanho das bases. Realiza segmentações que podem ser consideradas de boa qualidade e exibe recursos de visualização bastante precisos, que podem ser de real utilidade para quem está envolvido na tarefa de analisar e, possivelmente, extrair conhecimentos de grandes bases de dados.

Para os estudos de caso, foram selecionadas três bases de dados: DNA da cana de açúcar, automóveis e clientes "*churn*". Tais bases foram analisadas e segmentadas utilizando-se, além do *SearchCluster*, quatro ferramentas especializadas em mineração de dados disponíveis no mercado: o *gCLUTO*, o *Enterprise Miner*, o *Mineset* e o *Intelligent Miner*. Nos testes realizados, o *SearchCluster*, apresentou resultados muito semelhantes ou, ao menos, bastante compatíveis com aqueles apresentados pelos softwares comerciais, como no caso dos automóveis, onde a determinação do número de grupos e a distribuição dos objetos, tanto para base original quanto para a reduzida, foi coerente entre as ferramentas utilizadas.

Naturalmente, as cinco ferramentas não coincidiram exatamente em todos os resultados obtidos durante a análise das três bases, provavelmente devido às diferentes técnicas de segmentação e de avaliação de qualidade utilizadas, como o CCC no *Enterprise Miner* e o Critério de *Condorcet*, no *Intelligent Miner*. Diferentes abordagens podem, portanto, produzir resultados inesperados ou inapropriados para a base analisada, demonstrando que dificilmente uma só ferramenta pode assumir a condição de solução definitiva, devendo-se considerar a possibilidade da ocorrência de casos que requeiram a utilização de dois ou mais aplicativos, a fim de que os resultados gerados sejam comparados e combinados, proporcionando aos especialistas insumos mais abrangentes e diferentes ângulos de percepção para deduções mais úteis e confiáveis.

Acredita-se que algumas melhorias devam ser desenvolvidas no presente trabalho: o estudo da possibilidade de refinamento do algoritmo, de forma que bases extremamente grandes possam ser analisadas na íntegra; a inclusão de uma técnica que otimize a divisão inicial dos grupos, de onde partiriam as realocações dos objetos em busca da formação ideal; A implementação da possibilidade do usuário optar entre variados métodos particionais de segmentação e critérios de avaliação da qualidade de agrupamentos a serem empregados no processamento; a implementação da visualização de objetos em três dimensões, com possibilidade de rotação em todas as direções; e a implementação da visualização de um mapa de correlação de grupos.

Por fim, pode-se afirmar que a implementação de um procedimento interativo para agrupamento de dados foi atingida com sucesso. O aplicativo desenvolvido com o uso de métodos bastante estudados e uma linguagem de programação consagrada, executa sua tarefa de forma eficiente, produzindo resultados estáveis e coerentes e auxiliando na aplicação e difusão das técnicas de mineração de dados e descoberta de conhecimento.

REFERÊNCIAS BIBLIOGRÁFICAS

[1] HAN, J. KAMBER, M. *Data Mining Concepts and Techniques*, San Francisco, Morgan Kaufmann Publishers, 2001.

[2] KANDOGAN, ESER. “Visualizing Multi-dimensional Clusters, Trends, and Outiliers using Star Coordinates”, In: *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, pp 107-116, 2001.

[3] CRIVISQUI, EDUARDO, Métodos de Clasificación, Programme de Recherche et d'Enseignementen Statistique Appliquée, Notas de cursos de los "Seminarios locales" de métodos exploratorios multivariados, 1999. Disponível em: http://www.ulb.ac.be/assoc/presta/Cursos/cursos.html.

[4] Clustan Ltd. Disponível em: http://www.clustan.com, 2003.

[5] BATISTA, PAULO, SILVA, MÁRIO J., “*Prospecção dos dados de Acesso a um Servidor de Notícias na Web*”, In: Relatório Técnico ARIADNE TR-99-5, Universidade de Lisboa, Lisboa,1999.

[6] J. M. Pessanha, R. M. G. Velásquez, A. C. G. Melo, R. P. Caldas, Técnicas de *Cluster Analysis* na Construção de Tipologias de Curva de Carga, XV Seminário Nacional de Distribuição de Energia Elétrica – SENDI 2002, Bahia, Brasil, 2002.

[7] FURTADO, MARIA INÊS VASCONCELOS, *inteligência Competitiva para o Ensino Superior Privado: Uma Abordagem Através da Mineração de textos,* Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2004.

[8] LIMA, J. W. MARANGON, et al, “Regulação da Qualidade de Suprimento por Comparação na Receita Permitida”, *XVI SNPTEE*, São Paulo, outubro**,**2001.

[9] RODRIGUES, M. C. PRATES, “Potencial de Desenvolvimento dos Municípios Fluminenses: Uma Metodologia Alternativa ao IQM, com Base na Análise Fatorial Exploratória e na Análise de *Clusters*”, *Caderno de Pesquisas em Administração*, v. 09, nº 1, pp 75-89, 2002.

[10] SANTOS, S. M., NORONHA, C. P., “Padrões espaciais de mortalidade e diferenciais sócio-econômicos na cidade do Rio de Janeiro”, *Caderno de Saúde Pública*, v. 17, nº 5, pp 1099-1110, 2001.

[11] CARDOSO, PAULO, CORREIA, MANUELA, “Aplicação de dados de Detecção Remota em Ecologia”, ESIG’99, V Encontro Sobre Sistemas de Informação Geográfica, II Congresso da Informação Geográfica, Oeiras, Portugal, 1999.

[12] FAYYAD, U. M., PIATETESKY-SHAPIRO, G., SMITH, P., “Knowledge Discovery and Data Mining: Towards a Unifying Framework”, *II International Conference on KD & DM*, Oregon, agosto, 1996.

[13] ESTIVILL-CASTRO, VLADIMIR, “Why so many clustering algorithms – A position Paper”, ACM *SIGKDD Explorations Newsletter*, v. 04, Issue 1, pp 65-75, 2002.

[14] FEITEIRA, JOSÉ FLÁVIO SILVEIRA, *Extensão Sintética de Respostas Impulsivas de Modelos Reduzidos para Acústica de Salas*. Tese de D. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2002.

[15] StatSoft, Inc., STATISTICA for Windows [Computer program manual]. Tulsa, OK: StatSoft, Inc., 1995.

[16] MACHADO FILHO, OTTO MOURA, *Exploração e Análise de Agrupamento de Dados.* Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2002.

[17] KAUFMAN, L., ROUSSEEUW, P. J., *Finding Groups in Data: An introduction to Cluster Analysis*. New York: John Wiley & Sons, 1990.

[18] COELHO, P. S. S., EBECKEN N. F. F,. Segmentação de Dados em Um Número Desconhecido de Grupos Usando Algoritmos Genéticos,.Simpósio Brasileiro de Pesquisa Operacional – SBPO 2001, São Paulo, Brasil, 2001.

[19] CALINSKI, T., HARABASZ, J., “A Dendrite Method for Cluster Analysis”, *Communications in Statistics*, v. 3, n. 1, pp. 1-27, 1974.

[20] MILLIGAN, G. W., COOPER, M. C., “An examination of Procedures for Determining the Number of Clusters of Data set”, Psychometrika, pp 159-179, 1985.

[21] ALIS, O. FARUK, KARAKURT, ERTAN, MELLI, PIERO, “Data Mining for Database Marketing at Garanti Bank”, In: Ebecken, N. F. F., Brebbia, C. A., et

al. (Eds.) *Data Mining II*, Information and Communication Technologies volume 25, Wessex Institute of Technology, pp 93 – 107, 2000.

[22] SAS Institute Inc, *Enterprise Miner™ Release* 4.3 *Cubic Clustering Criterion Documentation©*, SAS Institute Inc, NC, 2002.

[23] PAKHIRA, M. K., BANDYOPADHYAY, S, MAULIK, U., "Validity Index For Crisp and Fuzzy Clusters", Pattern Recognition, vol. 37, n° 3, pp 487-501, 2004.

[24] SOUKUP, TOM, DAVIDSON, IAN, *Visual Data Mining: techniques and tools for data visualization and mining*, New York, John Wiley & Sons, Inc., 2002.

[25] PUNTAR, SÉRGIO GONÇALVES, *Métodos e Visualização de Agrupamento de Dados.* Tese de M. Sc., COPPE/UFRJ, Rio de Janeiro, RJ, Brasil, 2003.

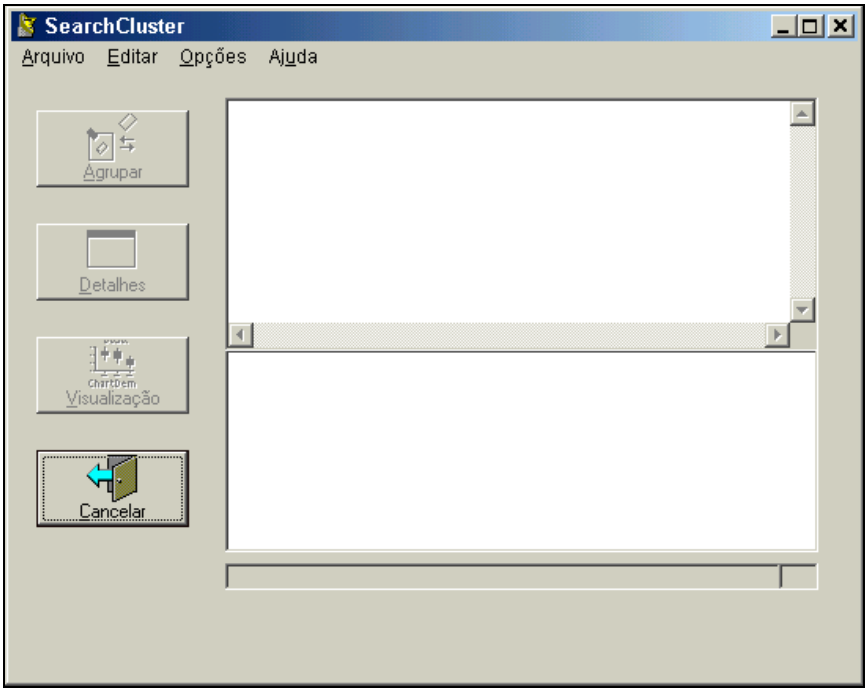
[26] Silicon Graphics, Inc, *Mineset™ Enterprise Edition User´s Guide for the Windows® Client*, Silicon Graphics, Inc, CA, 2000.

[27] HAIR JR., JOSEPH F.; ANDERSON, ROLPH E.; TATHAM, RONALD L.; BLACK, WILLIAM C*. Multivariate data analysis*. 5.ed., Prentice Hall, 1998.

APÊNDICE I

Listagem do programa SearchCluster

FORMULÁRIO FRMCLUSTER




```
Option Explicit
Dim arquivo, msg, arq As String
Dim erro, T, desvpad, medbase() As Single
Dim nrcluster, k As Integer
Dim carrega_ok As Boolean
Dim arqt As Integer

Private Sub Form_Load()
 monta_tabela_cores
 FrmCluster.Agrupar.Enabled = True
 arqt = 0

End Sub

Private Sub menu_arqentrada_Click()
 dialog.Flags = cdlOFNFileMustExist
 dialog.ShowOpen
 arq = dialog.FileName
 st(1) = "Arquivo de dados - leitura: " + arq
```

```
 monta_results1
 If st(4) <> Empty Then
  FrmCluster.Agrupar.Enabled = True
  FrmCluster.Visualização.Enabled = True
 End If
 Agrupamentos_ok = False
 FrmCluster.results2 = Empty

End Sub

Private Sub menu_arqsaida_clu_Click()
 If FrmCluster.menu_arqsaida_clu.Checked = True Then
  FrmCluster.menu_arqsaida_clu.Checked = False
 Else
  FrmCluster.menu_arqsaida_clu.Checked = True
 End If
 st(2) = "Arquivo de dados - gravação(grupos): saida_clu.txt"
 monta_results1

End Sub

Private Sub menu_arqsaida_obj_Click()
  If FrmCluster.menu_arqsaida_obj.Checked = True Then
  FrmCluster.menu_arqsaida_obj.Checked = False
 Else
  FrmCluster.menu_arqsaida_obj.Checked = True
 End If
 st(3) = "Arquivo de dados - gravação(objetos): saida_obj.txt"
 monta_results1

End Sub

Private Sub menu_ed_copiar_Click()

 If results1.SelLength > 0 Then
   Clipboard.SetText results1.SelText
   results1.SelLength = 0
 Else
  Clipboard.SetText results2.SelText
 End If

End Sub

Private Sub menu_parm_Click()
Dim i As Integer

 For i = 4 To 40
  st(i) = Empty
 Next i
```

```
 FrmCluster.Enabled = False
 FrmParm.Show

End Sub

Private Sub menu_ajsobre_Click()
 FrmCluster.Enabled = False
 FrmSobre.Show

End Sub

Private Sub menu_arqsair_Click()
 End

End Sub

Private Sub Agrupar_Click()
Dim i As Integer
Dim msg1, msg2 As String
FrmCluster.results2 = Empty
FrmCluster.Refresh

If arq = Empty Then
 msg1 = "Arquivo de entrada: "
 msg2 = "preenchimento obrigatório."
 MsgBox msg1 + vbCrLf + vbCrLf + msg2, vbExclamation, "Erro"
Else
 FrmCluster.Detalhes.Enabled = False
 processamento
 FrmCluster.Refresh
End If

Agrupamentos_ok = True
FrmCluster.menu_ed_copiar.Enabled = True

End Sub

Private Sub Cancelar_Click()
 End

End Sub

Private Sub detalhes_Click()

Dim i, x As Integer
Dim erromd As Single
Dim texto, esp As String

FrmResumo.txtresumo = Empty
i = 1
```

```
While i <= nrotimocl

 texto = CStr(i)
 texto = Space((4 - Len(texto)) * 2) + texto
 FrmResumo.txtresumo = FrmResumo.txtresumo + texto + Space(12)

 texto = CStr(tabresumo(i, 31))
 texto = Space((6 - Len(texto)) * 2) + texto
 FrmResumo.txtresumo = FrmResumo.txtresumo + texto + Space(11)

 If tabresumo(i, 32) < 0.000001 Or tabresumo(i, 32) > 100000 Then
  texto = Format(tabresumo(i, 32), "000E+")
 Else
   texto = Format(tabresumo(i, 32), "0.00000")
 End If
 texto = Space((14 - Len(texto)) * 2) + texto
 FrmResumo.txtresumo = FrmResumo.txtresumo + texto + Space(3)

 erromd = tabresumo(i, 32) / tabresumo(i, 31)
 If erromd < 0.000001 Or erromd > 100000 Then
  texto = Format(erromd, "0000E+")
 Else
  texto = Format(erromd, "0.000000")
 End If
 texto = Space((14 - Len(texto)) * 2) + texto
 FrmResumo.txtresumo = FrmResumo.txtresumo + texto + vbCrLf

 i = i + 1
Wend

FrmResumo.Show

End Sub
Private Sub visualização_Click()
Dim i As Integer

If Agrupamentos_ok = False Then
 inicia_atributos
 If carrega_ok Then
  For i = 1 To linmax
   atrib(i, 32) = 31
   tabxyobj(i, 3) = 31
  Next i
  FrmGrafico.menu_cluster.Enabled = False
  FrmGrafico.menu_objgraf.Checked = True
  FrmGrafico.menu_legenda.Enabled = False
  FrmGrafico.Show
 End If
Else
```

```
 abre_arquivo
 carrega_tab_atributos
 fecha_arquivo

 FrmGrafico.menu_cluster.Enabled = True
 FrmGrafico.menu_objgraf.Checked = True
 FrmGrafico.menu_legenda.Enabled = True
 FrmGrafico.Show

End If

End Sub

Sub inicia_atributos()
On Error GoTo ChamaRotinaErro
Dim vazio As String

Dim i As Integer

 msg = "Inicialização de atributos"

 ProgressBar1.Value = 0.0001
 Text1.Text = Empty
 Text1.Refresh

 linmax = 1
 abre_arquivo
 Input #arquivo, vazio
 While Not EOF(arquivo)
  linmax = linmax + 1
  Input #arquivo, vazio
 Wend
 fecha_arquivo
 colmax = FrmParm.nmratrib
 Erase tabxycl, atrib, tabxyobj, tabresumo, medbase
 Erase tabdcl, tabagrcl, tabagrobj
 ReDim atrib(linmax, 32), tabxyobj(linmax, 3), medbase(colmax)
 abre_arquivo
 carrega_tab_atributos
 fecha_arquivo
Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Private Sub processamento()
On Error GoTo ChamaRotinaErro

Dim W, B, G, Gmax As Single
```

```
Dim ist, i As Integer
Dim s1 As Single

Gmax = -900000000000#
T = 0
ist = 11
inicia_atributos
linmax = linmax
If Not carrega_ok Then
 Exit Sub 'não conseguiu carregar a tabela de atributos, encerra processamento'
End If

calcula_dif_geral

For nrcluster = FrmParm.nmrmincluster To FrmParm.nmrmaxcluster

 W = 9000000000000#

 If nrcluster >= linmax Then
  Exit For 'interrompe a busca com nmr de cluster > metade do nmr objs
 End If
 monta_cluster_inicial
 calcula_erro
 realoca_objetos
 calcula_erro
 If erro = 0 Then
  Exit For 'interrompe a busca com erro = 0
 End If

 msg = "Processamento dos resultados"
 W = erro
 B = T - W
 k = nrcluster
 G = ((linmax - k) * B) / ((k - 1) * W)
 If FrmParm.chkerros = 1 Then
  ist = ist + 1
  s1 = k
  st(ist) = CStr(k) + " grupos: "
  s1 = W
  st(ist) = st(ist) + " erro " + CStr(W)
  s1 = G
  st(ist) = st(ist) + "  Função Objetivo " + CStr(G)
  monta_results1
 End If
 If G > Gmax Then
  Gmax = G
  nrotimocl = k
  s1 = k
  FrmCluster.results2 = "Número ótimo de grupos: " + CStr(k) + vbCrLf + vbCrLf
  s1 = W
```

```
 FrmCluster.results2 = FrmCluster.results2 + "Erro quadrático: " + CStr(W) + vbCrLf
 s1 = Gmax
 FrmCluster.results2 = FrmCluster.results2 + "Função objetivo: " + CStr(Gmax)
 carrega_tabresumo
 For i = 1 To linmax
  tabxyobj(i, 3) = atrib(i, 32)
 Next i
End If
If FrmCluster.menu_arqsaida_obj.Checked = True And _
   FrmParm.chkarquivos = 1 Then
 grava_arquivo_obj
End If
ProgressBar1.Value = nrcluster * 100 / FrmParm.nmrmaxcluster
Text1.Text = nrcluster
Text1.Refresh
Next nrcluster

If FrmCluster.menu_arqsaida_obj.Checked = True And _
   FrmParm.chkarquivos = 0 Then
 grava_arquivo_obj
End If
If FrmCluster.menu_arqsaida_clu.Checked = True Then
 grava_arquivo_clu
End If
FrmCluster.Detalhes.Enabled = True
FrmCluster.Visualização.Enabled = True
FrmGrafico.menu_cluster.Enabled = True
FrmGrafico.menu_legenda.Enabled = True

If nrotimocl > 30 Then
 FrmGrafico.menu_grupocor.Enabled = False
Else
 FrmGrafico.menu_grupocor.Enabled = True
End If

Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub abre_arquivo()
On Error GoTo ChamaRotinaErro
Dim vazio As String

arquivo = FreeFile
msg = "Abertura do arquivo de dados"

Open arq For Input As arquivo
Exit Sub
```

```
ChamaRotinaErro: RotinaErro

End Sub

Sub carrega_tab_atributos()
On Error GoTo ChamaRotinaErro

Dim col As Integer
Dim despcol As Single
Dim desplin As String

linmax = 1
carrega_ok = True
If FrmParm.chkplin = 1 Then 'despreza 1.a linha
 Input #arquivo, desplin
End If
If Not EOF(arquivo) Then
 msg = "Leitura do arquivo de dados." + vbCrLf + _
       "Verifique Número de atributos e Colunas identificadoras."
 If FrmParm.chkpcol.Value = 1 Then
  Input #arquivo, despcol
 End If
 For col = 1 To colmax
  Input #arquivo, atrib(linmax, col)
  atrib(linmax, col) = atrib(linmax, col) * FrmParm.Pesocol(col)
  atrib(linmax, 31) = atrib(linmax, 31) + atrib(linmax, col)
  medbase(col) = medbase(col) + atrib(linmax, col)
 Next col
 If FrmParm.chkucol.Value = 1 Then
    Input #arquivo, despcol
 End If
Else
 msg = "Arquivo de dados vazio"
 GoTo ChamaRotinaErro
End If

While Not EOF(arquivo)
 linmax = linmax + 1
 If FrmParm.chkpcol.Value = 1 Then
  Input #arquivo, despcol
 End If
 For col = 1 To colmax
  Input #arquivo, atrib(linmax, col)
  atrib(linmax, col) = atrib(linmax, col) * FrmParm.Pesocol(col)
  atrib(linmax, 31) = atrib(linmax, 31) + atrib(linmax, col)
  medbase(col) = medbase(col) + atrib(linmax, col)
 Next col
 If FrmParm.chkucol.Value = 1 Then
  Input #arquivo, despcol
```

```
 End If
Wend

If FrmParm.chkulin.Value = 1 Then 'despreza ultima linha
 For col = 1 To colmax
  medbase(col) = medbase(col) - atrib(linmax, col)
 Next col
 linmax = linmax - 1
End If

Exit Sub

ChamaRotinaErro: RotinaErro
carrega_ok = False
FrmCluster.Agrupar.Enabled = False
FrmCluster.Visualização.Enabled = False


End Sub

Sub fecha_arquivo()
On Error GoTo ChamaRotinaErro

msg = "Fechamento do arquivo de dados"
Close arquivo
Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub calcula_dif_geral()

Dim i As Integer
Dim dif, lin As Single
msg = "Cálculo da diferença geral dos dados"

For i = 1 To colmax 'calcula media de todos os objetos
 medbase(i) = medbase(i) / linmax
Next i
For lin = 1 To linmax 'calcula o erro
 For i = 1 To colmax
  If FrmParm.Pesocol(i) <> 0 Then
    dif = atrib(lin, i) - medbase(i) 'calcula dif entre o atrib de cada obj e a media desse
atrib de todos obj da base
    T = T + (dif * dif)
  End If
 Next i
Next lin
```

```
End Sub

Sub monta_cluster_inicial()
Dim nrobj, lincl, cont, i As Integer
Dim lin As Single

msg = "Montagem de clusters iniciais"
nrobj = (linmax / nrcluster) - 0.5   'determina qtd obj ini p/ cluster
lin = 1
For cont = 1 To nrcluster        'inclui nr cluster ini para cada objeto
 For i = 1 To nrobj
  atrib(lin, 32) = cont
  lin = lin + 1
 Next i
Next cont
For i = lin To linmax
 atrib(i, 32) = nrcluster
Next i
Erase tabcl

For lin = 1 To linmax
 lincl = atrib(lin, 32)
 For i = 1 To colmax
  tabcl(lincl, i) = tabcl(lincl, i) + atrib(lin, i) 'soma os atrib dos obj de um cluster
 Next i
 tabcl(lincl, 31) = tabcl(lincl, 31) + 1           'soma qtd obj do cluster
Next lin

For lincl = 1 To nrcluster
 For i = 1 To colmax
  tabcl(lincl, i) = tabcl(lincl, i) / tabcl(lincl, 31)
 Next i
Next lincl

End Sub

Sub calcula_erro()
Dim dif, dist, lin As Single
Dim lincl, i As Integer

msg = "Cálculo do erro dos atributos"
erro = 0

For i = 1 To FrmParm.nmrmaxcluster
 tabcl(i, 32) = 0
Next i

For lin = 1 To linmax 'calcula o erro
 dist = 0
 lincl = atrib(lin, 32)
```

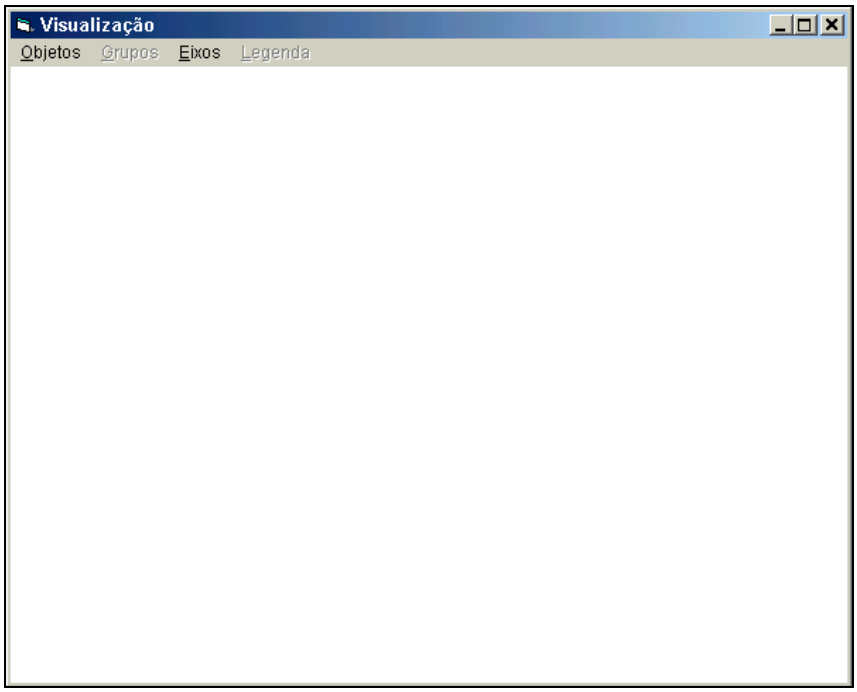
```
 For i = 1 To colmax
  If FrmParm.Pesocol(i) <> 0 Then
   dif = (atrib(lin, i) - tabcl(lincl, i)) 'calcula dif entre o atrib de cada obj e a media desse
atrib do cluster
   dist = dist + (dif * dif)
   erro = erro + (dif * dif)
  End If
 Next i
 atrib(lin, 31) = dist
 tabcl(lincl, 32) = tabcl(lincl, 32) + atrib(lin, 31)
Next lin

End Sub

Sub realoca_objetos()
Dim lincl, i, cluster_origem As Integer
Dim lin, dif, dist, nv_nr_obj, nr_obj_origem As Single
Dim troca As Boolean

msg = "Realocação de objetos"

troca = True
While troca
troca = False
lin = 1
While lin <= linmax
 cluster_origem = atrib(lin, 32)
 For lincl = 1 To nrcluster
  dist = 0
  For i = 1 To colmax 'calcula a distancia de cada obj ao centro de cada cluster
   If FrmParm.Pesocol(i) <> 0 Then
    dif = (atrib(lin, i) - tabcl(lincl, i))
    dist = dist + (dif * dif)
   End If
  Next i
  If dist < atrib(lin, 31) Then 'coloca o objeto no cluster com a menor distancia
   atrib(lin, 31) = dist
   atrib(lin, 32) = lincl
  End If
 Next lincl
 If atrib(lin, 32) <> cluster_origem Then
  troca = True
  lincl = cluster_origem
  'calcula novo centro cluster origem
  nv_nr_obj = tabcl(lincl, 31) - 1
  nr_obj_origem = nv_nr_obj
  For i = 1 To colmax
   If FrmParm.Pesocol(i) <> 0 Then
    tabcl(lincl, i) = tabcl(lincl, i) * tabcl(lincl, 31)
```

```
    tabcl(lincl, i) = tabcl(lincl, i) - atrib(lin, i) 'diminui os atrib do obj que mudou de
cluster
    If nv_nr_obj <> 0 Then
     tabcl(lincl, i) = tabcl(lincl, i) / nv_nr_obj
    End If
   End If
  Next i
  tabcl(lincl, 31) = nv_nr_obj
  'calcula novo centro cluster destino
  lincl = atrib(lin, 32)
  nv_nr_obj = tabcl(lincl, 31) + 1
  For i = 1 To colmax
   If FrmParm.Pesocol(i) <> 0 Then
    tabcl(lincl, i) = tabcl(lincl, i) * tabcl(lincl, 31)
    tabcl(lincl, i) = tabcl(lincl, i) + atrib(lin, i) 'soma os atrib do obj que mudou de cluster
    tabcl(lincl, i) = tabcl(lincl, i) / nv_nr_obj
   End If
  Next i
  tabcl(lincl, 31) = nv_nr_obj
 End If
  lin = lin + 1

 If nr_obj_origem = 1 Then 'encerra a realocação qdo algum cluster ficar com apenas
um obj
  Exit Sub
 End If

Wend
Wend

End Sub

Sub carrega_tabresumo()

Dim i, c As Integer

msg = "carregamento da tabela de resumo"

Erase tabresumo

For i = 1 To nrcluster
 For c = 1 To colmax
  tabresumo(i, c) = tabcl(i, c)
 Next c
 tabresumo(i, 31) = tabcl(i, 31)
 tabresumo(i, 32) = tabcl(i, 32)
Next i

End Sub
```

```
Sub grava_arquivo_obj()
On Error GoTo ChamaRotinaErro

Dim i, c As Single
Dim linha, num, arqs As String

msg = "Gravação arquivo de objetos"
arqs = FreeFile

If FrmParm.chkarquivos = 1 Then
 Open "saida_obj" + Format(Format(nrcluster, "000"), "@@@") + ".txt" For Output As
arquivo
Else
 Open "saida_obj.txt" For Output As arquivo
End If

For i = 1 To linmax
 linha = Empty
 For c = 1 To colmax
  If atrib(i, c) = 0 Then
   num = CStr(0)
  Else
   If atrib(i, c) < 0.000001 Then
    num = Format(atrib(i, c), "0000E+")
   Else
    num = Format(atrib(i, c), "0.######")
   End If
  End If
  linha = linha + Format(num, "!@@@@@@@@@") + " "
 Next c
 linha = linha + Format(CStr(atrib(i, 31)), "!@@@@@@@@@") + " "

 linha = linha + Format(CStr(atrib(i, 32)), "@@@")

 Write #arquivo, linha
Next i

Close arquivo

Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub grava_arquivo_clu()
On Error GoTo ChamaRotinaErro

Dim i, c As Integer
Dim linha, num As String
```

```
msg = "Gravação arquivo de clusters"

arquivo = FreeFile

Open "saida_clu.txt" For Output As arquivo

i = 1
While i <= Int(nrotimocl)
 linha = Empty
 For c = 1 To colmax
  If c = 17 Then
   c = 17
  End If
  If tabresumo(i, c) = 0 Then
   num = CStr(0)
  Else
   If tabresumo(i, c) < 0.000001 Then
    num = Format(tabresumo(i, c), "0000E+")
   Else
    num = Format(tabresumo(i, c), "0.######")
   End If
  End If
  linha = linha + Format(num, "!@@@@@@@@@") + " "
 Next c
 linha = linha + Format(CStr(tabresumo(i, 31)), "@@@@") + " "

 linha = linha + Format(CStr(tabresumo(i, 32)), "!@@@@@@@@@")
 Write #arquivo, linha
 i = i + 1
Wend

Close arquivo

Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub RotinaErro()

  MsgBox msg + vbCrLf + vbCrLf + "Erro: " + Str(Err) + _
         " (" + Error + ")", vbCritical, "*** ERRO ***"

End Sub
```

FORMULÁRIO FRMGRAFICO




```
Dim tabmobj() As Single  'coluna 1: valor mínimo do atributo em todas as linhas
                         'coluna 2: valor máximo do atributo em todas as linhas
Dim escalarv, escalarh, escalah, escalav, desx, desy As Single
Dim escaladh, escaladv As Single
Dim Cosx(), Seny(), tabc(), tabcobj() As Single
Dim x, xi, xf, Y, yi, yf, cor, margem, imnivel As Integer
Dim nmrcl, auxi, auxj, ordem, cont, lintc As Integer
Dim objagr, clagr, pvez As Boolean
Dim R, G, B, tom As Integer

Private Sub Form_Activate()

ReDim tabmobj(colmax, 2)

calcula_Cosx_Seny
prepara

End Sub
Private Sub prepara()

objagr = False
objagrcl = False
pvez = True
calcula_xy_objeto
transforma_max_min_obj
escalah = 11600 / (maxx * 2)
escalav = 8600 / (maxy * 2)
escalarh = escalah / 4
```

```
escalarv = escalav / 4
cont = 3
escaladh = 0

End Sub

Private Sub Form_Resize()
ReDim tabmobj(colmax, 2)

If FrmGrafico.menu_eixos.Enabled = True Then
 calcula_Cosx_Seny
 prepara
Else
 FrmGrafico.Refresh
 If FrmGrafico.menu_objdendro.Checked = True Then
  monta_dendrograma "obj"
 Else
  monta_dendrograma "cl"
 End If
End If
End Sub

Private Sub Form_Unload(Cancel As Integer)
 x = Empty
 If clagr = True Then
  FrmHisto.Hide
 End If

End Sub

Private Sub menu_objgraf_Click()

FrmGrafico.menu_eixos.Enabled = True

If FrmGrafico.menu_objgraf.Checked = True Then
 FrmGrafico.menu_objgraf.Checked = False
 monta_grafico
Else
 FrmGrafico.menu_objgraf.Checked = True
 If FrmGrafico.menu_clgraf.Checked = False Then
  FrmGrafico.Refresh
 End If
 plota_objeto
End If
FrmGrafico.menu_cldendro.Checked = False
FrmGrafico.menu_clhist.Checked = False
FrmGrafico.menu_clhist.Enabled = False
FrmGrafico.menu_objdendro.Checked = False
FrmGrafico.menu_objhist.Checked = False
FrmGrafico.menu_objhist.Enabled = False
```

```
End Sub

Private Sub menu_objdendro_Click()

FrmGrafico.menu_eixos.Enabled = False
FrmGrafico.menu_idgrupo.Checked = False
FrmGrafico.menu_idgrupo.Enabled = False

If objagr = False Then
 FrmGrafico.menu_objhist.Enabled = True
 calcula_distancia_objeto
 executa_agregação_objeto
End If
escaladh = 11600 / ((linmax + 1) * 1)
escaladv = 8400 / (tabagrobj(linmax - 1, 5) * 1)
FrmGrafico.menu_cldendro.Checked = False
FrmGrafico.menu_clhist.Checked = False
FrmGrafico.menu_clhist.Enabled = False
FrmGrafico.menu_clgraf.Checked = False
FrmGrafico.menu_objgraf.Checked = False
FrmGrafico.menu_objdendro.Checked = True
FrmGrafico.menu_clhist.Checked = False
FrmGrafico.menu_clhist.Enabled = True
FrmGrafico.Refresh
monta_dendrograma "obj"

End Sub
Private Sub menu_objhist_Click()
 If FrmGrafico.menu_objhist.Checked = True Then
  FrmGrafico.menu_objhist.Checked = False
  FrmHisto.Hide
  FrmGrafico.Refresh
  monta_dendrograma "obj"
 Else
  FrmGrafico.menu_objhist.Checked = True
  tipo_histo = "obj"
  FrmHisto.Show
 End If

End Sub

Private Sub menu_clgraf_Click()

FrmGrafico.menu_eixos.Enabled = True

If FrmGrafico.menu_clgraf.Checked = True Then
 FrmGrafico.menu_clgraf.Checked = False
 FrmGrafico.menu_clhist.Enabled = False
 FrmGrafico.menu_idgrupo.Enabled = False
```

```
 FrmGrafico.menu_idgrupo.Checked = False
 monta_grafico
Else
 FrmGrafico.menu_clgraf.Checked = True
 FrmGrafico.menu_idgrupo.Enabled = True
 If FrmGrafico.menu_objgraf.Checked = False Then
 End If
 calcula_xy_cluster
 plota_cluster
End If

End Sub

Private Sub menu_cldendro_Click()

FrmGrafico.menu_eixos.Enabled = False
FrmGrafico.menu_idgrupo.Checked = False
FrmGrafico.menu_idgrupo.Enabled = False

If clagr = False Then
 FrmGrafico.menu_clhist.Enabled = True
 calcula_distancia_cluster
 executa_agregação_cluster
 escaladh = 11600 / ((nmrcl + 1) * 2)
 escaladv = 8400 / (tabagrcl(nmrcl - 1, 5) * 2)
End If
FrmGrafico.menu_clgraf.Checked = False
FrmGrafico.menu_objgraf.Checked = False
FrmGrafico.menu_cldendro.Checked = True
FrmGrafico.menu_objdendro.Checked = False
FrmGrafico.menu_objhist.Checked = False
FrmGrafico.menu_objhist.Enabled = False
FrmGrafico.Refresh
monta_dendrograma "cl"

End Sub

Private Sub menu_clhist_Click()
 If FrmGrafico.menu_clhist.Checked = True Then
  FrmGrafico.menu_clhist.Checked = False
  FrmHisto.Hide
  FrmGrafico.Refresh
  monta_dendrograma "cl"
 Else
  FrmGrafico.menu_clhist.Checked = True
  tipo_histo = "cl"
  FrmHisto.Show
 End If

End Sub
```

```
Private Sub menu_eixosexb_Click()

If FrmGrafico.menu_eixosexb.Checked = False Then
 FrmGrafico.menu_eixosexb.Checked = True
 traca_eixos
Else
 FrmGrafico.menu_eixosexb.Checked = False
 FrmGrafico.Refresh
 monta_grafico
End If

End Sub

Private Sub menu_grupocor_Click()
If FrmGrafico.menu_grupocor.Checked = False Then
 FrmGrafico.menu_grupocor.Checked = True
 exibe_legenda
Else
 FrmGrafico.menu_grupocor.Checked = False
 If FrmGrafico.menu_idgrupo.Checked = True Then
  exibe_idgrupo
 End If
 FrmGrafico.Refresh
End If

End Sub

Private Sub menu_idgrupo_Click()
If FrmGrafico.menu_idgrupo.Checked = False Then
 FrmGrafico.menu_idgrupo.Checked = True
 exibe_idgrupo
Else
 FrmGrafico.menu_idgrupo.Checked = False
 If FrmGrafico.menu_grupocor.Checked = True Then
  exibe_legenda
 End If
 FrmGrafico.Refresh
End If

End Sub

Sub calcula_Cosx_Seny_original_salvo()
Dim i As Integer
Dim ang, pi As Single
ReDim Cosx(colmax), Seny(colmax)

ang = 360 / colmax
pi = Atn(1) * 4
```

```
For i = 1 To colmax
 Cosx(i) = Cos((ang * (i - 1)) * (pi / 180))
 Seny(i) = Sin((ang * (i - 1)) * (pi / 180))
Next i

End Sub

Sub calcula_Cosx_Seny()
Dim i, eixo As Integer
Dim ang, pi As Single
ReDim Cosx(colmax), Seny(colmax)

ang = 360 / (colmax - contcol0)
pi = Atn(1) * 4
eixo = 0
For i = 1 To colmax
 If FrmParm.Pesocol(i) <> 0 Then
  If FrmParm.Pesocol(i) = 1 Then
   peso = 1
  Else
   peso = colmax / 10
   peso = 1 + peso * peso
   If FrmParm.Pesocol(i) < 1 Then
    peso = FrmParm.Pesocol(i)
   End If
  End If
 Cosx(i) = Cos((-1) * (ang * (eixo)) * (pi / 180)) * peso
 Seny(i) = Sin((-1) * (ang * (eixo)) * (pi / 180)) * peso
 eixo = eixo + 1

 End If
Next i

End Sub

Sub calcula_xy_objeto()
Dim i, c As Integer
Dim Sx, Sy, ux, uy, max, min As Single
Dim UE, UEx, UEy As Single

msg = "calculo de coordenadas dos objetos"
For c = 1 To colmax

 If FrmParm.Pesocol(c) <> 0 Then

 tabmobj(c, 1) = atrib(1, c)
 tabmobj(c, 2) = atrib(1, c)
 For i = 1 To linmax
  If tabmobj(c, 1) > atrib(i, c) Then
   tabmobj(c, 1) = atrib(i, c)
```

```
  End If
  If tabmobj(c, 2) < atrib(i, c) Then
   tabmobj(c, 2) = atrib(i, c)
  End If
 Next i

 End If

Next c

For i = 1 To linmax
 Sx = 0
 Sy = 0
 For c = 1 To colmax

  If FrmParm.Pesocol(c) <> 0 Then
   min = tabmobj(c, 1)
   max = tabmobj(c, 2)
   If max = min Then '\/correção para max = min
    UE = 0
   Else
    E = c
    UE = E / (max - min)
   End If
   UEx = UE * Cosx(c)
   UEy = UE * Seny(c)
   Sx = UEx * (atrib(i, c) - min) + Sx
   Sy = UEy * (atrib(i, c) - min) + Sy
  End If

 Next c
 tabxyobj(i, 1) = Sx
 tabxyobj(i, 2) = Sy

Next i

End Sub

Sub monta_grafico()
FrmGrafico.Cls
If FrmGrafico.menu_objgraf.Checked = True Then
 plota_objeto
End If
If FrmGrafico.menu_clgraf.Checked = True Then
 plota_cluster
End If

If FrmGrafico.menu_grupocor.Checked = True Then
 exibe_legenda
End If
```

```
If FrmGrafico.menu_idgrupo.Checked = True Then
 exibe_idgrupo
End If

End Sub

Sub plota_objeto()

If FrmGrafico.menu_eixosexb.Checked = True Then
 traca_eixos
End If

For i = 1 To linmax
 x = tabxyobj(i, 1) * escalah
 Y = tabxyobj(i, 2) * escalav

 lintc = tabxyobj(i, 3) - (Int(tabxyobj(i, 3) / 30) * 30) 'MOD

 If (tabxyobj(i, 3) > 30) Then 'ate 30 grupos usa tabela, apos tabela + tons
  tom = Int(tabxyobj(i, 3) / 5)
 Else
  tom = 0
 End If

 R = tc(lintc, 1) + tom
 G = tc(lintc, 2) + tom
 B = tc(lintc, 3) + tom
 For j = 0 To 25  'reta horizontal
  FrmGrafico.PSet (x + j, Y), RGB(R, G, B)
  FrmGrafico.PSet (x - j, Y), RGB(R, G, B)
 Next j
 For j = 0 To 25  'reta vertical
  FrmGrafico.PSet (x, Y + j), RGB(R, G, B)
  FrmGrafico.PSet (x, Y - j), RGB(R, G, B)
 Next j
Next i

End Sub

Sub plota_cluster()

If FrmGrafico.menu_eixosexb.Checked = True Then
 traca_eixos
End If

For i = 1 To nrotimocl
 x = tabxycl(i, 1) * escalah
 Y = tabxycl(i, 2) * escalav
```

```
 lintc = i - (Int(i / 30) * 30) 'MOD

 If (i > 30) Then
  tom = Int(i / 5)
 Else
  tom = 0
 End If

 R = tc(lintc, 1) + tom
 G = tc(lintc, 2) + tom
 B = tc(lintc, 3) + tom
 For j = 25 To 120  'reta horizontal com centro em branco
  For k = 0 To 10
   FrmGrafico.PSet (x + j, Y + k), RGB(R, G, B)
   FrmGrafico.PSet (x - j, Y + k), RGB(R, G, B)
   FrmGrafico.PSet (x + j, Y - k), RGB(R, G, B)
   FrmGrafico.PSet (x - j, Y - k), RGB(R, G, B)
  Next k
 Next j
 For j = 25 To 120  'reta vertical com centro em branco
  For k = 0 To 10
   FrmGrafico.PSet (x + k, Y + j), RGB(R, G, B)
   FrmGrafico.PSet (x - k, Y + j), RGB(R, G, B)
   FrmGrafico.PSet (x + k, Y - j), RGB(R, G, B)
   FrmGrafico.PSet (x - k, Y + j), RGB(R, G, B)
  Next k
 Next j
Next i

End Sub

Sub traca_eixos()
Dim cx, cy, seh, sev, extremoy, extremox As Single
Dim j As Integer

cx = (minx + maxx) / 2
cy = (miny + maxy) / 2
cx = cx * escalah
cy = cy * escalav
seh = cx - minx * escalah
sev = cy - miny * escalav
j = 1

FrmGrafico.ForeColor = &H88&   'preto
FrmGrafico.FillColor = &H88&

For i = 1 To colmax

If FrmParm.Pesocol(i) <> 0 Then
```

```
 'multiplicação por 1.3 para prolongar o eixo em 30%
 extremoy = cx - (seh * 1.3 * Seny(i))
 extremox = cy - (sev * 1.3 * Cosx(i))
 Line (cx, cy)-(extremoy, extremox), QBColor(0)
 FrmGrafico.CurrentX = (cx - (seh * Seny(i))) + 50
 FrmGrafico.CurrentY = (cy - (sev * Cosx(i))) + 50
 While FrmParm.Pesocol(j) = 0
  j = j + 1
 Wend
 If j > 18 Then
  j = j
End If

 FrmGrafico.Print CStr(j)
 j = j + 1

End If

Next i

End Sub

Private Sub Form_KeyPress(KeyAscii As Integer)
If FrmGrafico.menu_objgraf.Checked = True Or FrmGrafico.menu_clgraf.Checked =
True Then
 If KeyAscii = 43 Or KeyAscii = 45 Then
  If KeyAscii = 43 Then
   If cont < 8 Then
    escalah = escalah + escalarh
    escalav = escalav + escalarv
    cont = cont + 1
   End If
  Else
   If cont > 0 Then
    escalah = escalah - escalarh
    escalav = escalav - escalarv
    cont = cont - 1
   End If
  End If
 End If
 monta_grafico
Else
 If FrmGrafico.menu_cldendro.Checked = True Then
  monta_dendrograma "cl"
 Else
  monta_dendrograma "obj"
 End If
End If

End Sub
```

```
Private Sub Form_MouseDown(Button As Integer, Shift As Integer, x As Single, Y As
Single)
If FrmGrafico.menu_objgraf.Checked = True Or FrmGrafico.menu_clgraf.Checked =
True Then
 If Button = 2 Then
  If cont < 8 Then
   escalah = escalah + (escalah / 4)
   escalav = escalav + (escalav / 4)
   cont = cont + 1
  End If
 Else
  If Button = 1 Then
   If cont > 0 Then
    escalah = escalah - (escalah / 4)
    escalav = escalav - (escalav / 4)
    cont = cont - 1
   End If
  End If
 End If
 monta_grafico
Else
 If FrmGrafico.menu_cldendro.Checked = True Then
  monta_dendrograma "cl"
 Else
  monta_dendrograma "obj"
 End If
End If

End Sub

Private Sub Form_paint()

If FrmGrafico.menu_objgraf.Checked = True Or FrmGrafico.menu_clgraf.Checked =
True Then
 monta_grafico
Else
 If FrmGrafico.menu_cldendro.Checked = True Then
  monta_dendrograma "cl"
 Else
  If FrmGrafico.menu_cldendro.Checked = True Then
   monta_dendrograma "obj"
  End If
 End If
End If

End Sub

Sub calcula_xy_cluster()
```

```
Dim i, c As Integer

msg = "calculo de coordenadas do centro dos grupos"
Erase tabxycl
ReDim tabxycl(nrotimocl, 2)

For i = 1 To linmax
 c = tabxyobj(i, 3)
 tabxycl(c, 1) = tabxycl(c, 1) + tabxyobj(i, 1)
 tabxycl(c, 2) = tabxycl(c, 2) + tabxyobj(i, 2)
Next i

For c = 1 To nrotimocl
 tabxycl(c, 1) = (tabxycl(c, 1) / tabresumo(c, 31))
 tabxycl(c, 2) = (tabxycl(c, 2) / tabresumo(c, 31))
Next c

End Sub

Sub transforma_max_min_obj()
Dim i, c As Integer

msg = "transformação de máximos e mínimos dos objetos"

maxx = tabxyobj(1, 1)
minx = tabxyobj(1, 1)
maxy = tabxyobj(1, 2)
miny = tabxyobj(1, 2)

For i = 1 To linmax
 If tabxyobj(i, 1) > maxx Then
  maxx = tabxyobj(i, 1)
 Else
  If tabxyobj(i, 1) < minx Then
   minx = tabxyobj(i, 1)
  End If
 End If
 If tabxyobj(i, 2) > maxy Then
  maxy = tabxyobj(i, 2)
 Else
  If tabxyobj(i, 2) < miny Then
   miny = tabxyobj(i, 2)
  End If
 End If
Next i

desx = 0
desy = 0

If minx <= 0 Then
```

```
 desx = (minx * -1) + 1
End If
If miny <= 0 Then
 desy = (miny * -1) + 1
End If
maxx = maxx + desx
maxy = maxy + desy
minx = minx + desx
miny = miny + desy
For i = 1 To linmax
 tabxyobj(i, 1) = tabxyobj(i, 1) + desx
 tabxyobj(i, 2) = tabxyobj(i, 2) + desy
Next i

End Sub

Sub calcula_distancia_objeto()
On Error GoTo ChamaRotinaErro

Dim i, j, c As Integer
Dim dif, dist As Single
ReDim tabdobj(linmax, linmax)

msg = "cálculo da distância entre os objetos"

For i = 1 To linmax
 For j = 1 To linmax
  dist = 0
   For c = 1 To colmax
    If FrmParm.Pesocol(c) <> 0 Then
     dif = atrib(i, c) - atrib(j, c)
     dist = dist + (dif * dif)
    End If
   Next c
   tabdobj(i, j) = Sqr(dist)
   If tabdobj(i, j) = 0 Then
    If i <> j Then
     i = i
    End If
   End If
 Next j
Next i

Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub executa_agregação_objeto()
```

```
Dim i, j, L, c As Integer
Dim vpeso() As Integer
ReDim tabagrobj(linmax, 5), vpeso(2, linmax)

For i = 1 To linmax
 vpeso(1, i) = i
 vpeso(2, i) = 1
Next i

L = 0
ordem = linmax
tabagrobj(1, 1) = linmax
While ordem > 1
 L = L + 1
 tabagrobj(L, 1) = linmax + L          'nó
 tabagrobj(L, 5) = distmin("obj")      'nível
 tabagrobj(L, 2) = vpeso(1, auxi)     '1.o pto
 tabagrobj(L, 3) = vpeso(1, auxj)     '2.o pto
 vpeso(1, auxi) = tabagrobj(L, 1)
 vpeso(2, auxi) = vpeso(2, auxi) + vpeso(2, auxj)
 tabagrobj(L, 4) = vpeso(2, auxi)     'peso
                                '\/ reorganiza a matriz
 For j = 1 To ordem
  If tabdobj(auxi, j) > tabdobj(auxj, j) Then
   tabdobj(auxi, j) = tabdobj(auxj, j)
  End If
  If tabdobj(j, auxi) > tabdobj(j, auxj) Then
   tabdobj(j, auxi) = tabdobj(j, auxj)
  End If
 Next j

 If auxj < ordem Then
 For i = auxj + 1 To ordem
  vpeso(1, i - 1) = vpeso(1, i)
  vpeso(2, i - 1) = vpeso(2, i)
  For j = 1 To ordem
   tabdobj(i - 1, j) = tabdobj(i, j)
   If i = ordem Then
    tabdobj(i, j) = 0
   End If
  Next j
 Next i

 For i = 1 To ordem
  For j = auxj + 1 To ordem
   tabdobj(i, j - 1) = tabdobj(i, j)
   If j = ordem Then
    tabdobj(i, j) = 0
   End If
  Next j
```

```
 Next i
 End If

 ordem = ordem - 1
                                    '\
Wend
objagr = True

End Sub

Sub calcula_distancia_cluster()
On Error GoTo ChamaRotinaErro

Dim i, j, c As Integer
Dim dif, dist As Single
ReDim tabdcl(nrotimocl, nrotimocl)

msg = "cálculo da distância entre os clusters"
nmrcl = nrotimocl

For i = 1 To nmrcl
 For j = 1 To nmrcl
  dist = 0
   For c = 1 To colmax
    If FrmParm.Pesocol(c) <> 0 Then
     dif = tabresumo(i, c) - tabresumo(j, c)
     dist = dist + (dif * dif)
    End If
   Next c
   tabdcl(i, j) = Sqr(dist)
 Next j
Next i

Exit Sub

ChamaRotinaErro: RotinaErro

End Sub

Sub executa_agregação_cluster()
Dim i, j, L, c As Integer
Dim vpeso() As Integer
ReDim tabagrcl(nmrcl, 5), vpeso(2, nmrcl)

For i = 1 To nmrcl
 vpeso(1, i) = i
 vpeso(2, i) = 1
Next i

L = 0
```

```
ordem = nmrcl
tabagrcl(1, 1) = nmrcl
While ordem > 1
 L = L + 1
 tabagrcl(L, 1) = nmrcl + L          'nó
 tabagrcl(L, 5) = distmin("cl")      'nível
 tabagrcl(L, 2) = vpeso(1, auxi)     '1.o pto
 tabagrcl(L, 3) = vpeso(1, auxj)     '2.o pto
 vpeso(1, auxi) = tabagrcl(L, 1)
 vpeso(2, auxi) = vpeso(2, auxi) + vpeso(2, auxj)
 tabagrcl(L, 4) = vpeso(2, auxi)     'peso
                                     '\/ reorganiza a matriz
 For j = 1 To ordem
  If tabdcl(auxi, j) > tabdcl(auxj, j) Then
   tabdcl(auxi, j) = tabdcl(auxj, j)
  End If
  If tabdcl(j, auxi) > tabdcl(j, auxj) Then
   tabdcl(j, auxi) = tabdcl(j, auxj)
  End If
 Next j

 If auxj < ordem Then
 For i = auxj + 1 To ordem
  vpeso(1, i - 1) = vpeso(1, i)
  vpeso(2, i - 1) = vpeso(2, i)
  For j = 1 To ordem
   tabdcl(i - 1, j) = tabdcl(i, j)
   If i = ordem Then
    tabdcl(i, j) = 0
   End If
  Next j
 Next i

 For i = 1 To ordem
  For j = auxj + 1 To ordem
   tabdcl(i, j - 1) = tabdcl(i, j)
   If j = ordem Then
    tabdcl(i, j) = 0
   End If
  Next j
 Next i
 End If

 ordem = ordem - 1
                                     '/\
Wend
objagrcl = True

End Sub
```

```
Function distmin(tpdist As String) As Single

auxi = 1
auxj = 2
For i = 1 To ordem - 1  'só busca acima da diagonal principal
 For j = i + 1 To ordem
  If (tpdist = "obj") Then
   If tabdobj(i, j) < tabdobj(auxi, auxj) Then
    auxi = i
    auxj = j
   End If
  Else
   If tabdcl(i, j) < tabdcl(auxi, auxj) Then
    auxi = i
    auxj = j
   End If
  End If
 Next j
Next i
If (tpdist = "obj") Then
 distmin = tabdobj(auxi, auxj)
Else
 distmin = tabdcl(auxi, auxj)
End If

End Function

Sub monta_dendrograma(tpdendro As String)

margem = 10

If tpdendro = "obj" Then
 calcula_centro_compostas_obj
 traça_reta_vertical_simples_obj
 traça_reta_vertical_agregada_obj
 traça_reta_horizontal_obj
Else
 calcula_centro_compostas
 traça_reta_vertical_simples
 R = 0
 G = 0
 B = 0
 traça_reta_vertical_agregada
 traça_reta_horizontal
End If
If FrmGrafico.menu_grupocor.Checked = True Then
 exibe_legenda
End If

End Sub
```

```
Sub calcula_centro_compostas_obj()
Dim L, j As Integer
ReDim tabcobj(linmax)

For L = 1 To linmax - 1
  If tabagrobj(L, 2) <= linmax Then
  xi = tabagrobj(L, 2)
 Else
  j = tabagrobj(L, 2) - linmax
  xi = tabcobj(j)
 End If
 If tabagrobj(L, 3) <= linmax Then
  xf = tabagrobj(L, 3)
 Else
  j = tabagrobj(L, 3) - linmax
  xf = tabcobj(j)
 End If
 tabcobj(L) = (xi + xf) / 2
Next L

End Sub

Sub traça_reta_vertical_simples_obj()
Dim L, j, obj As Integer

For L = 1 To linmax - 1
 yf = tabagrobj(L, 5) * escaladv + margem
 If tabagrobj(L, 2) <= linmax Then
  yi = margem
  x = escaladh * tabagrobj(L, 2)
  obj = tabagrobj(L, 2)
  lintc = tabxyobj(obj, 3) - (Int(tabxyobj(obj, 3) / 30) * 30) 'MOD
  R = tc(lintc, 1)
  G = tc(lintc, 2)
  B = tc(lintc, 3)
  ligapixel "v", 0
 End If
 If tabagrobj(L, 3) <= linmax Then
  yi = margem
  x = escaladh * tabagrobj(L, 3)
  obj = tabagrobj(L, 3)
  lintc = tabxyobj(obj, 3) - (Int(tabxyobj(obj, 3) / 30) * 30) 'MOD
  R = tc(lintc, 1)
  G = tc(lintc, 2)
  B = tc(lintc, 3)
  ligapixel "v", 0
 End If
Next L
```

```
End Sub

Sub traça_reta_vertical_agregada_obj()
Dim L, j As Integer

R = 0
G = 0
B = 0
For L = 1 To linmax - 1
 yf = tabagrobj(L, 5) * escaladv + margem
 If tabagrobj(L, 2) > linmax Then
  j = tabagrobj(L, 2) - linmax
  yi = tabagrobj(j, 5) * escaladv + margem
  x = escaladh * tabcobj(j)
  ligapixel "v", 0
 End If
 If tabagrobj(L, 3) > linmax Then
  j = tabagrobj(L, 3) - linmax
  yi = tabagrobj(j, 5) * escaladv + margem
  x = escaladh * tabcobj(j)
  ligapixel "v", 0
 End If
Next L
End Sub

Sub traça_reta_horizontal_obj()
Dim L, j As Integer

For L = 1 To linmax - 1
 Y = tabagrobj(L, 5) * escaladv + margem
 If tabagrobj(L, 2) <= linmax Then
  xi = escaladh * tabagrobj(L, 2)
 Else
  j = tabagrobj(L, 2) - linmax
  xi = escaladh * tabcobj(j)
 End If
 If tabagrobj(L, 3) <= linmax Then
  xf = escaladh * tabagrobj(L, 3)
 Else
  j = tabagrobj(L, 3) - linmax
  xf = escaladh * tabcobj(j)
 End If
 ligapixel "hobj", CInt(tabagrobj(L, 1))
Next L

End Sub

Sub calcula_centro_compostas()
Dim L, j As Integer
ReDim tabc(nrotimocl)
```

```
For L = 1 To nmrcl - 1
 If tabagrcl(L, 2) <= nmrcl Then
  xi = tabagrcl(L, 2)
 Else
  j = tabagrcl(L, 2) - nmrcl
  xi = tabc(j)
 End If
 If tabagrcl(L, 3) <= nmrcl Then
  xf = tabagrcl(L, 3)
 Else
  j = tabagrcl(L, 3) - nmrcl
  xf = tabc(j)
 End If
 tabc(L) = (xi + xf) / 2
Next L

End Sub

Sub traça_reta_vertical_simples()
Dim L, j, obj As Integer

For L = 1 To nmrcl - 1
 yf = tabagrcl(L, 5) * escaladv + margem
 If tabagrcl(L, 2) <= nmrcl Then
  yi = margem
  x = escaladh * tabagrcl(L, 2)
  lintc = tabagrcl(L, 2) - (Int(tabagrcl(L, 2) / 30) * 30) 'MOD
  R = tc(lintc, 1)
  G = tc(lintc, 2)
  B = tc(lintc, 3)

  ligapixel "v", 0
 End If
 If tabagrcl(L, 3) <= nmrcl Then
  yi = margem
  x = escaladh * tabagrcl(L, 3)
  lintc = tabagrcl(L, 2) - (Int(tabagrcl(L, 2) / 30) * 30) 'MOD
  R = tc(lintc, 1)
  G = tc(lintc, 2)
  B = tc(lintc, 3)

  ligapixel "v", 0
 End If
Next L

End Sub

Sub traça_reta_vertical_agregada()
Dim L, j As Integer
```

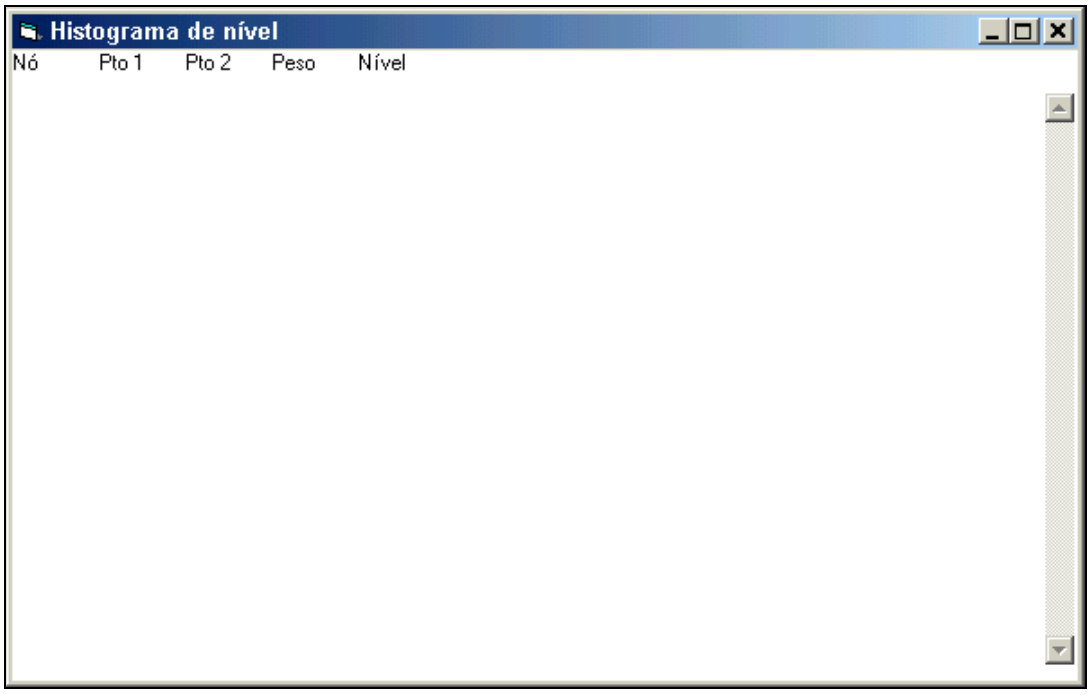
```
For L = 1 To nmrcl - 1
 yf = tabagrcl(L, 5) * escaladv + margem
 If tabagrcl(L, 2) > nmrcl Then
  j = tabagrcl(L, 2) - nmrcl
  yi = tabagrcl(j, 5) * escaladv + margem
  x = escaladh * tabc(j)
  ligapixel "v", 0
 End If
 If tabagrcl(L, 3) > nmrcl Then
  j = tabagrcl(L, 3) - nmrcl
  yi = tabagrcl(j, 5) * escaladv + margem
  x = escaladh * tabc(j)
  ligapixel "v", 0
 End If
Next L
End Sub

Sub traça_reta_horizontal()
Dim L, j As Integer

For L = 1 To nmrcl - 1
 Y = tabagrcl(L, 5) * escaladv + margem
 If tabagrcl(L, 2) <= nmrcl Then
  xi = escaladh * tabagrcl(L, 2)
 Else
  j = tabagrcl(L, 2) - nmrcl
  xi = escaladh * tabc(j)
 End If
 If tabagrcl(L, 3) <= nmrcl Then
  xf = escaladh * tabagrcl(L, 3)
 Else
  j = tabagrcl(L, 3) - nmrcl
  xf = escaladh * tabc(j)
 End If
 ligapixel "hcl", CInt(tabagrcl(L, 1))
Next L

End Sub

Private Sub ligapixel(sentido As String, nó As Integer)
If sentido = "v" Then
 If yi > yf Then
  aux = yi
  yi = yf
  yf = aux
 End If
 Line (x, 8600 - yi)-(x, 8600 - yf), RGB(R, G, B)
Else
 If xi > xf Then
```

```
 aux = xi
 xi = xf
 xf = aux
End If
Line (xi, 8600 - Y)-(xf, 8600 - Y), RGB(R, G, B)
FrmGrafico.CurrentX = CInt((xi + xf) / 2) - 200
FrmGrafico.CurrentY = 8620 - Y
If sentido = "hcl" Then
 FrmGrafico.Print CStr(nó)
End If
End If

End Sub

Sub exibe_caracteres()

Dim x, Y, i, j As Integer

x = 4200
Y = 150

FrmGrafico.ForeColor = &HFF&   'vermelho
FrmGrafico.FillColor = &HFF&

FrmGrafico.CurrentX = x
FrmGrafico.CurrentY = Y
FrmGrafico.Print "Nó   Pto 1    Pto 2    Peso    Nível"

i = 1
imnivel = 1
While tabagrcl(i, 1) <> Empty
 x = 4200
 Y = Y + 200
 FrmGrafico.CurrentX = x
 FrmGrafico.CurrentY = Y
 FrmGrafico.Print CStr(tabagrcl(i, 1))
 x = x + 360
 For j = 3 To 5
  FrmGrafico.CurrentX = x
  FrmGrafico.CurrentY = Y
  FrmGrafico.Print CStr(tabagrcl(i, j))
  x = x + 550
 Next j
 FrmGrafico.CurrentX = x
 FrmGrafico.CurrentY = Y
 FrmGrafico.Print CStr(tabagrcl(i, 2))
 If tabagrcl(imnivel, 5) < tabagrcl(i, 5) Then
  imnivel = i
 End If
 i = i + 1
```

```
Wend

End Sub

Sub exibe_barras()
Dim i, j, x, Y, xf, yi, yf As Integer
Dim escala As Single

escala = 4880 / tabagrcl(imnivel, 5)
i = 1
yi = 394
While tabagrcl(i, 1) <> Empty
 xf = 6960 + (tabagrcl(i, 5) * escala)
 Line (6960, yi)-(xf, yi + 150),, B
 i = i + 1
 yi = yi + 194
Wend
FrmGrafico.ForeColor = &HC0C0C0   'cinza
FrmGrafico.FillColor = &HC0C0C0
Line (4080, 1)-(4100, yi + 60),, B
Line (4100, yi + 50)-(12000, yi + 60),, B

FrmGrafico.ForeColor = &HFFFFFF   'branco

End Sub

Sub exibe_legenda()

Dim x, Y, i, j, maxleg As Integer

x = 80
Y = 15

tam = FrmGrafico.Width
x = tam - 940

FrmGrafico.CurrentX = x
FrmGrafico.CurrentY = Y
FrmGrafico.Print "Grupo  Cor"
x = x + 170
 maxleg = nrotimocl

For i = 1 To maxleg
 Y = Y + 210
 FrmGrafico.CurrentX = x
 FrmGrafico.CurrentY = Y
 FrmGrafico.Print CStr(i)
Next i

x = x + 380
```

```
yi = 220

For i = 1 To maxleg

 R = tc(i, 1)
 G = tc(i, 2)
 B = tc(i, 3)
 FrmGrafico.Line (x, yi)-(x + 200, yi + 150), RGB(R, G, B), BF
 yi = yi + 210

Next i

End Sub

Sub exibe_idgrupo()

FrmGrafico.Font.Name = "arial"
FrmGrafico.Font.Size = 7

For i = 1 To nrotimocl
 FrmGrafico.CurrentX = tabxycl(i, 1) * escalah + 118
 FrmGrafico.CurrentY = tabxycl(i, 2) * escalav + 12
 FrmGrafico.Print CStr(i)
Next i
FrmGrafico.Font.Name = "ms sans serif"
FrmGrafico.Font.Size = 8

End Sub
```

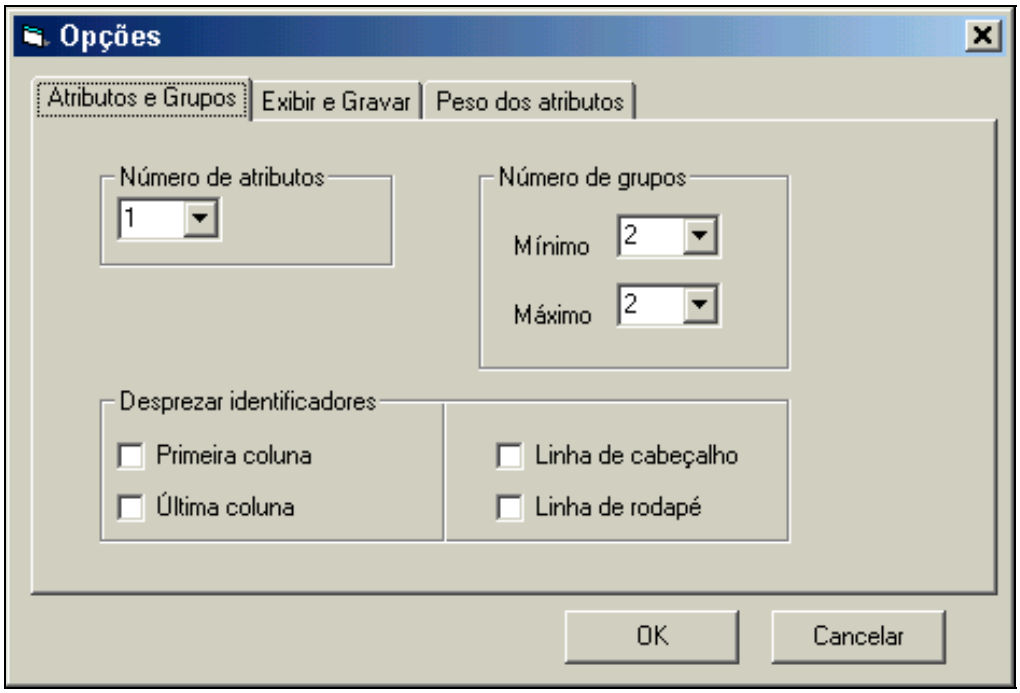
FORMULÁRIO FRMHISTO



```
Option Explicit
Dim imnivel As Integer

Private Sub Form_Load()
Dim x As Currency
Dim i, j, L, escala, max As Integer
Dim barra() As String

max = 84
ReDim barra(max + 1)
FrmHisto.titcol = "Nó" + vbTab + "Pto 1" + vbTab + "Pto 2" + vbTab
FrmHisto.titcol = FrmHisto.titcol + "Peso" + vbTab + "Nível"

For i = 1 To max + 1
 For j = 1 To i
  barra(i) = barra(i) + "_"
 Next j
Next i

i = 1  'determina o maior nível e a escala do histograma
imnivel = 1
If tipo_histo = "obj" Then
 While tabagrobj(i, 1) <> Empty
  If tabagrobj(imnivel, 5) < tabagrobj(i, 5) Then
   imnivel = i
  End If
  i = i + 1
 Wend
 escala = max / tabagrobj(imnivel, 5)
 Else
 While tabagrcl(i, 1) <> Empty
  If tabagrcl(imnivel, 5) < tabagrcl(i, 5) Then
   imnivel = i
  End If
  i = i + 1
 Wend
 escala = max / tabagrcl(imnivel, 5)
End If

i = 1
If tipo_histo = "obj" Then
While tabagrobj(i, 1) <> Empty
 For j = 1 To 5
  x = tabagrobj(i, j)
  FrmHisto.colhisto = FrmHisto.colhisto + CStr(x) + vbTab
 Next j
 L = Int((escala * tabagrobj(i, 5))) + 1
 FrmHisto.colhisto = FrmHisto.colhisto + barra(L) + vbCrLf
 i = i + 1
```

```
Wend
Else
While tabagrcl(i, 1) <> Empty
 For j = 1 To 5
  x = tabagrcl(i, j)
  FrmHisto.colhisto = FrmHisto.colhisto + CStr(x) + vbTab
 Next j
 L = Int((escala * tabagrcl(i, 5))) + 1
 FrmHisto.colhisto = FrmHisto.colhisto + barra(L) + vbCrLf
 i = i + 1
Wend
End If

End Sub

Private Sub Form_Resize()
 FrmGrafico.Refresh
 If FrmGrafico.menu_cldendro.Checked = True Then
  FrmGrafico.monta_dendrograma "cl"
 Else
  FrmGrafico.monta_dendrograma "obj"
 End If
End Sub

Private Sub Form_Unload(Cancel As Integer)
 FrmGrafico.menu_objhist.Checked = False
 FrmGrafico.menu_clhist.Checked = False
End Sub
```

FORMULÁRIO FRMPARM



```
Option Explicit
Dim vparm(10) As String
Dim vparmcol(30) As Integer
Dim x, pesoant, guiaant, indant As Integer
Dim errocol As Boolean
Dim msg As String

Private Sub Form_Activate()

Dim i As Integer
 errocol = False
 guiaant = 1
 indant = 1
 msg = "O peso deve estar entre 0 e 99999"

 vparm(1) = FrmParm.nmratrib
 vparm(2) = FrmParm.nmrmincluster
 vparm(3) = FrmParm.nmrmaxcluster
 vparm(4) = FrmParm.chkpcol.Value
 vparm(5) = FrmParm.chkucol.Value
 vparm(6) = FrmParm.chkplin.Value
 vparm(7) = FrmParm.chkulin.Value
 vparm(9) = FrmParm.chkerros.Value
 vparm(10) = FrmParm.chkarquivos.Value

 seta_despcol

 For i = 1 To vparm(1)
  vparmcol(i) = Pesocol(i)
 Next i

End Sub

Private Sub Guia_Click()
Dim i, j As Integer

If guiaant = 3 And errocol = True Then
 errocol = False
 MsgBox msg, vbCritical, "*** ERRO ***"
 Guia.Tabs(3).Selected = True
Else
 For i = 1 To Guia.Tabs.Count
  If i = Guia.SelectedItem.Index Then
   Picparm(i).Visible = True
   If i = 3 Then
    seta_despcol
    If Pesocol(indant) = Empty Then
     errocol = True
    End If
```

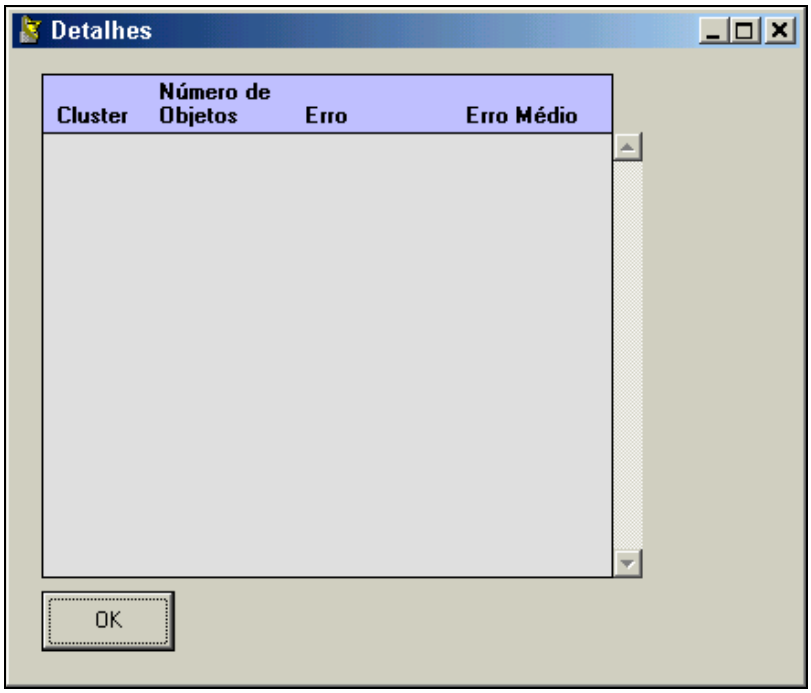
```
   End If
  Else
   Picparm(i).Visible = False
  End If
 Next i
 guiaant = Guia.SelectedItem.Index
End If

End Sub

Private Sub Pesocol_Change(Index As Integer)
 errocol = False
 If Pesocol(Index) = Empty Then
  errocol = True
 Else
  critica_pesocol (Index)
 End If

End Sub

Private Sub Pesocol_GotFocus(Index As Integer)
pesoant = Pesocol(Index)
indant = Index
End Sub

Private Sub Pesocol_LostFocus(Index As Integer)
 errocol = False
 If FrmParm.Cancelar = False Then
   critica_pesocol (Index)
 End If
End Sub

Private Sub OK_Click()
Dim msg1, msg2 As String

 If FrmParm.chkarquivos = 1 Then
  FrmCluster.menu_arqsaida_obj.Checked = True
 End If

 FrmCluster.Enabled = True
 If CInt(FrmParm.nmrmaxcluster) < CInt(FrmParm.nmrmincluster) Then
  FrmParm.nmrmincluster.SetFocus
 Else
  If CInt(FrmParm.nmrmincluster) < 2 Then
   FrmParm.nmrmincluster = 2
   FrmParm.nmrmincluster.SetFocus
  Else
   If CInt(FrmParm.nmrmaxcluster) > 200 Then
    FrmParm.nmrmaxcluster = 200
    FrmParm.nmrmaxcluster.SetFocus
```

```
   Else
    FrmParm.Hide
    carrega_msg
    monta_results1
   End If
  End If
 End If

End Sub

Private Sub Cancelar_Click()
Dim i As Integer

FrmCluster.Enabled = True
FrmParm.nmratrib = vparm(1)
FrmParm.nmrmincluster = vparm(2)
FrmParm.nmrmaxcluster = vparm(3)
FrmParm.chkpcol.Value = vparm(4)
FrmParm.chkucol.Value = vparm(5)
FrmParm.chkplin.Value = vparm(6)
FrmParm.chkulin.Value = vparm(7)
FrmParm.chkerros.Value = vparm(9)
FrmParm.chkarquivos.Value = vparm(10)

For i = 11 To vparm(1)
 Pesocol(i) = vparmcol(i)
Next i

If st(4) <> Empty Then
 carrega_msg
 monta_results1
End If

FrmParm.Hide

End Sub

Private Sub seta_despcol()
Dim i As Integer

 For i = 1 To 30
  If i <= FrmParm.nmratrib Then
   FrmParm.Pesocol(i).Visible = True
   FrmParm.Lbpcol(i).Visible = True
  Else
   FrmParm.Pesocol(i).Visible = False
   FrmParm.Pesocol(i) = 1
   FrmParm.Lbpcol(i).Visible = False
  End If
 Next i
```

```
End Sub

Private Sub critica_pesocol(Index As Integer)
Dim j As Integer

If Not IsNumeric(Pesocol(Index)) Then
 errocol = True
Else
 If Pesocol(Index) < 0 Or Pesocol(Index) > 99999 Then
  errocol = True
 End If
End If
If errocol = True Then
 Pesocol(Index).SetFocus
 MsgBox msg, vbCritical, "*** ERRO ***"
 Pesocol(Index) = pesoant
End If

End Sub

Private Sub carrega_msg()
Dim i, j, nenhum As Integer
Dim s1 As String

 i = 4
 s1 = FrmParm.nmratrib
 st(i) = "Número de atributos: " + s1
 s1 = FrmParm.nmrmincluster
 i = i + 1
 st(i) = "Faixa de grupos a testar: de " + s1 + " a "
 s1 = FrmParm.nmrmaxcluster
 st(i) = st(i) + s1
 i = i + 1
 st(i) = " coluna identificadora inicial "
 If FrmParm.chkpcol = 1 Then
  st(i) = "Possui" + st(i) + "(desprezada)"
 Else
  st(i) = "Não possui" + st(i)
 End If
 i = i + 1
 st(i) = " coluna identificadora final "
 If FrmParm.chkucol = 1 Then
  st(i) = "Possui" + st(i) + "(desprezada)"
 Else
  st(i) = "Não possui" + st(i)
 End If
 i = i + 1
 st(i) = " linha de cabeçalho "
 If FrmParm.chkplin = 1 Then
```

```
 st(i) = "Possui" + st(i) + "(desprezada)"
Else
 st(i) = "Não possui" + st(i)
End If
i = i + 1
st(i) = " linha de rodapé "
If FrmParm.chkulin = 1 Then
 st(i) = "Possui" + st(i) + "(desprezada)"
Else
 st(i) = "Não possui" + st(i)
End If
If FrmParm.chkarquivos = 1 Then
 i = i + 1
 st(i) = "Gravar arquivo de objetos para cada cluster testado"
End If

nenhum = 0
contcol0 = 0
i = i + 1
st(i) = "Atributos a desconsiderar:"
s1 = " "
For j = 1 To FrmParm.nmratrib
 If FrmParm.Pesocol(j) = 0 Then
  st(i) = st(i) + s1 + FrmParm.Lbpcol(j).Caption
  s1 = ", "
  nenhum = 1
  contcol0 = contcol0 + 1
 End If
Next j
If nenhum = 0 Then
 st(i) = "Considerar todos os atributos"
End If

End Sub

Private Sub Form_deactivate()

If st(1) <> Empty And st(4) <> Empty Then
 FrmCluster.Agrupar.Enabled = True
 FrmCluster.Visualização.Enabled = True
End If

FrmParm.Guia.Tabs(1).Selected = True

End Sub
```

FORMULÁRIO FRMRESUMO




```
Private Sub OK_Click()
FrmResumo.Hide

End Sub
```

MODULES MODULO1

```
Public tabxycl() As Single 'coluna 1: coord X do centro do cluster
                           'coluna 2: coord Y do centro do cluster

Public atrib() As Single  'coluna 31: distancia do objeto ao centro do cluster
                          'coluna 32: número do cluster

Public tabxyobj() As Single 'coluna 1: coord X do objeto
                            'coluna 2: coord Y do objeto
                            'coluna 3: numero do cluster

Public tabcl(200, 32) As Single 'coluna 31: qtd obj do cluster
                                'coluna 32: somatorio dos erros dos obj do cluster

Public tabresumo(200, 32) As Single  'coluna 31: qtd obj
                                     'coluna 32: erro do cluster

Public tabdobj() As Single 'tabela de distancias entre objetos

Public tabdcl() As Single 'tabela de distancias entre clusters
```

```
Public tabagrcl() As Single 'tabela resumo da agregação
        'col 1: nmr nó   col 2: nível   col 3: primeiro pto   col 4: segundo pto   col 5: peso
Public tabagrobj() As Single

Public linmax, maxx, minx, maxy, miny As Single

Public nrotimocl, colmax, contcol0 As Integer

Public st(216) As String 'vetor de string para msg resumo

Public Agrupamentos_ok As Boolean

Public tipo_histo As String

Public tc(31, 3) As Integer 'tabela de cores R G B

Public Sub monta_results1()
Dim i As Integer

FrmCluster.results1 = Empty
For i = 1 To 215
 If st(i) <> Empty Then
  FrmCluster.results1 = FrmCluster.results1 + st(i) + vbCrLf
 End If
Next i

End Sub

Sub monta_tabela_cores()
Erase tc
tc(1, 1) = 256: tc(2, 2) = 256: tc(3, 3) = 256: tc(4, 1) = 256: tc(4, 2) = 256
tc(5, 1) = 256: tc(5, 3) = 256: tc(6, 2) = 256: tc(6, 3) = 64: tc(7, 1) = 128
tc(7, 2) = 192: tc(7, 3) = 192: tc(8, 1) = 192: tc(8, 2) = 128: tc(8, 3) = 192
tc(9, 1) = 192: tc(9, 2) = 192: tc(9, 3) = 128: tc(10, 1) = 128: tc(10, 2) = 128
tc(10, 3) = 192: tc(11, 1) = 128: tc(11, 2) = 192: tc(11, 3) = 128: tc(12, 1) = 192
tc(12, 2) = 128: tc(12, 3) = 128: tc(13, 1) = 256: tc(13, 2) = 192: tc(13, 3) = 192
tc(14, 1) = 64: tc(14, 2) = 192: tc(14, 3) = 54: tc(15, 1) = 192: tc(15, 2) = 192
tc(15, 3) = 256: tc(16, 1) = 192: tc(16, 2) = 160: tc(17, 1) = 256: tc(17, 2) = 192
tc(17, 3) = 256: tc(18, 1) = 160: tc(18, 2) = 160: tc(18, 3) = 192: tc(19, 1) = 192
tc(20, 1) = 64: tc(20, 2) = 160: tc(20, 3) = 64: tc(21, 3) = 192: tc(22, 1) = 192
tc(22, 2) = 192: tc(23, 1) = 192: tc(23, 3) = 192: tc(24, 2) = 192: tc(24, 3) = 192
tc(25, 2) = 128: tc(25, 3) = 128: tc(26, 1) = 128: tc(26, 2) = 96: tc(26, 3) = 128
tc(27, 1) = 128: tc(27, 2) = 192: tc(28, 1) = 96: tc(28, 2) = 96: tc(28, 3) = 128
tc(29, 2) = 128: tc(30, 1) = 160: tc(30, 2) = 160: tc(30, 3) = 160

End Sub
```

Livros Grátis

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

Milhares de Livros para Download:

Baixar livros de Administração
Baixar livros de Agronomia
Baixar livros de Arquitetura
Baixar livros de Artes
Baixar livros de Astronomia
Baixar livros de Biologia Geral
Baixar livros de Ciência da Computação
Baixar livros de Ciência da Informação
Baixar livros de Ciência Política
Baixar livros de Ciências da Saúde
Baixar livros de Comunicação
Baixar livros do Conselho Nacional de Educação - CNE
Baixar livros de Defesa civil
Baixar livros de Direito
Baixar livros de Direitos humanos
Baixar livros de Economia
Baixar livros de Economia Doméstica
Baixar livros de Educação
Baixar livros de Educação - Trânsito
Baixar livros de Educação Física
Baixar livros de Engenharia Aeroespacial
Baixar livros de Farmácia
Baixar livros de Filosofia
Baixar livros de Física
Baixar livros de Geociências
Baixar livros de Geografia
Baixar livros de História
Baixar livros de Línguas

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