UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
INSTITUTO DE MATEMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

LEÔNCIO TEIXEIRA CRUZ

Utilizando uma Nova Abordagem para Análise de Cluster com Vistas a uma Gestão de Riscos mais Efetiva em Projetos de Software

Prof. Dr. Eber Assis Schmitz
Orientador

Prof. Dr. Antônio Juarez Alencar
Co-orientador

Rio de Janeiro, Março de 2009

Ficha Catalográfica

Teixeira Cruz, Leôncio

Utilizando uma Nova Abordagem para Análise de Cluster com Vistas a uma Gestão de Riscos mais Efetiva em Projetos de Software / Leôncio Teixeira Cruz. – Rio de Janeiro: UFRJ IM, 2009.

101 f.: il.

Dissertação (Mestrado em Informática) – Universidade Federal do Rio de Janeiro. Programa de Pós-Graduação em Informática, Rio de Janeiro, BR–RJ, 2009.

Orientador: Eber Assis Schmitz; Co-orientador: Antônio Juarez Alencar.

I. Assis Schmitz, Eber. II. Juarez Alencar, Antônio. III. Título.

Utilizando uma Nova Abordagem para Análise de Cluster com Vistas a uma Gestão de Riscos mais Efetiva em Projetos de Software

Leôncio Teixeira Cruz

Dissertação de Mestrado submetida ao Corpo Docente do Departamento de Ciência da Computação do Instituto de Matemática, e Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro, como parte dos requisitos necessários para obtenção do título de Mestre em Informática.

Aprovado por:

_____

Prof. Dr. Eber Assis Schmitz (Orientador)

_____

Prof. Dr. Antônio Juarez Alencar (Co-orientador)

_____

Profa. Dra. Maria Luiza Machado Campos

_____

Prof. Dr. Sérgio José Mecena da Silva Filho

Rio de Janeiro, Março de 2009

Dedico este trabalho aos meus pais, pelo amor, pelo carinho, pela dedicação, enfim por tudo que me faz ter orgulho de ser seu filho.

AGRADECIMENTOS

Em primeiro lugar a Deus. Aos meus pais pela oportunidade e apoio que me deram em todos os momentos da minha vida. Aos meus professores, em especial ao meu professor orientador Antônio Juarez Alencar pela sua dedicação e atenção. Agradeço também a todos os funcionários da secretaria de informática com os quais sempre pude contar quando precisei. Agradeço ao professor Eber Assis pelo seu apoio, ensinamentos e coorientação do meu trabalho. Agradeço aos coordenadores do curso Maria Luiza e Marcos Borges pelo esforço em fornecer uma bolsa de auxílio. Aos amigos, familiares, colegas de trabalho que sempre confiaram e torcem e acreditam em mim.

RESUMO

Este trabalho apresenta um método que oferece às organizações e seus gerentes um melhor entendimento das implicações dos riscos que seus projetos de softwares estão submetidos. O método, que é baseado em uma nova abordagem para análise de cluster, fornece um conjunto sentenças lógicas que indicam como fatores de risco podem influenciar a performance de projetos de software. Estas sentenças são, na verdade, regras de negócio que podem ser facilmente entendidas e interpretadas. O que possibilita que os gerentes de projeto possam identificar previamente que ações devem ser tomadas e quando e onde devem ser executadas, para evitar que o pior aconteça.

Palavras-chave: Análise de Cluster, Data Mining, Gerência de Projetos, Análise de Risco.

Using a Novel Approach to Cluster Analysis to Gain New Valuable Insights into Software-Project Risk Management

RESUMO

This work presents a method that provides organizations and their project managers with a better understanding of the implications of the risks that software projects are subjected to. The method, which is based upon a novel approach to cluster analysis, generates a set of business rules indicating how risk factors may influence project performance. These rules are easy to read and understand, making it easier for managers to identify what needs to be done and when it needs to be done, before the worst happens.

Palavras-chave: cluster analysis, data mining, project management, Risk Analysis.

LISTA DE FIGURAS

Figura 2.1: Questionário de avaliação de risco e performance de projetos. . . . 28

Figura 3.1: Relação entre o métodos aglomerativos e divisivos. 34

Figura 4.1: Grafo Deolísticos de predicados restritos. 48

Figura 5.1: Diagrama produzido pelo algoritmo de grafos Deolisticos. 56

Figura 6.1: Tela Inicial. 62

Figura 6.2: Tela Principal. 63

Figura 6.3: Menu File. 64

Figura 6.4: Menu Edit. 64

Figura 6.5: Menu View. 65

Figura 6.6: Menu Model. 66

Figura 6.7: Configuração do Modelo. 66

Figura 6.8: Configuração das Variáveis Categóricas. 69

Figura 6.9: Configuração do ISG. 70

Figura 6.10: Configurações Avançadas. 70

Figura 6.11: Menu Report. 71

Figura 6.12: Árvore de Segmentação. 72

Figura 6.13: Detalhes de um nó da Árvore de Segmentação. 73

Figura 6.14: Tela Regras de Segmentação. 73

Figura 6.15: Tela Regras Factuais. 74

Figura 6.16: Tela Regras Factuais Estendidas. 74

Figura 6.17: Tela Comportamental Analysis. 74
Figura 6.18: Menu Window. 75
Figura 6.19: Menu Help. 75
Figura 6.20: Barra de Ferramentas. 75

Figura 8.1: Abrindo o arquivo. 92
Figura 8.2: Configurando as variáveis. 93
Figura 8.3: Configurando o método e a medida de distância. 93
Figura 8.4: Configurando tamanho da amostra. 93
Figura 8.5: Árvore de segmentação. 94
Figura 8.6: Menu Report. 94
Figura 8.7: Regras de segmentação. 95
Figura 8.8: Regras factuais. 95
Figura 8.9: Regras factuais extendidas. 96
Figura 8.10: Análise comportamental do nó terminal 01. 96
Figura 8.11: Análise comportamental do nó terminal 02. 97
Figura 8.12: Análise comportamental do nó terminal 03. 97
Figura 8.13: Análise comportamental do nó terminal 04. 98
Figura 8.14: Análise Comportamental do Nó Terminal 05. 98
Figura 8.15: Análise comportamental do nó terminal 06. 99
Figura 8.16: Análise comportamental do nó terminal 07. 99
Figura 8.17: Análise comportamental do nó terminal 08. 100
Figura 8.18: Análise comportamental do nó terminal 09. 100
Figura 8.19: Análise comportamental do nó terminal 10. 101

LISTA DE TABELAS

Tabela 2.1: Exemplos de origens para riscos. 21

Tabela 3.1: Etapas gerais do processo de clusterização. 33

Tabela 3.2: Exemplo de variáveis ordinais. 38

Tabela 3.3: Exemplo de variáveis intervalares. 38

Tabela 3.4: Exemplo de variáveis que assumem valores na escala de razão. . . 39

Tabela 3.5: Operações matemáticas as escalas de mensuração. 39

Tabela 3.6: Transformação com o auxílio de variáveis *dummies*. 41

Tabela 3.7: Exemplo do cálculo da distância de Mahalanobis. 43

Tabela 5.1: Dados coletados pela $\Delta\Sigma$ Corp. 54

Tabela 5.2: Relacionando fatores à dimensões. 55

Tabela 5.3: Dados resumidos em dimensões. 55

Tabela 5.4: Regras que distinguem as observações de um nó terminal. 57

Tabela 5.5: Estatísticas das dimensões de performance por nó terminal. . . . 57

Tabela 6.1: Principais características da ferramenta. 61

Tabela 6.2: Tela de configuração dos rótulos para variáveis categóricas. 67

Tabela 6.3: Configurando o algoritmo. 67

Tabela 6.4: Configurando parâmetros avançados. 68

LISTA DE ABREVIATURAS E SIGLAS

IT	Tecnologia da Informação

SUMÁRIO

1 INTRODUÇÃO 15

1.1 Objetivo deste Trabalho 17

1.2 Organização do Trabalho 17

2 GESTÃO DE RISCOS 19

2.1 Visão Geral 19

2.2 Riscos, critérios de sucesso e fatores de risco 20

2.3 O Processo de gestão de riscos 22

2.4 Identificação dos Fatores Riscos 23

2.5 Análise de Riscos 23

2.6 Planejamento de Ações para tratamento de Riscos 24

2.7 Monitoramento de Riscos 25

2.8 Controle de Riscos 25

2.9 Evolução na Gestão de Riscos em projetos de TI 25

3 ARCABOUÇO CONCEITUAL 30

3.1 Classificação e o Processo de Aprendizado 30

3.2 As Origens do Processo de Classificação 30

3.3 O Processo de Classificação 31

3.4 Métodos de Análise de Cluster 33

3.4.1 Métodos de Partição 33

3.4.2 Métodos Hierárquicos 34
3.4.3 Métodos Baseados em Densidade 35
3.4.4 Métodos Baseados em *Grid* 36
3.4.5 Métodos Baseados em Modelo 36
3.4.6 Escolhendo o Algoritmo Mais Adequado 36
3.5 Tipos de Variávies 37
3.6 Transformação de Variáveis 39
3.6.1 Variáveis Contínuas 39
3.6.2 Variáveis Discretas 40
3.7 Medidas de Distância 41
3.7.1 Distância Euclidiana 41
3.7.2 Distância City-Block on Manhattan 41
3.7.3 Mahalanobis Distance 42

4 ALGORITMO 44
4.1 Teoria dos Grafos 44
4.2 Grafos Deolísticos 46
4.3 Clusterizando com Grafos Deolísticos 50

5 O MÉTODO 53
5.1 Passo 0: Contextualizando 53
5.2 Passo 1: Coletando as informações 54
5.3 Passo 2: Preparando os dados para análise 54
5.4 Passo 3: Analisando os dados 55
5.5 Passo 4: Conclusões 57
5.6 Passo 5: Tomando ações corretivas 58
5.7 Sumarização do Método 59

6 A FERRAMENTA 61
6.1 Apresentando a Tela Inicial 62

6.2 Apresentando a Tela Principal 62
6.3 Apresentando as Opções de Menu 63
6.3.1 Menu *File* (Arquivo) 63
6.3.2 Menu *Edit* (Editar) 64
6.3.3 Menu *View* (Visualizar) 65
6.3.4 Menu *Model* (Modelo) 65
6.3.5 Menu *Report* (Relatórios) 69
6.3.6 Menu *Window* (Janelas) 71
6.3.7 Menu *Help* (Ajuda) 72
6.4 Apresentando a barra de ferramentas 73

7 DISCUSSÃO E CONCLUSÕES 76
7.1 Perguntas e Respostas 76
7.2 Conclusão 81
7.3 Trabalhos Futuros 82

REFERÊNCIAS 84

8 APÊNDICE 92

1 INTRODUÇÃO

Apesar de todo esforço empregado no desenvolvimento de conceitos, métodos e ferramentas para a condução eficiente de projetos de tecnologia da informação (TI), a verdade desalentadora é que projetos de TI continuam falhando em grande escala (DANIELS; LAMARS-II, 2007). Por exemplo, em 1994, o *Standish Report* indicou que apenas 20% de todos os projetos de TI envolvidos em um estudo, realizado no Estados Unidos, pelo Standish Group[1], foram considerados como tendo obtido sucesso (THE-STANDISH-GROUP, 2004). Neste relatório foram considerados projetos que falharam:

- aqueles que foram cancelados,
- aqueles que estouraram seus respectivos orçamentos,
- aqueles que foram entregues após o prazo, e
- aqueles que não alcançaram as expectativas dos usuários.

Desde 1994, o *Standish Report* tem sido atualizado em bases regulares, e, ao longo dos anos, se tornou uma referência na área de TI, sendo citado frequentemente tanto por acadêmicos quanto por profissionais de mercado no mundo todo[2]. Passada uma década, em 2004, o *Standish Report* apresentou apenas uma pequena melhoria.

Dos 50 mil projetos examinados pelo *Standish Report*, somente 29% foram considerados como tendo alcançado sucesso. Os projetos envolvidos na pesquisa estavam

[1] Uma empresa de consultoria norte americana.

[2] Isso pode ser facilmente constatado através de uma consulta ao Google ou ao Google Acadêmico.

divididos assim: 58% foram executados nos Estados Unidos, 27% na Europa e 15% no resto do mundo. Além disso 45% destes projetos pertenciam a empresas presentes na lista das 1.000 (mil) mais ricas do mundo, segundo a revista FORTUNE (MASTICOLA, 2007). Em adição, o relatório de 2004 indicou que o total de dinheiro gasto em projetos que falharam excedeu 50 bilhões de dólares somente nos Estados Unidos (BEER, 2004).

Apesar das críticas de alguns autores sobre a acurácia do *Standish Report* (GLASS, 2006), existem evidências substanciais na literatura que sugerem que a tendência de falha entre projetos de TI persiste até os dias de hoje (MOLØKKEN; JØRGENSEN, 2003).

Observe que falhas no gerenciamento de projetos de TI podem ocasionar uma grande variedade de problemas além da perda financeira, incluindo-se aqui a diminuição da eficiência e da produtividade, existência de informações incorretas para a tomada de decisão, atritos com os clientes, perda de competitividade e diminuição do faturamento (BHATT; GROVER, 2005).

Por outro lado, completar projetos de TI com sucesso tornou-se um requisito extremamente importante na "Era da Informação" na qual vivemos, onde um número cada vez maior de empresas volta seus negócios para a Internet, praticando e-business e e-commerce, e se esforçam para serem competitivas tanto no mundo real quanto no mundo virtual (BARNES, 2007).

Na medida em que as organizações continuam investindo em TI com o objetivo de se tornarem mais ágeis, eficiêntes e competitivas, eventos, condições e elementos que podem pôr em risco o sucesso de projetos de *software* tornam-se objetos de grande interesse para a atividade gerencial (BACCARINI; SALM; LOVE, 2004). No jargão de TI, estes eventos, condições e elementos são chamados de riscos de projetos de *software* (WALLACE; KEIL; RAI, 2004).

Defensores do gerenciamento de riscos argumentam que ao se identificar e analisar os elementos que podem obstar o sucesso de um projeto, muda-se a percepção que os gerentes possuem de seus projetos, favorecendo a execução de ações que reduzem

substancialmente a chance dos projetos falharem (DEY; KINCH; OGUNLANA, 2007; KEILL et al., 2006).

1.1 Objetivo deste Trabalho

Este trabalho apresenta um método que permite que as organizações e seus gerentes identifiquem com mais clareza e precisão os riscos a que estão expostos os projetos de *software* que executam. O método, que é baseado em uma nova abordagem para a análise de cluster, fornece um conjunto de regras de negócio não-ambíguas que permite identificar como um conjunto bem definido de fatores de risco pode influenciar a performance de projetos de *software* .

Como estas regras são baseadas em uma lógica de predicados restrita, elas são fáceis de serem lidas, compreendidas e comunicadas as partes interessadas, favorecendo a antecipação das ações gerenciais que evitam que o pior aconteça.

Este trabalho apresenta também um *software*, desenvolvido pelo autor, que facilita a utilização do método e fornece estatísticas que ajudam o processo de negociação com clientes e fornecedores.

Resultados deste trabalho foram publicados em 2008 nos anais da *Third IEEE/IFIP International Workshop on Business-driven IT Management (BDIM)*, que são publicados juntos com os anais do IEEE/IFIP Network Operations & Management Symposium (NOMS), com o título "Using a Novel Approach to Cluster Analysis to Gain New Valuable Insights into Software-Project Risk Management" (ALENCAR et al., 2008).

1.2 Organização do Trabalho

Este trabalho esta organizado da seguinte forma. No Capítulo 2 fazemos uma revisão dos principais conceitos envolvidos na gestão de riscos. No Capítulo 3 são apresentados os conceitos envolvidos no desenvolvimento e aplicação do método. O Capítulo 4 contém a descrição da nova abordagem para a análise de cluster, a descrição formal de grafos Deolísticos e de um algoritmo de clusterização baseado

nestes grafos. No Capítulo 5 apresentamos o método com o auxílio de um exemplo. A ferramenta computacional que facilita o uso do algoritmo de clusterização é apresentada no Capítulo 6. No Capítulo 7 oferecemos uma discussão das implicações da existência do método para diferentes aspectos das estratégias de negócio e do gerenciamento de risco, assim como apresentamos as conclusões deste trabalho. O Apêndice apresenta a seqüência passo-a-passo de como a ferramenta descrita no Capítulo 6 foi utilizada para se obter os resultados apresentados no Capítulo 7.

2 GESTÃO DE RISCOS

2.1 Visão Geral

Os riscos são inerentes a qualquer projeto, porque eles dizem respeito a incertezas sobre eventos que poderão ou não ocorrer no futuro e o futuro é incerto por natureza (KENDRICK, 2009). Em conseqüência, como projetos estão repletos de incertezas, eles estão expostos a todo tipo de risco. Por exemplo,

- A equipe do projeto tem o tamanho adequado?
- Os objetivos estão claramente e completamente definidos?
- Os recursos financeiros colocados a disposição do projeto são suficiente?
- Ao produzir o resultado final a equipe do projeto estará infringindo a lei de alguma maneira?

Embora os riscos apresentem desafios para o gerente de projetos, a capacidade de assumir riscos é essencial para nossa evolução e as falhas são uma excelente oportunidade de aprendizado (GLENDON; CLARK; MCKENNA, 2006). Além disso, do ponto de vista econômico-financeiro, sem risco não há lucro, sem lucro não existe capacidade de investimento e sem investimento os produtos e serviços que oferecemos ficam desatualizados. Em um mercado competitivo, desatualização é receita certa para a perda da preferência dos clientes, queda nas vendas, diminuição do faturamento e, finalmente, para a falência do negócio.(ROBERT, 2000)

Quando os gerentes de projetos tentam tratar os riscos que seus projetos estão expostos, eles buscam influenciar, de alguma forma, o ambiente em que o projeto está (ou estará) sendo executado e sua vizinhança imediata, com vistas a redução das chances de resultados negativos.

Como estes ambientes costumam ser complexos, estarem repletos de incertezas e sujeitos a um grande número de possibilidades (isto é, à ação de um grande número de variáveis), ao longo do tempo, a forma como os riscos são gerenciados pode afetar de forma contundente as chances de sucesso de um projeto (WALLACE; KEIL; RAI, 2004).

2.2 Riscos, critérios de sucesso e fatores de risco

Antes que a discussão sobre gestão de riscos possa ser aprofundada, devemos primeiro definir o que são riscos a que projetos estão expostos:

> *Risco diz respeito a probabilidade de ocorrência de resultados negativos em um projeto*

Portanto os riscos estão relacionados a forma como o sucesso de um projeto é avaliado. Por exemplo, o não cumprimento do prazo acordado, o custo final do projeto sendo maior do que o valor orçado, produtos entregues que não atendem as expectativas dos usuários, etc.

O ambiente em que os projetos são executados pode favorecer consideravelmente a ocorrência ou origem dos riscos. Por exemplo, podemo citar : a pressão competitiva do mercado em que a empresa atua, o grau de saúde financeira da empresa, a sua cultura organizacional, entre outras (TURNER, 2006).

Na Tabela 2.1 temos exemplos de origens e conseqüências de riscos para os projetos.

Para que possamos avaliar se um projeto foi bem sucedido ou não temos que avaliar se os critérios de sucesso foram atendidos. Estes critérios são também conhecidos como critérios de performance. De forma geral, para um projeto XYZ

podemos ter os seguintes de critério de sucesso: (SCHMITZ; ALENCAR; VILAR, 2007)

- O prazo para entrada em funcionamento deve ser menor que seis meses
- O custo total do projeto deve ser menor quer US$ 500 mil e
- O estoque de produtos intermediários dever ser reduzido em 75%

Cada critério de sucesso deve estar associado a uma medida de interesse que avalia o atendimento a este critério. Por exemplo,

- O prazo decorrido entre início e término do projeto
- O custo total do projeto e
- A quantidade de produtos intermediários em estoque.

A variação nos valores destas medidas de interesse é influenciada por um conjunto de valores e eventos denominadas fatores de risco. Seguindo nosso exemplo, uma lista de fatores de riscos possíveis são:

- Qualificação da equipe técnica para personalizar o software;
- Facilidade de adaptação do pacote de software que está sendo utilizado e
- Motivação do usuário para efetuar as mudanças nos processos de negócio que se fizerem necessárias.

Tabela 2.1: Exemplos de origens para riscos.

Categorias de Origens de Risco		**Conseqüências Para os Projetos**
Missão e objetivos		Orçamento excedido
Fatores de tomada de decisão		Desvio do cronograma
Gestão Organizational		Funcionalidades inadequadas
Clientes e Usuários finais		Projetos cancelados
Orçamento e custo		Trocas inesperadas de pessoas chaves
Cronograma	$\Longrightarrow$	Insatisfação do cliente
Características do Projeto		Perda de imagem da empresa
Processo de desenvolvimento		Equipe desmoralizada
Ambiente de desenvolvimento		Produtos de baixa performance
Pessoas		Procedimentos legais
Ambiente Operacional		
Novas tecnologias		

Como valores possíveis para o fator de risco "Motivação do usuário para efetuar as mudanças nos processos de negócio que se fizerem necessárias"temos:

- Muito motivado,
- Passivo ou,
- Rejeição completa ao projeto.

Cada valor possível possui uma probabilidade de ocorrência. Assim sendo, se as chances de ocorrência de "Rejeição completa"for de 90%, contra somente 10% de "Muito motivado", então as chances de ocorrência de um cenário pessimista em relação aos critérios de sucesso são maiores.

Os fatores de risco de um projeto podem ser vistos como eventos cuja ocorrência é incerta e que, se vierem a se realizar, têm um efeito negativo nos critérios de sucesso de um projeto. (PMBOK, 2000)

2.3 O Processo de gestão de riscos

Muitas equipes de projeto buscam lidar com os riscos apenas quando eles ocorrem ou estão prestes a ocorrer, sendo necessário, neste caso, a execução de ações emergênciais de contingência. Esta forma de lidar com os riscos é conhecido como gerenciamento reativo e causa grandes desgastes nas equipes de desenvolvimento e no projeto como um todo.

O processo de gestão de riscos, também conhecido com gerência de riscos, faz uso de ferramentas que possibilitam agir de forma pró-ativa na tomada de decisões e execução de ações que visam evitar que fatores de risco assumam valores que são prejudiciais a um projeto (TURNER, 2006). As atividades do processo de gestão de riscos podem ser divididas em

- identificar os fatores de risco,
- analisar o impacto e a probabilidade de ocorrência deste fatores sobre o projeto,

- planejar ações de tratamento de risco,
- monitorar a aplicação das ações que foram planejadas e a evolução dos fatores de risco ao longo do tempo, e
- controlar o surgimento de novos fatores de risco.

2.4 Identificação dos Fatores Riscos

A primeira atividade da gestão do risco é a identificação dos fatores riscos que podem afetar os critérios de sucesso do projeto. Esta é uma atividade de descoberta e, portanto, difícil de ser sistematizada. Algumas técnicas são utilizadas para facilitar a identificação destes fatores de risco, entre elas podemos citar o uso de *check lists*, *brainstorm* e a técnica Delphi. (SCHMITZ; ALENCAR; VILAR, 2007)

2.5 Análise de Riscos

Nesta atividade buscamos estimar qual a probabilidade ou qual a chance de ocorrência do fator de risco, este valor varia entre 0% e 100%, e, além disto, estimar qual o possível impacto dos fatores de risco nos critérios de sucesso do projeto. (TURNER, 2006)

Quando um evento que determina a ocorrência de um fator de risco ocorre, os critérios de sucesso do projeto serão impactados de alguma forma. Para avaliarmos qual o nível do impacto nestes critérios, classificamos fatores de risco da seguinte forma:

- os que causam um baixo impacto,
- os que causam um impacto médio e
- aquele que geram um alto impacto prejudicando de forma significativa o sucesso do projeto.

Analisando a probabilidade e o impacto de um fator de risco, o gerente de projeto pode ser defrontado com as seguintes situações, entre outras:

- Um fator de risco possui grande probabilidade de ocorrência, porém um nível de impacto baixo. Neste caso ele pode optar por assumir uma postura otimista e decidir não tratá-lo.

- Para um cenário onde temos uma baixa probabilidade de ocorrência, porém um alto nível de impacto, talvez seja melhor o gerente assumir uma postura mais pessimista e agir de forma pró-ativa, buscando minimizar a probabilidade da ocorrência ou minimizar o impacto nos critérios de sucesso. Evitando portanto que o pior aconteça.

2.6 Planejamento de Ações para tratamento de Riscos

A atividade de planejamento é a terceira atividade do processo gestão de risco. Nesta atividade as informações sobre os fatores de risco, probabilidade e impacto, são usadas como insumos para tomada de decisões e definição de quais das três ações de tratamento serão utilizadas : eliminar, contingênciar ou ignorar o fator de risco.

Um forma de eliminar um fator de risco é adicionar novas atividades ao projeto, como por exemplo definir um treinamento exaustivo para a equipe técnica. Outra forma seria delegar a atividade a uma empresa especializada.

Nem sempre é possível eliminar todos os riscos de um projeto, até porque a probabilidade de ocorrência de todos os riscos no mesmo projeto é praticamente nula. Porém não podemos negligenciar a sua ocorrência. Tudo isso leva a inserção de atividade de contingênciamento, isto é, que atividades serão executadas caso um dos fatores de riscos venha a se realizar. Note que a ocorrência de um fator de risco pode ser monitorada através de indicadores associados a eles. (TURNER, 2006)

Estas atividades sempre estão acompanhadas de ações para redução da probabilidade ou do impacto do fator de risco. Esta ação é conhecida pelo nome de mitigação do fator de risco. (SCHMITZ; ALENCAR, 2007)

Por fim, aquele fatores de risco que possuem uma baixíssima probabilidade de ocorrência e um impacto desprezível, poderiam ser ignorados. Ao viés de sua ocorrência, lidar com suas conseqüências é algo que o gerente do projeto pode

facilmente executar sem prejudicar o orçamento e o prazo do projeto.

2.7 Monitoramento de Riscos

A quarta atividade da gestão de risco é o monitoramento. Nesta atividade os fatores de risco são acompanhados periodicamente através de indicadores que auxiliam o seu responsável a avaliar a eminência de sua ocorrência e o impacto esperado.

Durante as reuniões de revisão do projeto, o relatório de status dos riscos irá apresentar as informações de situação dos indicadores, quais os riscos que foram eliminados, qual o andamento atual dos planos de tratamento dos riscos, entre outras.

2.8 Controle de Riscos

O controle dos riscos é o último passo da gestão de risco. Esta atividade tem como objetivo controlar as variações no plano de tratamento dos riscos, definindo as ações para correção do plano. Por exemplo, dado que um fator de risco que possui um plano de contingência venha a ocorrer, controlar as ações ou as atividades prevista no plano são responsabilidades desta fase da gestão do risco.

2.9 Evolução na Gestão de Riscos em projetos de TI

Quando os gerentes de projetos tentam tratar os riscos que seus projetos estão expostos, eles buscam influenciar, de alguma forma, o ambiente em que o projeto está (ou estará) sendo executado e sua vizinhança imediata com vistas à redução das chances de resultados negativos. Como estes ambientes costumam ser complexos, estar repletos de incertezas e sujeitos a um grande número de possibilidades (isto é, à ação de um grande número de variáveis), ao longo do tempo, diversos modelos tem sido propostos com o intuito the facilitar a identificação dos riscos que podem afetar de forma mais intensa as chances de sucesso de um projeto (WALLACE; KEIL; RAI, 2004).

Em um modelo de análise de risco, as informações relacionadas a risco costumam ser representadas por um conjunto finito de variáveis, chamadas fatores de risco, que influenciam negativamente as chances de sucesso de um projeto a partir do momento em que assumem valores dentro de um conjunto indesejado de elementos (WESTERMAN; HUNTER, 2007).

Em sintonia com estas definições, ao longo do tempo, várias listas de fatores de risco relacionados a TI foram propostas, tais como (MOYNIHAN, 1989), (MOYNIHAN, 1997), (KAROLAK, 1995), (HEEMSTRA; KUSTERS, 1996), (SCHMIDT et al., 2001), (VITHARANA, 2003), e, mais recentemente, (WALLACE; KEIL; RAI, 2004).

Além de ajudar os gerentes a identificar os riscos com maior facilidade, em geral, estes modelos (na forma de listas de fatores de risco) fornecem meios para se relacionar fatores de risco com indicadores de desempenho de projetos tais como : prazo, custo e conformidade com os requisitos funcionais do usuário, permitindo que o gerente reflita sobre o melhor curso de ação a ser seguido em cada situação, através de um processo de tentativa e erro.

Por exemplo, Moynihan (MOYNIHAN, 1989, 1997), um dos pioneiros na identificação e gerenciamento de riscos relacionados a TI, sugere uma lista de vinte e três fatores de risco que são relacionados a quatro dimensões de performance de projetos, a saber: (1) adequação ao custo, (2) cumprimento do prazo, (3) conformidade com as especificações de requisitos e(4) robustez do produto final.

Ao modificar alguns elementos do projeto, o gerente pode tentar influenciar a probabilidade de ocorrência de um ou mais fatores de risco e, em conseqüência, influenciar positivamente os indicadores de performance de projeto.

Já Karolak (KAROLAK, 1995), e Heemstra e Kusters (HEEMSTRA; KUSTERS, 1996) propõem o uso de listas mais extensas de fatores de risco, contendo respectivamente oitenta e sete, e cento e trinta e dois fatores de risco. Para tornar mais fácil a reflexão sobre como os fatores de risco podem influenciar o sucesso de um projeto, Karolak agrupa os fatores de risco que identificou em nove dimensões de risco,

incluindo a maturidade da estrutura organizacional, a acurácia das estimativas de custo, a efetividade dos mecanismos de identificação de problemas, a familiaridade com as metodologias de desenvolvimento, etc. Essas dimensões são então conectadas a indicadores de performance de projetos.

O modelo proposto por Heemstra and Kusters também agrupa fatores de risco em dimensões. Entretanto, estes autores sugerem que a conexão com os indicadores de performance deve ser feita pelo gerente de projeto, levando em consideração o ambiente de negócio no qual o projeto esta sendo desenvolvido.

Schmidt *et al.* (SCHMIDT et al., 2001) dão um passo adiante no uso de modelos de análise de risco, ao demonstrar a importância de se considerar a influência das diferenças culturais sobre os riscos a que um projeto de TI está exposto. A contribuição do trabalho de Schmidt *et al.* se traduz em uma lista culturalmente ajustada de fatores de risco, que são agrupados em quatorze dimensões de risco através de um conjunto de equações.

O trabalho de Linda Wallace *et al.* (WALLACE; KEIL; RAI, 2004) fornece uma contribuição importante para um melhor entendimento da relação que existe entre os fatores de risco e os indicadores de performance de projetos. Para obter os resultados apresentados no seu trabalho, Linda Wallace *et al.* elaboraram um questionário baseado na escala de Likert [1] contendo dezenas de afirmações (*statements*) sobre o estado corrente de um projeto de *software*. Posteriormente, este questionário foi utilizado para coletar dados de iniciativas de *software* em uma grande quantidade de organizações nos Estados Unidos e outros lugares do mundo.

Uma vez que os dados foram coletados, Linda Wallace *et al.* utilizaram análise fatorial (*factor analysis*) para agrupar os fatores de risco em dimensões de risco, que foram relacionadas com indicadores de performance de projeto com o auxílio de técnicas de análise de cluster. (HAIR et al., 2005) apresentam uma discussão abrangente sobre análise fatorial. A Figura **??** apresenta o questionário elaborado por Linda Wallace *et al.*

[1] Concebida pelo psicólogo e pesquisador Americano Rensis Likert (1903–1981).

Part I: Risk Perception

#	Risk Dimension	#	Risk Factor	Evaluation Scale ↓ *strongly disagree* 1	2	3	4	5	6	7 *strongly agree* ↓
RD_1	Team	RF_1	Frequent conflicts between development team members	○	○	○	○	○	○	○
		RF_2	Frequent turnover within the project team	○	○	○	○	○	○	○
		RF_3	Team members not familiar with the task(s) being automated	○	○	○	○	○	○	○
		RF_4	Team members lack specialized skills required by the project	○	○	○	○	○	○	○
		RF_5	Inadequately trained development team members	○	○	○	○	○	○	○
		RF_6	Team members lack commitment to the project	○	○	○	○	○	○	○
		RF_7	Inexperienced team members	○	○	○	○	○	○	○
RD_2	Organizational Environment	RF_8	Lack of top management support for the project	○	○	○	○	○	○	○
		RF_9	Change in organizational management during the project	○	○	○	○	○	○	○
		RF_{10}	Organization undergoing restructuring during the project	○	○	○	○	○	○	○
		RF_{11}	Unstable organizational environment	○	○	○	○	○	○	○
		RF_{12}	Corporate politics with negative effect on project	○	○	○	○	○	○	○
		RF_{13}	Changes in priorities shift resources away from the project	○	○	○	○	○	○	○
RD_3	Requirements	RF_{14}	Incorrect system requirements	○	○	○	○	○	○	○
		RF_{15}	Users lack understanding of system capabilities and limitations	○	○	○	○	○	○	○
		RF_{16}	Undefined project success criteria	○	○	○	○	○	○	○
		RF_{17}	Conflicting system requirements	○	○	○	○	○	○	○
		RF_{18}	Difficulty in defining the inputs and outputs of the system	○	○	○	○	○	○	○
		RF_{19}	Unclear system requirements	○	○	○	○	○	○	○
		RF_{20}	System requirements not adequately identified	○	○	○	○	○	○	○
		RF_{21}	Continually changing system requirements	○	○	○	○	○	○	○
RD_4	Planing and Control	RF_{22}	Project milestones not clearly defined	○	○	○	○	○	○	○
		RF_{23}	Project progress not monitored closely enough	○	○	○	○	○	○	○
		RF_{24}	Lack of an effective project management methodology	○	○	○	○	○	○	○
		RF_{25}	Inexperienced project manager	○	○	○	○	○	○	○
		RF_{26}	Poor project planning	○	○	○	○	○	○	○
		RF_{27}	Lack of people skills in project leadership	○	○	○	○	○	○	○
		RF_{28}	Ineffective communication	○	○	○	○	○	○	○
		RF_{29}	Inadequate estimation of required resources	○	○	○	○	○	○	○
		RF_{30}	Inadequate estimation of project schedule	○	○	○	○	○	○	○
RD_5	User	RF_{31}	Lack of cooperation from users	○	○	○	○	○	○	○
		RF_{32}	Users resistant to change	○	○	○	○	○	○	○
		RF_{33}	Users not committed to the project	○	○	○	○	○	○	○
		RF_{34}	Lack of user participation	○	○	○	○	○	○	○
		RF_{35}	Conflict between users	○	○	○	○	○	○	○
		RF_{36}	Users with negative attitudes toward the project	○	○	○	○	○	○	○
RD_6	Complexity	RF_{37}	Project uses technology that has not been used in prior projects	○	○	○	○	○	○	○
		RF_{38}	Large number of links to other systems required	○	○	○	○	○	○	○
		RF_{39}	High level of technical complexity	○	○	○	○	○	○	○
		RF_{40}	One of the largest projects attempted by the organization	○	○	○	○	○	○	○
		RF_{41}	Project involved the use of new technology	○	○	○	○	○	○	○
		RF_{42}	Many external suppliers involved in the development project	○	○	○	○	○	○	○
		RF_{43}	Immature technology	○	○	○	○	○	○	○
		RF_{44}	Highly complex task being automated	○	○	○	○	○	○	○

Part II: Project Performance Perception

#	Performance Dimension	#	Project Performance Factor	Evaluation Scale ↓ *strongly disagree* 1	2	3	4	5	6	7 *strongly agree* ↓
PD_1	Technical	PPF_1	The application developed is reliable	○	○	○	○	○	○	○
		PPF_2	The application developed is easy to maintain	○	○	○	○	○	○	○
		PPF_3	The system meets intended functional requirements	○	○	○	○	○	○	○
PD_2	User	PPF_4	The users are satisfied with the developed application	○	○	○	○	○	○	○
		PPF_5	The system meets user response-time expectations	○	○	○	○	○	○	○
		PPF_6	The system meets user ease-of-use expectations	○	○	○	○	○	○	○
		PPF_7	The overall quality of the developed application is high	○	○	○	○	○	○	○
PD_3	Cost	PPF_8	The system was completed within budget	○	○	○	○	○	○	○
PD_4	Time	PPF_9	The system was completed within schedule	○	○	○	○	○	○	○

Figura 2.1: Questionário de avaliação de risco e performance de projetos.

A despeito de toda a evolução que a análise de risco tem experimentado nestas últimas décadas e da influência positiva que tem exercido no gerenciamento de projetos de software de grande complexidade e magnitude, pouco foi feito para desenvolver modelos que indiquem claramente as ações que os gerentes devem tomar (assim como as conseqüências destas ações) para aumentar as chances de sucesso

dos projetos pelos quais são responsáveis. Existe, no entanto, uma excessão que vale a pena ser mencionada. Alencar *et al.* (ALENCAR et al., 2007) usam um modelo baseado em simulação de Monte Carlo e árvore de classificação decorada para indicar que conjunto de ações gerenciais favorecem positivamente a entrega de um projeto de *software* dentro do prazo.

3 ARCABOUÇO CONCEITUAL

3.1 Classificação e o Processo de Aprendizado

Um dos principais elementos do processo de aprendizagem humana, que nos distinguem da maioria dos seres vivos a nossa volta, é a nossa capacidade de agrupar objetos que compartilham propriedades similares. Ao nos conscientizarmos que certos grupos de objetos compartilham as mesmas propriedades, podemos aplicar o conhecimento previamente adquirido a objetos que encontramos pela primeira vez.

Sem esta habilidade fundamental, que os seres humanos desenvolvem instintivamente ao longo do seu processo evolutivo, qualquer criatura inteligente se encontraria em uma situação na qual todo objeto cuja existência ela perceba pela primeira vez teria que ser tratado como uma entidade única, completamente diferente de qualquer outra no universo. Tudo isso tornaria o processo de aprendizado uma tarefa muito mais complexa do que ela já é (KAUFMAN; ROUSSEEUW, 1990).

3.2 As Origens do Processo de Classificação

A idéia de agrupar objetos, isto é, classificação, é tão antiga quanto a própria raça humana. Sua origem tem raízes na época em que os primeiros seres humanos disputavam, diariamente, comida e abrigo com outros seres vivos. Neste ambiente extremamente hostil, saber que tipos de frutas e raízes são comestíveis, que grupos de animais apresentam perigo, que tipos de abrigos provêm segurança e que tipos de fontes produzem água potável, era crítico para a sobrevivência (DAVIS, 2004).

Ao desempenhar um papel central na luta pela sobrevivência dos primeiros seres humanos, a classificação acabou por encontrar o seu caminho para as linguagens escritas e faladas, que consistem, em termos gerais, de grupos de palavras que nos permitem referenciar, de forma apropriada, a eventos, entidades, objetos e pessoas que encontramos.

Entre estes grupos de palavras encontramos os substantivos, que agrupam seres vivos, objetos inanimados, lugares e atividades que compartilham similaridades, tais como: animais, carros, lagos e caça. Em adição, encontramos os adjetivos, que propiciam um refinamento maior do processo de classificação, permitindo uma descrição mais precisa dos substantivos que desejamos nos referir. Por exemplo: pássaros coloridos, maçãs vermelhas, mulheres bonitas, idéias brilhantes, etc. (THOMSON; MARTINET, 1986).

No decorrer do tempo, o uso de classificação nas linguagens faladas e escritas fez com que ela se tornasse parte intrínseca do nosso dia-a-dia. Nos dias de hoje, no mundo cada vez mais complexo em que vivemos, a classificação se tornou parte integrante de todos os ramos da ciência e áreas de negócio. A importância da classificação para o desenvolvimento científico e para os negócios é sintetizada na seguinte citação de Carl Von Linné (LINNAEUS, 1737), naturalista sueco do século XVI:

> "Todo o conhecimento que adquirimos depende do método que utilizamos para distinguir o similar do dissimilar. Quanto maior o número de diferenças que este método consegue distinguir, mais claro tornam-se as nossas idéias sobre as coisas. Quanto maior for o número de objetos que atraem nossa atenção, mais difícil é formar tal método, e mais necessário isso se torna".

3.3 O Processo de Classificação

A relevância da classificação para o aprendizado tem motivado extensas investigações matemáticas sobre assunto. Neste sentido, a análise de cluster é a área

de pesquisa na qual os métodos quantitativos para classificação são concebidos, analisados e refinados.

Do ponto de vista estritamente numérico, clusterização, isto é, classificação, é um processo iterativo de vários passos onde informações sobre um determinado conjunto de objetos são capturadas e utilizadas sucessivamente para classificá-los em segmentos. É desejável que esses segmentos possuam alto grau de homogeneidade interno e alto grau de heterogeneidade externa, isto é: entre segmentos. A primeira fase do processo chega ao fim quando um determinado número de segmentos é obtido (EVERITT; LANDAU; LEESE, 2001).

Uma vez que a primeira fase tenha sido completada, o analista de dados tem que executar duas tarefas importantes: identificar o que distingue os segmentos revelados pelo processo de clusterização e determinar que propriedades os objetos colocados em um mesmo segmento têm em comum. É importante se observar que estas distinções e propriedades são o que torna, em última instância, o conjunto de segmentos úteis para fins práticos. Sem um entendimento claro destas diferenças e similaridades, o aprendizado não ocorre no seu sentido mais amplo. A Tabela 3.1 sumariza as etapas gerais de um processo de clusterização (HAIR et al., 2005).

Infelizmente, a despeito de todo o desenvolvimento que a análise de cluster tem sofrido nestas últimas décadas, os métodos de clusterização atuais tendem a oferecer suporte limitado neste aspecto. Como conseqüência muitos analistas de dados recorrem á estatística descritiva e testes de hipóteses para estabelecer estas diferenças e similaridades.

Tudo isso tende a tornar a análise de cluster uma ferramenta de análise multivariada que muitos profissionais acham difícil de usar, especialmente aqueles que têm conhecimento restrito de matemática e estatística. Este é justamente o caso da grande maioria dos que se encontram empregados nas organizações dos mais diversos tipos e tamanhos. Everitt *et al.* (EVERITT; LANDAU; LEESE, 2001) apresentam uma discussão abrangente sobre métodos e conceitos utilizados na análise de cluster.

3.4 Métodos de Análise de Cluster

Segundo (EVERITT; LANDAU; LEESE, 2001; MYERS, 1996; BERSON; SMITH; THEARLING, 2002), a despeito da enorme variedade de métodos de clusterização, os algoritmos existentes podem ser classificados em cinco grupos distintos, a saber:

- métodos de partição,
- métodos hierárquicos,
- métodos baseados em densidade,
- métodos baseados em *grid* e
- métodos baseados em modelos.

3.4.1 Métodos de Partição

Os métodos de partição criam inicialmente, de forma arbitrária, k partições, onde k é um parâmetro fornecido pelo usuário. Em seguida, cada objeto é associado

Tabela 3.1: Etapas gerais do processo de clusterização.

Etapa		**Descrição**
Número	**Nome**	
1	Coleta de Dados	Seleciona os objetos a serem classificados e coleta tantas informações relevantes sobre estes objetos quanto for possível.
2	Seleção das Variáveis	Escolhe as dimensões (variáveis) a serem consideradas no processo de clusterização.
3	Seleção do Método	Indica o método de classificação que será utilizado para congregar objetos similares.
4	Escolha da Métrica de Similaridade	Seleciona a métrica de similaridade a ser utilizada pelo método de classificação para calcular o grau de similaridade entre dois objetos quaisquer.
5	Transformação de Variáveis	Em sintonia com a métrica de similaridade transforma as variáveis de tal forma que elas possam ser utilizadas corretamente no processo de clusterização.
6	Número de Segmentos	Seleciona o número de segmentos a serem produzidos pelo processo de clusterização.
7	Classificação	Aplica o método de clusterização a um dado conjunto de objetos.
8	Descrição e Avaliação	Descreve e avalia os segmentos produzidos pelo método de clusterização. Se os segmentos são satisfatórios vá para a Etapa 9, caso contrário revise todas as decisões tomadas até o momento e volte a Etapa 1.
9	Nomeação	Nomeia os diversos segmentos de objetos produzidos pelo processo de clusterização.

a uma determinada partição de acordo com uma métrica de distância também de escolha do usuário. Posteriormente, o centro de cada partição é recalculado e os objetos são reassociados. O processo termina quando o sistema adquire um nível de estabilidade pré-determinado pelo usuário. São exemplos de algoritmos de partição: K-means, K-medoids e CLARANS e suas melhorias (BERSON; SMITH; THEARLING, 2002; WISHART, 2001). São exemplos de métricas de distância: a distância Euclidiana, a distância de Mahalanobis, a distância Manhattan e a distância City Block (EVERITT; LANDAU; LEESE, 2001).

3.4.2 Métodos Hierárquicos

Os métodos hierárquicos criam uma decomposição hierárquica para um certo conjunto de dados. Eles podem ser classificados em métodos aglomerativos (*bottom-up*) e métodos divisivos (*top-down*), dependendo de como a decomposição é efetuada.

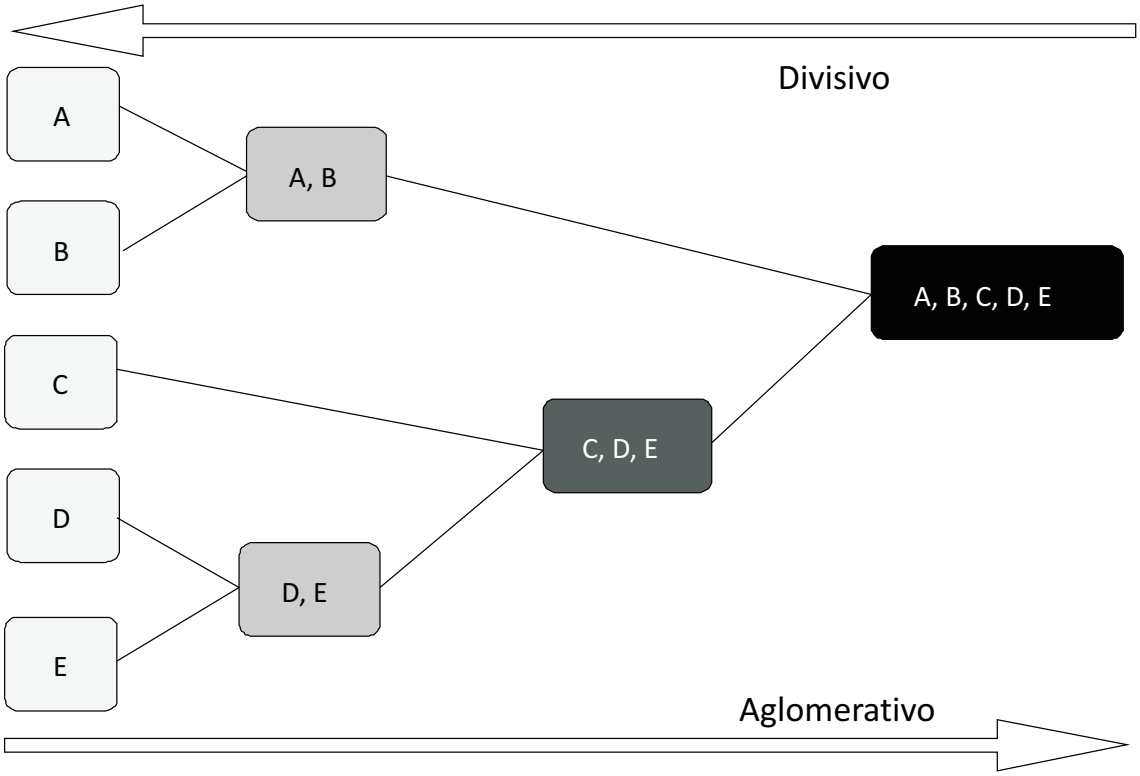


Figura 3.1: Relação entre o métodos aglomerativos e divisivos.

Na abordagem aglomerativa, cada objeto é inicialmente associado a um segmento distinto. De forma sucessiva cada um destes segmentos é agrupado em novos seg-

mentos, de acordo com a distância entre eles. Por sua vez, cada um desses recém criados segmentos é agrupado em novos segmentos, e assim por diante. O processo termina quando não existem mais segmentos para serem agrupados. Exemplos de métodos aglomerativos são :

- *Single Linkage* - onde dois segmentos passam a formar um segmento maior se a distância entre seus objetos mais próximos é a menor de todas,
- *Complete Linkage* - onde dois segmentos passam a formar um segmento maior se a distância entre seus objetos mais afastados é a menor de todas,
- *Average Linkage* - onde dois segmentos passam a formar um segmento novo se a média das distâncias entre seus membros e a menor de todas, e
- *Centroid Linkage* - onde dois segmentos passam a formar um segmento maior se a distância entre os centros dos dois segmentos é a menor de todas.

Na abordagem divisiva, o processo é inverso. Inicialmente todos os objetos fazem parte de um único segmento. Este segmento inicial é dividido em novos segmentos de acordo com algum critério escolhido pelo usuário. Por sua vez, cada um destes segmentos recém criados é dividido em novos segmentos, e assim por diante.

O processo termina quando não existem mais segmentos para serem divididos, ou quando algum critério de término do processo previamente estabelecido for atingido. Neste dois casos, a seqüência de aglomeração ou divisão de objetos estabelece uma ordem ou hierarquia entre eles (LAAN; POLLARD, 2003). Um exemplo de método divisivo é o método de *MacNaughton-Smith* (VENABLES; RIPLEY, 2003).

A figura 3.4.2 apresenta a diferença entre as abordagens aglomerativa e divisiva.

3.4.3 Métodos Baseados em Densidade

Os métodos baseados em densidade partem do princípio de que em um espaço métrico n-dimensional existem regiões com diferentes densidades de pontos. Em outras palavras, enquanto uma determinada região do espaço apresenta uma alta densidade de pontos, outras regiões apresentam densidades relativamente baixas. Os

algoritmos que exploram esta idéia agrupam os pontos que fazem parte de regiões de alta densidade em segmentos distintos. Pontos em regiões pouco densas do espaço tendem a formar segmentos com pequeno número de objetos. Uma medida relativamente popular de densidade de um espaço n-dimensional é a menor esfera em torno de um ponto que contém k observações, onde k é um parâmetro fornecido pelo usuário (WONG; LANE, 1983).

3.4.4 Métodos Baseados em *Grid*

Os métodos baseados em *grid* dividem um espaço n-dimensional em um determinado número de regiões delimitadas que formam uma estrutura de *grid*. As operações de agrupamento são feitas com base na posição dos objetos na *Grid*. Exemplos típicos destes métodos são : *Sting*, *WaveCluster* e *Clique*.

3.4.5 Métodos Baseados em Modelo

Os métodos baseados em modelos efetuam ajustes em uma estrutura matemática pré-determinada de tal forma que um conjunto de dados é dividido no melhor número de segmentos possível. O critério utilizado para efetuar os ajustes varia de acordo com a estrutura matemática escolhida e a determinada pelo usuário. Existem, basicamente, dois grandes grupos de métodos baseados em modelos: as redes neurais e os métodos baseados em funções de densidade de probabilidade.

3.4.6 Escolhendo o Algoritmo Mais Adequado

A diversidade de algoritmos de clusterização existentes na literatura permite que o usuário faça uma escolha daquele que é mais adequado ao tipo de dado disponível e ao objetivo da análise. A análise de cluster é utilizada como uma ferramenta descritiva e exploratória e é possível experimentar vários algoritmos para um mesmo conjunto de dados para observar o que é revelado, e, assim, efetuar a melhor escolha possível (HAN; KAMBER, 2001).

3.5 Tipos de Variávies

No final da década de 40, Stanley S. Stevens (1906-1973), um pesquisador da Universidade de Harvard, nos Estados Unidos, simplificou consideravelmente a forma como refletimos sobre as variáveis de um experimento ao propor um modelo para classificá-las segundo à natureza da informação que elas contém.

Segundo Stevens *op. cit.* ao coletarmos valores sobre os quais vamos realizar análises estatísticas e agruparmos estes valores em variáveis, é importante ficarmos atentos às operações matemáticas e estatísticas que podem ser executadas sobre as variáveis, já que nem todos os tipos de valores se prestam a serem manipulados por todos os tipos de operações (STEVENS, 1946).

Por exemplo, sobre valores tais como distância, peso e quantidade podem ser executadas tanto operações matemáticas simples tais como: soma, subtração, divisão e multiplicação, quanto operações matemáticas mais complexas tais como raiz quadrada, logaritmo e exponenciação. Entretanto, as mesmas operações não podem ser executadas sobre dados tais como sexo, estado civil e profissão.

De acordo com o modelo proposto por Stevens *op. cit.* os valores (e em decorrência as variáveis) se agrupam em quatro níveis de mensuração, a saber: nominal, ordinal, intervalar e razão, cada um com seu conjunto particular de propriedades.

A escala nominal é a que oferece a menor capacidade de mensuração. Em adição, o conjunto de operações matemáticas e estatísticas que podem ser executadas sobre estes dados é o menor de todos. Nesta escala cada valor é representado por um rótulo, cuja escolha é arbitrária. São exemplos de dados de escala nominal: sexo, estado civil, profissão, bairro, marca de relógio, tipo de automóvel, etc. Algumas vezes variáveis nominais são apresentadas como categóricas.

Na escala ordinal, os valores, além de representarem coleções de objetos, podem ser ordenados do menor para o maior ou vice-versa. Nesta escala, comparações de menor, maior e igual podem ser feitas. Entretanto, outras operações como subtração e adição não possuem significado. Na Tabela 3.2 são apresentados alguns exemplos de variáveis que assumem valores na escala nominal.

Apesar de sua semelhança com a escala ordinal, na escala de mensuração intervalar a diferença entre os valores possui significado, embora não exista um zero absoluto. Por exemplo, nas escalas Celsius e Fahrenheit, que são utilizadas para medir temperatura, não existe um ponto de partida natural, já que o ponto zero da escala não índica ausência de temperatura. Um tipo de escala que é freqüentemente tratada como intervalar é a escala de Likert (BRACE, 2008). A Tabela 3.3 oferece exemplos de variáveis desta escala.

A escala de razão é aquela que possui maior capacidade de mensuração. Esta escala possui um zero absoluto e também a proporção entre os valores possui significado. Portanto é valido dizer que um determinado valor é o dobro do outro.

Tabela 3.2: Exemplo de variáveis ordinais.

Variável	**Significado**	**Valor**
Patente	Qual a sua patente?	[ ] Soldado [ ] Cabo [ ] Sargento [ ] Subtenente ou Suboficial [ ] Tenente [ ] Capitão ⋮ [ ] General
Grau de Instrução	Qual o seu grau de instrução?	[ ] Ensino básico [ ] Ensino fundamental [ ] Ensino médio [ ] Superior [ ] Especialista [ ] Mestrado [ ] Doutorado
Posição Hierárquica	Qual a sua posição hierárquica?	[ ] Funcionário Operacional [ ] Chefe de Serviço [ ] Supervisor [ ] Gerente [ ] Diretor [ ] Presidente ou Vice-Presidente [ ] Sócio

Tabela 3.3: Exemplo de variáveis intervalares.

Variável	**Significado**	**Valores**
Temperatura	Temperatura do paciente	[ ]^{0}C
Preocupação	Meu médico demonstra grande preocupação com meu estado de saúde.	1 ◯ ← Concordo plenamente 2 ◯ 3 ◯ 4 ◯ 5 ◯ ← Discordo totalmente

Algumas medidas físicas tais como massa, tamanho, energia são mensuradas nesta escala. A Tabela 3.4 oferece exemplos de variáveis desta escala. A Tabela 3.5 sumariza as operações que podem ser comumente executadas sobre valores de cada tipo de escala de mensuração.

Tabela 3.4: Exemplo de variáveis que assumem valores na escala de razão.

Variável	**Significado**	**Valor**
Idade	Qual a sua idade ?	[ ] anos
Altura	Qual a sua altura ?	[ ] m e [ ] cm
No. de Moradores	Quantas pessoas moram na sua residência ?	[ ] pessoas

Tabela 3.5: Operações matemáticas as escalas de mensuração.

Escala	**Operações**			
	Freqüência, Moda e $=$	**Mediana, Quartis e** $\leq$	$+, -$, **Média e Desvio Padrão**	$\times, /$, **e Coef. de Variação**
Nominal	☒			
Ordinal	☒	☒		
Intervalar	☒	☒	☒	
Razão	☒	☒	☒	☒

3.6 Transformação de Variáveis

A transformação de variáveis, também conhecida como normalização, é uma etapa importante na análise de cluster. É a transformação de variáveis que garante que as variáveis envolvidas no processo possam ser usadas corretamente no processo de clusterização, em sintonia com as particularidades de cada método, e que não exerçam uma influência indevida na formação dos cluster (EVERITT; LANDAU; LEESE, 2001).

É importante se observar que a transformação de variáveis discretas (sejam elas nominais ou ordinais) é diferente do processo de transformação de variáveis contínuas (sejam elas intervalares ou de razão).

3.6.1 Variáveis Contínuas

Existem várias propostas na literatura para normalização de variáveis contínuas, entre as técnicas mais conhecidas estão o *Z-score* (LARSEN; MARX, 1981) e o

mim-max cutoff (SCHAFFER; GREEN, 1996).

Na técnica *Z-score* o valor de cada observação é subtraído da média e dividido pelo desvio padrão isto é:

$$x_i{}' = \frac{(x_i - \bar{x})}{\sigma},$$

onde x_i, $x_i{}'$ são, respectivamente, o valor e o valor normalizado da i-ésima observação, e $\bar{x}$ e σ são, respectivamente, a média e o desvio padrão. Após a aplicação desta transformação o conjunto de observações normalizadas passa a ter média 0 e desvio padrão 1.

A técnica *min-max cutoff* utiliza uma abordagem um pouco diferente. Nesta técnica o valor de cada observação é subtraído do menor valor do conjunto de observações (o mínimo) e divido pela diferença entre o maior valor do conjunto (o máximo) e o mínimo. Em termos mais formais:

$$x_i{}' = \frac{(x_i - x_{min})}{x_{max} - x_{min}}.$$

onde x_i, $x_i{}'$ são, respectivamente, o valor e o valor normalizado da i-ésimo observação, e x_{min} e x_{max} são, respectivamente, o mínimo e o máximo. Observe que, de forma similar a técnica *Z-score*, o *min-max cutoff* também gera valores normalizados no intervalo que vai de 0 a 1.

3.6.2 Variáveis Discretas

A técnica mais difundia para transformação das variáveis discretas é a transformação com variáveis *dummy* (HAIR et al., 2005). Nesta técnica, uma variável que toma valor em um conjunto de n elementos é transformada em $n - 1$ variáveis *dummies*, onde os valores 0 e 1 indicam a presença de uma determinada categoria de valores. Vejam a Tabela 3.6 um exemplo deste tipo de transformação.

3.7 Medidas de Distância

Um conceito intrínseco da análise de cluster é a similaridade, ou seja, o conjunto de regras que servem como critério para agrupar ou separar observações. Um dos elementos para indicar similaridade mais conhecido é a distância entre os objetos. Várias medidas de distância são propostas na literatura, a seguir citamos algumas das mais populares (EVERITT; LANDAU; LEESE, 2001; DEER; EKLUND; NORMAN, 1996).

3.7.1 Distância Euclidiana

A medida de distância Euclidiana é provavelmente a mais comumente utilizada na análise de cluster. O cálculo da distância d entre duas observações $P = (p_1, p_2, \cdots, p_n)$ e $Q = (q_1, q_2, \cdots, q_n)$ é dada pela seguinte fórmula:

$$d_{PQ} = \left(\sum_{i=1}^{n} (p_i - q_i)^2 \right)^{\frac{1}{2}},$$

3.7.2 Distância City-Block on Manhattan

A distância City-Block, também conhecida como distância Manhattan, entre duas observações $P = (p_1, p_2, \cdots, p_n)$ e $Q = (q_1, q_2, \cdots, q_n)$ dada pela seguinte fórmula:

Tabela 3.6: Transformação com o auxílio de variáveis *dummies*.

Antes de Transformação

Variável	**Valores**
Estado Civil	Solteiro Casado Separado, Desquitado, Divorciado Viúvo

Depois da Transformação

	Variáveis			**Significado**
	$\boldsymbol{Dummy_1}$	$\boldsymbol{Dummy_2}$	$\boldsymbol{Dummy_3}$	
Valor	0	0	0	Solteiro
	0	0	1	Casado
	0	1	0	Separado ...
	1	0	0	Viúvo

$$d_{PQ} = \sum_{k=1}^{n} |p_i - q_i|,$$

3.7.3 Mahalanobis Distance

A distância de Mahalanobis para duas observações $P = (p_1, p_2, \cdots, p_n)$ e $Q = (q_1, q_2, \cdots, q_n)$ com matriz de covariância $\mathcal{C}$ é dada pela fórmula:

$$d_{PQ} = \sqrt{(P - Q)^T \mathcal{C}^{-1} (P - Q)}$$

onde $(P - Q) = (p_1 - q_1, p_2 - q_2, \cdots, p_n - q_n)$ e $\mathcal{C}^{-1}$ é a inversa da matriz de covariância.

A Tabela 3.7 exemplifica o cálculo da distância de Mahalanobis.

Tabela 3.7: Exemplo do cálculo da distância de Mahalanobis.

Valores dos Pontos

Pontos	**x**	**y**	**z**	**w**
P1	331,7	798,0	984,6	962,8
P2	666,6	900,0	748,2	803,4
P3	104,8	228,4	313,4	545,2
P4	503,4	549,9	588,8	120,6
mean	401,6250	619,0625	658,7500	608,0000
stddev	240,5303	299,0693	281,8623	367,6805

Valores dos Pontos após Normalização

Pontos	**x**	**y**	**z**	**w**
P1	-0,2907	0,5981	1,1561	0,9650
P2	1,1016	0,9394	0,3174	0,5314
P3	-1,2340	-1,3063	-1,2252	-0,1708
P4	0,4231	-0,2313	-0,2482	-1,3256

Matriz de Covariância

Covar	**x**	**y**	**z**	**w**
P1	0	0,593770648	0,355130404	-0,011301099
P2	0,593770648	0	0,661871866	0,401520522
P3	0,355130404	0,661871866	0	0,455616357
P4	-0,011301099	0,401520522	0,455616357	0

Matriz Inversa da Covariância

Inv Covar	10,71303237	2,729489905	-2,139688692	-12,3153965
	2,729489905	-0,161785389	0,210278646	-1,892477392
	-2,139688692	0,210278646	-0,238384826	3,557139607
	-12,3153965	-1,892477392	3,557139607	12,34843567

Distância entre os Pontos

P1	**P2**	**P3**	**P4**
0	6,956116949	3,207810036	11,27742257
6,956116949	0	6,257738946	17,11045964
3,207810036	6,257738946	0	9,618667618
11,27742257	17,11045964	9,618667618	0

4 ALGORITMO

Neste capitulo apresentamos a descrição formal de grafos Deolísticos e de um algoritmo de clusterização baseado nestes grafos. Entretanto algumas definições e conceitos precisam ser introduzidos antes que o algoritmo possa ser descrito em detalhes.

4.1 Teoria dos Grafos

Um grafo é definido como um grupo de objetos, chamados de vértices ou nós, conectados por linhas ou arcos chamados de arestas as quais podem ser associadas uma direção. Em geral grafos são conhecidos como um conjunto de vértices conectados por arestas (YELLEN, 2007). Outra definição comumente encontrada na literatura é apresentada a seguir por Jonathan e Jay (GROSS; YELLEN, 2003):

Definição 1 *Um gafo* $\boldsymbol{G}$ *é um par* $(\boldsymbol{V},\boldsymbol{E})$ *onde:*

- $\boldsymbol{V} = \{ v_1, v_2, \ldots, v_n \}$ *é um conjunto de elementos chamados de vértices, e*
- $\boldsymbol{E} = \{ e_1, e_2, \ldots, e_n \}$ *é um conjunto de elementos chamados de arestas ou arcos.*

Onde o cada elemento aresta $e_{1 \leq k \leq n}$ *é identificado por um par não ordenado* (v_i, v_j) □

Definição 2 *Dado um vértice* v_l *de um grafo* $\boldsymbol{G}(\boldsymbol{V},\boldsymbol{E})$*, uma aresta* $e_k = (v_i, v_j) \in \boldsymbol{E}$ *é incidente em* $v_l \in \boldsymbol{V}$ *se* $v_i = v_l$ *ou* $v_j = v_l$ □

Definição 3 *Um vértice v_l de um grafo $\boldsymbol{G(V,E)}$ possui um auto-ciclo se existe uma aresta $e_k = (v_i, v_j) \in \boldsymbol{E}$ tal que $v_i = v_l$ e $v_j = v_l$* □

Definição 4 *Dado um vértice v_i, o grau de v_i, $d(v_i)$, é o número de arestas incidentes em v_i, onde os auto-ciclos são contados duas vezes* □

Definição 5 *Dado um grafo $\boldsymbol{G}$, um caminho é definido como uma seqüência alternada de vértices e arestas em $\boldsymbol{G}$, começando e terminando em vértices, onde cada aresta é incidente aos vértices predecessores e sucessores a ela. Uma aresta não pode aparecer mais de uma vez em um caminho* □

Definição 6 *O tamanho de um caminho P, isto é, length(P), é o número de arestas que este caminho utiliza, contando múltiplas arestas e tantas vezes quanto elas aparecerem no caminho* □

Definição 7 *Um caminho fechado, é um caminho que começa e termina no mesmo vértice* □

Definição 8 *Um caminho simples é um caminho onde cada vértice aparece somente uma única vez* □

Definição 9 *Um caminho fechado, onde nenhum vértice aparece mais de uma vez, com exceção do primeiro e do ultimo, é chamado de circuito* □

Definição 10 *Um grafo $\boldsymbol{G}$ é dito conexo se existe pelo menos um caminho simples entre todos os pares de vértice em $\boldsymbol{G}$. Caso contrário ele é dito desconexo* □

Definição 11 *Uma árvore é um grafo conexo sem circuitos* □

Definição 12 *Uma árvore, onde um vértice chamado de raiz é diferenciado dos demais, é chamada de árvore com raiz* □

Definição 13 *Uma árvore binária é uma árvore com raiz, onde a raiz tem tem grau dois e os demais vértices possuem grau um ou três* □

Isto se deve ao fato de que uma árvore binária é um grafo acíclico conexo no qual entre quaisquer dois vértices existe um único caminho simples (BONDY; MURTY, 2008; DEO, 2004)

Definição 14 *Dado um vértice v_i de uma árvore com raiz $\boldsymbol{G}$ e um caminho simples P que começa na raiz da árvore e termina em v_i, o vértice pai de v_i é o vértice que imediatamente precede v_i em P, o vértice pai é também conhecido como nó pai, ancestral ou predecessor. Os vértices que imediatamente sucedem v_i em P são chamados de descendentes ou filhos de v_i. Podemos ter o caso em que v_i não possui nenhum descendente, neste caso ele é considerado um nó pendente ou folha* □

Definição 15 *Dado um vértice interno v_i de uma árvore binária $\boldsymbol{G}$, um dos vértices descendentes é arbitrariamente chamado de filho da esquerda, enquanto o outro é chamado de filho da direita. Neste caso left(v_i) e right(v_i) indicam os filhos da esquerda e direita de v_i respectivamente* □

Quando desenhamos grafos, adotamos a sensata convenção de desenhar v_i no centro, seu filho da esquerda a sua esquerda e seu filho da direita a sua direita.

4.2 Grafos Deolísticos

Para que possamos definir como os grafos Deolisticos são formados, precisamos apresentar formalmente dois conceitos:

- *variável* - um valor não determinado de um conjunto de valores possíveis, que representa uma propriedade de uma população de objetos, tal como: altura, peso, renda, escolaridade, cor, tamanho, etc., e
- *observação* - um conjunto de medidas de interesse em uma população de objetos, onde cada medida é uma instância de uma determinada variável.

Definição 16 *Um observação o é uma tupla $(a_1, a_2, \ldots, a_n)$, onde cada compomente a_i pode ter de assumir valor em um domínio específico. Se $o = (a_1,\ a_2,\ \ldots,\ a_n)$ é uma observação, então $o[i]$ é o valor do componente a_i de o, para $i \in [1\,..\,n]$.*

Definição 17 *Dado um conjunto de observações* $\boldsymbol{O} = \{ o_1, o_2, \ldots, o_m \}$, *tal que cada* $o_i \in \boldsymbol{O}$ *é uma tupla* $(a_1, a_2, \ldots, a_n)$, *uma variável* w_i *em* $\boldsymbol{O}$ *assume um valor no conjunto* $\{o_1[i], o_2[i], \ldots, o_m[i]\}$, *para* $i \in [1 \,..\, n]$.

De acordo com Narsingh Deo (DEO, 2004) cada vértice de uma árvore binária representa um teste com duas respostas possíveis. Iniciando-se na raiz, a saída produzida pelo teste vai indicar em qual vértice descendente será realizado um novo teste. Este processo se repete até que um vértice pendente seja alcançado. Sendo assim, qualquer afirmação que possa ser feita a respeito de um vértice pendente como resultado do processo de teste, revela uma propriedade daquele vértice em particular. Baseado nas idéias de Deo chegamos à definição de Grafos Deolísticos.

Definição 18 *Dado um sistema lógico* $\boldsymbol{L}$ *e um conjunto de observações* $\boldsymbol{O} = \{ o_1, o_2, \ldots, o_m \}$, *tal que acada* o_i *em* $\boldsymbol{O}$ *é uma tupla* $(a_1, a_2, \ldots, a_n)$, *um grafo Deolístico* $\boldsymbol{G} = (\boldsymbol{V}, \boldsymbol{E})$ *é uma árvore binária na qual*

1. *cada vértice* v_i *em* $\boldsymbol{V}$ *possui associado um conjunto de observações* $\boldsymbol{O}_i \subseteq \boldsymbol{O}$,

2. *cada vértice interno* v_k *em* $\boldsymbol{V}$ *possui associado uma sentença bem definida* l_k *em* $\boldsymbol{L}$ *envolvendo as variáveis em* $\boldsymbol{O}_k$, *tal que:*

   (a) *o filho a esquerda de* v_k, *isto é* $left(v_k)$, *possui as observações de* v_k *onde a sentença* l_k *é verdadeira,*

   (b) *o filho a direita de* v_k, *isto é* $right(v_k)$, *possui as observações de* v_k *onde a sentença* $\neg l_k$ *é verdadeira e*

   (c) $l_k \wedge \neg l_k$ *é falso.*

Definição 19 *Um grafo Deolístico baseado em predicados é um grafo Deolístico que possui a lógica de predicados como sistema lógico. Um grafo Deolístico baseado em predicados restrito é um grafo onde os predicados associados a cada vértice possuem a seguinte forma:*

- $w_i \leq r$, *se* w_i *é uma variável continua, ou*

- $w_i = k$, *se* w_i *é uma variável categórica,*

onde w_i *é uma variável em* $\boldsymbol{O}$*, o conjunto de observações, e* r *e* k *são constantes ou variáveis de um tipo apropriado.*

A Figura 1 apresenta um grafo Deolístico de predicados restrito construído com dados apresentados por Witten e Frank (WITTEN; FRANK, 2005) a respeito de condições atmosféricas favoráveis à pratica do *golf*. É importante observar que os grafos Deolísticos não são árvores de classificação, uma classe de algoritmos de mineração de dados muito conhecida (ROKACH; MAIMON, 2005). Enquanto as árvores de classificação requerem a existência de uma variável *target*[1], os grafos Deolísticos não requerem a existência deste tipo de variável.

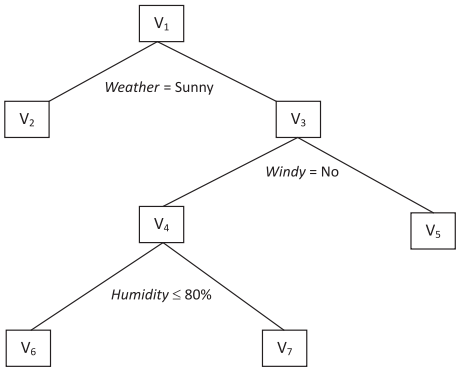


Figura 4.1: Grafo Deolísticos de predicados restritos.

As propriedades abaixo pertencentes aos grafos Deolísticos de predicados derivam diretamente de suas definições.

Propiedade 1 *Dados um grafo Deolístico* $\boldsymbol{G}$ *e um caminho* P *que começa na raiz de* $\boldsymbol{G}$*, se* v_i *imediatamente precede* v_j *em* P*, então o conjunto de observações associadas a* v_i *contém as observações associadas a* v_j*, i.e.,* $\boldsymbol{O}_i \supseteq \boldsymbol{O}_j$*.*

Prova: *Seja* l_i *o predicado associado a* v_i*. Se* v_i *precede* v_j *em* P*, então ou* v_j *é o filho esquerdo ou direito de* v_i*. Se* v_j *é o filho esquerdo, então* v_j *contém tão e somente os objetos em* $\boldsymbol{O}_i$ *para os quais* l_i *é*

[1]Variável *target* é aquela cujo valor vai ser modelado, predito ou explicado por outras variáveis

verdadeiro. Por outro lado, se v_j é o filho da direita, então v_j contém tão e somente os objetos em $\boldsymbol{O}_i$ para os quais $\neg l_i$ é verdadeiro. Portanto, $\boldsymbol{O}_i \supseteq \boldsymbol{O}_j$ □

Propiedade 2 *Dado um grafo Deolístico* $\boldsymbol{G}$*=(*$\boldsymbol{V}$*,*$\boldsymbol{E}$*) e um caminho* $P = v_1 \overset{e_1}{\frown} v_2 \overset{e_2}{\frown} v_3 \ldots v_n$*, começando na raiz de* $\boldsymbol{G}$*, tal que* $n \geq 2$*, então o conjunto de observações associados a* v_1 *contém as observações associadas a* v_2 *que contém o conjunto de observações associadas a* $v_3 \ldots$*, i.e.* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_2 \ldots \boldsymbol{O}_n$*.*

Prova: *Seja Q um sub-caminho de P, tal que* $Q = v_1 \overset{e_1}{\frown} v_2 \ldots v_k$*, onde 1* $\leq k \leq n$*. Note que se k = 1, então, obviamente,* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_1$*. Se k = 2, então* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_2$*, de acordo com a Proposição 1. Agora, suponha que* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_2 \ldots \boldsymbol{O}_k$ *é verdadeiro, para algum* $k \geq 2$*. Para* $k+1$*,* $\boldsymbol{O}_k \supseteq \boldsymbol{O}_{k+1}$*, de acordo com a Proposição 1, uma vez que k predece imediatamente k+1 em Q. Já que* $\supseteq$ *é uma relação transitiva, segue que* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_2 \ldots \boldsymbol{O}_{k+1}$*. Portanto, por indução* $\boldsymbol{O}_1 \supseteq \boldsymbol{O}_2 \ldots \boldsymbol{O}_n$ □

Propiedade 3 *Dados um grafo Deolístico* $\boldsymbol{G}$*=(*$\boldsymbol{V}$*,*$\boldsymbol{E}$*) e um caminho* $P = v_1 \overset{e_1}{\frown} v_2 \overset{e_2}{\frown} v_3 \ldots v_n$*, começando na raiz de* $\boldsymbol{G}$*, tal que* $n \geq 2$*, então a conjunção* $q_1 \wedge q_2 \ldots q_{n-1}$ *pertence a* $\boldsymbol{O}_n$*, as observações associadas a* v_n*, onde* q_i *é* l_i *ou* q_i *é* $\neg l_i$*, para* $i \in [1..n{-}1]$*, e* $l_1, l_2 \ldots l_{n-1}$ *são as sentenças associadas a* $v_1, v_2 \ldots v_{n-1}$ *respectivamente.*

Prova: *Seja Q um subcaminho de P, tal que* $Q = v_1 \overset{e_1}{\frown} v_2 \ldots v_k$*, onde* $k \geq 2$*. Note que se k = 2, então, de acordo com a Definição 18,* l_1 *ou* $\neg l_1$ *é verdadeiro para* $\boldsymbol{O}_2$*. Portanto, existe um* q_1 *para o qual* $\boldsymbol{O}_2$ *é verdadeiro.*

Agora, suponha que para algum $k > 2$*,* $q_1 \wedge q_2 \wedge q_3 \ldots q_{k-1}$ *é verdadeiro para* $\boldsymbol{O}_k$*. De acordo com a Definição 18,* l_k *ou* $\neg l_k$ *é verdadeiro para* $\boldsymbol{O}_{k+1}$*. Portanto, existe um* q_k *que é verdadeiro para* $\boldsymbol{O}_{k+1}$*. Entretanto a Definição 18 nos diz que,* $\boldsymbol{O}_k \supseteq \boldsymbol{O}_{k+1}$*. Se* $q_1 \wedge q_2 \wedge q_3 \ldots q_{k-1}$ *é verdadeiro para* $\boldsymbol{O}_k$*, ele é verdadeiro para qualquer subconjunto de* $\boldsymbol{O}_k$*, em particular para* $\boldsymbol{O}_{k+1}$*. Consequente-*

mente, como tanto q_k quanto $q_1 \wedge q_2 \wedge q_3 \ldots q_{k-1}$ são verdadeiros para $\boldsymbol{O}_{k+1}$, $q_1 \wedge q_2 \wedge q_3 \ldots q_{k-1} \wedge q_k$ é verdadeiro para $\boldsymbol{O}_{k+1}$. Por indução finita, $q_1 \wedge q_2 \wedge q_3 \wedge \cdots \wedge q_{n-1}$ é verdadeiro para $\boldsymbol{O}_n$ $\square$

4.3 Clusterizando com Grafos Deolísticos

Definição 20 *Dado duas observações o_i e o_j, seja $d(o_i, o_j)$ uma métrica que indica quão similar o_i e o_j são. A matriz de similaridades $\boldsymbol{M}$ de um conjunto de observações $\boldsymbol{O} = \{o_1, o_2, \ldots, o_m\}$ é uma matriz $m \times m$ tal que $M(\boldsymbol{O})(i,j) = d(o_i, o_j)$, para $i, j \in [1 \,..\, m]$.*

Definição 21 *Para um conjunto de observações$\boldsymbol{O} = \{\ o_1,\ o_2,\ \ldots,\ o_m\ \}$ e sua correspondente matriz de similaridades M($\boldsymbol{O}$), seja $\lambda(M(\boldsymbol{O})) \geq 0$ uma métrica que indica quão diversos são os elementos de M($\boldsymbol{O}$). Quanto menor o valor de $\lambda(M(\boldsymbol{O}))$, mais similares são os objetos em $\boldsymbol{O}$.*

Dadas estas definições, o Algoritmo 1 adiciona exatamente um nó a um grafo Deolístico, o Algoritmo 2 constrói um grafo Deolístico por completo e o Algoritmo 3 poda recursivamente um grafo Deolístico até que um número de vértices pendentes seja obtido.

Algoritmo 1 *Dado um vértice v_i de um grafo Deolistic $\boldsymbol{G}$, o conjunto de observações $\boldsymbol{O}_i = \{o_1, o_2, \ldots, o_m\}$ associadas a este vértice, e uma taxa mínima aceitável de redução de diversidade $\xi > 0$*

1. *Procure a sentença l que maximiza a equação*

$$\beta = \lambda(M(\boldsymbol{O}_i)) \times \frac{|\boldsymbol{O}_i|}{|\boldsymbol{O}|} - \lambda(M(\boldsymbol{O}_{left(v_i)})) \times \frac{|\boldsymbol{O}_{left(v_i)}|}{|\boldsymbol{O}|}$$

$$-\lambda(M(\boldsymbol{O}_{right(v_i)})) \times \frac{|\boldsymbol{O}_{right(v_i)}|}{|\boldsymbol{O}|},$$

sujeito à

$$\lambda(M(\boldsymbol{O}_{left(v_i)})), \lambda(M(\boldsymbol{O}_{right(v_i)})) \leq \lambda(M(\boldsymbol{O}_i))$$

onde $\boldsymbol{O}_{left(v_i)}$ *e* $\boldsymbol{O}_{right(v_i)}$ *é o conjunto de observações a ser associado aos filhos da esquerda e da direita de* v_i *respectivamente, se* l *é associado a* v_i*. Note que* β *é a diferença ponderada entre as medidas de diversidade dos objetos associados a* v_i *e as medidas de diversidade dos objetos associados a seus filhos.*

2. *If* $\beta \geq \xi$ *then*

   (a) *Associe* l *a* v_i*,*

   (b) *Associe* β *a* v_i*,*

   (c) *Crie o filho da esquerda de* v_i*, isto é,* $v_{left(v_i)}$*, e conecte-o à* v_i*,*

   (d) *Associe as observações em* $\boldsymbol{O}_i$ *para as quais* l *é verdadeiro à* $v_{left(v_i)}$ *.*

   (e) *Crie o filho da direita de* v_i*, isto é,* $v_{right(v_i)}$*, é conecte-o à* v_i*,*

   (f) *Associe as observações de* $\boldsymbol{O}_i$ *para as quais* $\neg l$ *é verdadeiro à* $v_{right(v_i)}$*.*

Algoritmo 2 *Dado um grafo* $\boldsymbol{G}(\boldsymbol{V}, \boldsymbol{E})$*, tal que* $|\boldsymbol{V}| = 1$ *e* $\boldsymbol{E} = \{\}$*, um conjunto não vazio de observações* $\boldsymbol{O} = \{o_1, o_2, \ldots, o_n\}$ *e um taxa mínima de redução de diversidade* $\xi > 0$

1. *Seja* v *em* V *o nó raiz de* $\boldsymbol{G}$*,*

2. *Associe* $\boldsymbol{O}$ *ao vértice raiz* v*,*

3. *Aplique os procedimentos descritos no Algoritmo 1 para* v *e recursivamente para cada um de seus filhos.*

Algoritmo 3 *Dado um grafo Deolístico* $\boldsymbol{G}(\boldsymbol{V}, \boldsymbol{E})$ *e o número máximo de segmentos desejados* $\eta > 0$

1. *Seja* $\rho(\boldsymbol{G})$ *o conjunto de nós pendentes em* G*,*

2. *Enquanto* $|\rho(\boldsymbol{G})| > \eta$

   (a) *Seja* $\phi(\boldsymbol{G}) = \{v \in \boldsymbol{V} \wedge v \notin \rho(\boldsymbol{G}) \,|\, left(v) \in \rho(\boldsymbol{G}) \wedge right(v) \in \rho(\boldsymbol{G})\}$,

   (b) *Seja* v_i *o vertice em* $\phi(\boldsymbol{G})$ *que possui o menor valor de* β *associado,*

   (c) *Remova o filho da esquerda de* v_i*, isto é,* $left(v_i)$*, de* $\boldsymbol{G}$ *e o vértice que conecta* v_i *a* $left(v_i)$*,*

   (d) *Remova o filho da direita de* v_i*, isto é,* $right(v_i)$*, de* $\boldsymbol{G}$ *e vértice que conecta* v_i *a* $right(v_i)$*,*

   (e) *Desassocie* l_i*, a sentença lógica associada a* v_i*, de* v_i*.*

Ao longo deste trabalho, os grafos Deolísticos de predicados restrito é utilizado em todos os processos de clusterização e a distância de Mahalanobis é a métrica d usada para medir a similaridade entre objetos, uma vez que ela lida bem com os casos em que as variáveis envolvidas na clusterização estejam fortemente correlacionadas. Consulte (EVERITT; LANDAU; LEESE, 2001) para uma discussão sobre distância de Mahalanobis e correlação. Além disso, a distância média entre os membros de um mesmo cluster é escolhida como a métrica λ que buscamos minimizar.

Nestas circunstâncias, ainda que não exista uma regra única para identificar o melhor valor de η (número de máximo de segmentos), Marchi (MARCHI, 2007) sugere um processo de poda baseado nas idéias de (MOJENA, 1977) e (MILLIGAN; COOPER, 1985), similar ao Algoritmo 3, que busca reduzir o tamanho da árvore gerada pelo algoritmo Deolístico. Segundo Marchi *op. cit.* o tamanho ideal da árvore gerada por um grafo Deolistico é aquele no qual β excede imediatamente

$$\mu_\beta - \tau \times \sigma_\beta,$$

onde μ_β é a media das diminuições de diversidade associadas aos diversos nós internos da árvores, σ_β é o desvio padrão correspondente e τ é um valor escolhido arbitrariamente entre 1,25 e 3,50. Em adição, Marchi *op. cit.* oferece algumas sugestões de como determinar o melhor valor de τ.

5 O MÉTODO

De acordo com Albert Schweitzer (1875-1965), ganhador do premio nobel da paz de 1952:

> *"Exemplo não é a principal forma de influenciar os outros, é a única forma".*

Em conseqüência, o método proposto neste trabalho é introduzido, passo a passo, com a ajuda de um exemplo inspirado no mundo real[1].

5.1 Passo 0: Contextualizando

Considere um empresa como a Accenture, Tata Consulting Servies(TCS), Electronic Data Systems (EDS) e outras que executam um grande número de projetos de desenvolvimento de *software* anualmente, em vários locais do mundo[2]. No corpo deste trabalho essa empresa será chamada de $\Delta\Sigma$ CORP.

A competição acirrada por contratos de desenvolvimento de *software* que marca os dias de hoje faz com que essas empresas sejam obrigadas a tomar ações que aumentem as chances de sucesso de seus projetos. Já que a performance de um projeto depende substancialmente da percepção dos clientes e de como os gerentes dos projetos lidam com os eventos, condições e elementos que mais podem afetar as chances

[1] Por exemplo inspirado no mundo real quer-se dizer que os dados que aparecem nas tabelas e gráficos ao longo deste trabalho poderiam muito bem terem sido obtidos de projetos do mundo real.

[2] Accenture é uma empresa Americana de prestação de serviços em tecnologia e *outsourcing* (tercerização). TCS é uma empresa Indiana de tecnologia da informação. EDS é uma empresa Americana de tecnologia da informação e consultoria em processos de negócio.

de sucesso do projeto, ou seja, os fatores de risco (SLOVIC, 2000), $\Delta\Sigma$ CORP decidiu criar uma base de conhecimento com informações sobre os projetos finalizados ao longo dos últimos doze meses, nas localidades onde faz terceirização de serviços

5.2 Passo 1: Coletando as informações

Para identificar os riscos a que os projetos de *software* estavam sujeitos a $\Delta\Sigma$ CORP decidiu usar a primeira parte do questionário de Linda Wallace *et al.* para entrevistar os gerentes destes projetos. Em adição, para avaliar a performance dos projetos, a $\Delta\Sigma$ CORP decidiu usar a segunda parte do questionário de Linda Wallace *et al.* para entrevistar as organizações clientes dos projetos que havia desenvolvido. Veja a Figura 2.9 para uma descrição do questionário de Linda Wallace *et al.*. A Tabela 5.1 contém os dados coletados pela $\Delta\Sigma$ CORP.

Tabela 5.1: Dados coletados pela $\Delta\Sigma$ CORP.

Projeto	**Factores de Risco**			**Fatores de Performance do Projeto**			
	$\mathbf{RF_1}$	$\cdots$	$\mathbf{RF_{44}}$	$\mathbf{PPF_1}$	$\mathbf{PPF_2}$	$\cdots$	$\mathbf{PPF_9}$
P_1	4	$\cdots$	3	1	4	$\cdots$	2
P_2	6	$\cdots$	5	2	1	$\cdots$	3
P_3	5	$\cdots$	6	3	5	$\cdots$	1
P_4	7	$\cdots$	7	6	2	$\cdots$	3
P_5	3	$\cdots$	4	5	4	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
P_{299}	7	$\cdots$	4	1	3	$\cdots$	3
P_{300}	5	$\cdots$	2	3	6	$\cdots$	1

5.3 Passo 2: Preparando os dados para análise

Para evitar que os gerentes de projetos tenham que analisar um grande número de variáveis, os dados na Tabela 5.1 são resumidos em dimensões de risco e performance (WALLACE; KEIL; RAI, 2004). Neste caso, o valor de uma dimensão é a média dos valores dos fatores riscos e de performance associados a ela. A Tabela 5.2 apresenta a fórmula que relaciona fatores de risco e de performance com suas respectivas dimensões de risco e performance.

A Tabela 5.3 apresenta dos dados da Tabela 5.1 resumidos em suas respectivas

dimensões utilizando as fórmulas da Tabela 5.2.

5.4 Passo 3: Analisando os dados

Em seguida, os dados na Tabela 5.3 são submetidos ao algoritmo de clusterização de grafos Deolístico descrito na Seção 4.3. A Figura 5.4 mostra o diagrama em forma de árvore gerado pelo algoritmo, obtido com o suporte de um *software* desenvolvido pelos autores do artigo.

O nó IN_1, no topo do diagrama, isto é, a raiz da árvore, indica que 300 observações foram usadas no processo de clusterização e que a variância entre as distâncias destas observações é 7,2. Deve-se notar que o diagrama em forma de árvore contém dois tipos diferentes de nós: os nós internos (INs), que sempre possuem dois nós descendentes, e os nós terminais (TNs), que não possuem nós descendentes.

Tabela 5.2: Relacionando fatores à dimensões.

Dimensão de Risco	**Fórmula**
RD_1: Equipe	$RD_1 = (RF_1 + RF_2 + \cdots + RF_7)/7$
RD_2: Ambiente Organizacional	$RD_2 = (RF_8 + RF_9 + \cdots + RF_{13})/6$
RD_3: Requisitos	$RD_3 = (RF_{14} + RF_{15} + \cdots + RF_{21})/8$
RD_4: Planejamento e Controle	$RD_4 = (RF_{22} + RF_{23} + \cdots + RF_{30})/9$
RD_5: Usuário	$RD_5 = (RF_{31} + RF_{32} + \cdots + RF_{36})/6$
RD_6: Complexidade	$RD_6 = (RF_{37} + RF_{38} + \cdots + RF_{44})/8$

Dimensão de Performance	**Fórmula**
PD_1: Técnica	$PD_1 = (PPF_1 + PPF_2 + PPF_3)/2$
PD_2: Usuário	$PD_2 = (PPF_4 + PPF_4 + \cdots + PPF_7)/5$
PD_3: Custo	$PD_3 = PPF_8$
PD_4: Prazo	$PD_4 = PPF_9$

Tabela 5.3: Dados resumidos em dimensões.

Projeto	**Dimensão de Risco**				**Dimensão de Performance**			
	$\mathbf{RD_1}$	$\mathbf{RD_2}$	$\cdots$	$\mathbf{RD_6}$	$\mathbf{PD_1}$	$\mathbf{PD_2}$	$\mathbf{PD_3}$	$\mathbf{PD_4}$
P_1	5,2	1,8	$\cdots$	2,8	3,4	5,3	4,7	1,9
P_2	4,9	6,7	$\cdots$	4,7	3,9	2,1	3,8	4,4
P_3	6,2	2,8	$\cdots$	5,7	2,7	4,7	5,3	2,5
P_4	2,7	4,2	$\cdots$	5,6	6,1	3,7	2,7	4,8
P_5	1,3	5,9	$\cdots$	5,1	6,0	2,9	3,4	3,8
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
P_{299}	6,3	3,5	1,7	3,8	6,1	5,2	2,5	4,1
P_{300}	2,1	5,1	2,1	5,5	4,7	4,6	2,9	2,2

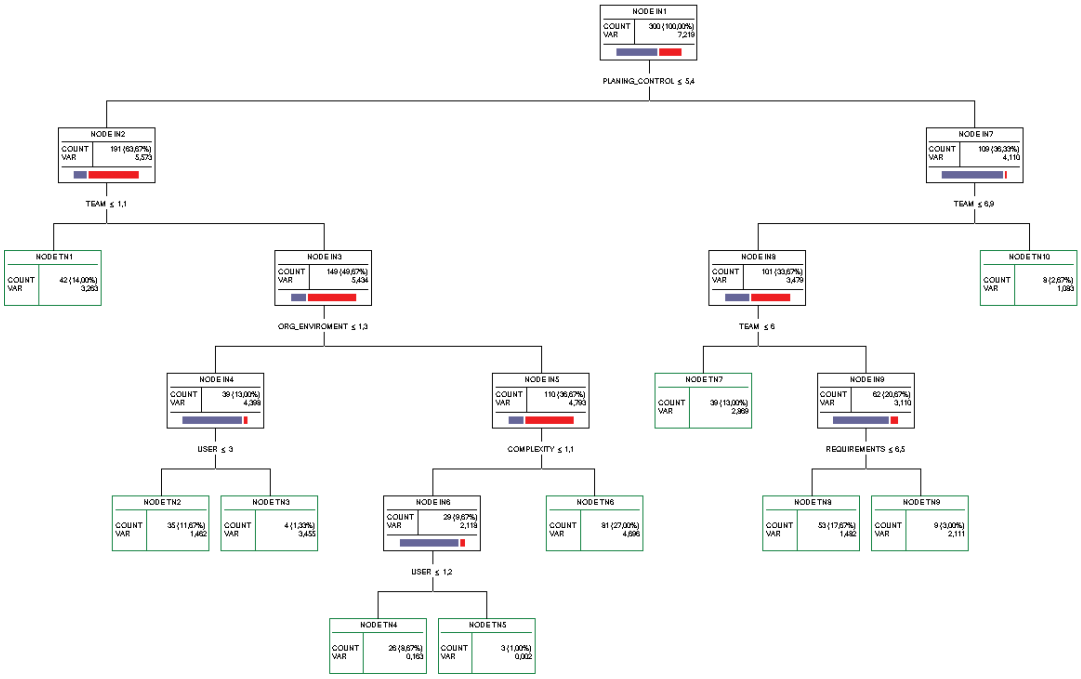


Figura 5.1: Diagrama produzido pelo algoritmo de grafos Deolisticos.

O predicado *Planejamento e Controle* $\leq$ 5,4 imediatamente abaixo do nó IN_1 (nó interno 1) indica a primeira divisão a que os dados coletados pela $\Delta\Sigma$ CORP foram submetidos. Em decorrência, IN_2 (nó terminal 2) contém as 191 observações nas quais o valor da dimensão de risco *Planejamento e Controle* é menor ou igual a 5,4 e IN_7 (nó interno 7) contém as 109 observações nas quais *Planejamento e Controle* é maior do que 5,4.

O fato IN_2 ser um nó interno é um indicativo de que as observações neste nó são divididas em dois novos conjuntos. O predicado logo abaixo IN_2 indica como essa divisão é feita. Portanto, o primeiro conjunto contém observações nas quais o valor da dimensão de risco *Equipe* é menor ou igual a 1,1 e o segundo contém o restante das observações. Em decorrência, as observações que pertencem a TN_1 são aquelas onde *Planejamento e Controle* $\leq$ 5,4 $\wedge$ *Equipe* $\leq$ 1,1 e as observações em IN_3 são aquelas onde *Planejamento e Controle* $\leq$ 5,4 $\wedge$ *Equipe* $>$ 1,1.

Seguindo esta linha de raciocínio, cada caminho partindo da raíz da árvore tanto para um nó interno quanto para um terminal produz uma regra, na forma de uma conjunção de predicados envolvendo variáveis de dimensão de risco, indicando o que distingue as observações pertencentes a um nó de todos os outros nós. A Tabela

5.4 mostra todas as regras envolvendo nós terminais que podem ser derivadas do diagrama em árvore apresentado na Figura 5.4.

Tabela 5.4: Regras que distinguem as observações de um nó terminal.

Nó	**Regra**
TN_1	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $\leq$ 1,1
TN_2	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $>$ 1,1 $\wedge$ Ambiente Organizacional $\leq$ 1,3 $\wedge$ Usuário $\leq$ 3
TN_3	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $>$ 1,1 $\wedge$ Ambiente Organizacional $\leq$ 1,3 $\wedge$ Usuário $>$ 3
TN_4	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $>$ 1,1 $\wedge$ Ambiente Organizacional $>$ 1,3 $\wedge$ Usuário $\leq$ 1,2
TN_5	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $>$ 1,1 $\wedge$ Ambiente Organizacional $>$ 1,3 $\wedge$ Complexidade $\leq$ 1,1 and Usuário $>$ 1,2
TN_6	Planejamento e Controle $\leq$ 5,4 $\wedge$ Equipe $>$ 1,1 $\wedge$ Ambiente Organizacional $>$ 1,3 $\wedge$ Complexidade $>$ 1,1
TN_7	Planejamento e Controle $>$ 5,4 $\wedge$ Equipe $\leq$ 6,9 $\wedge$ Equipe $\leq$ 6
TN_8	Planejamento e Controle $>$ 5,4 $\wedge$ Equipe $\leq$ 6,9 $\wedge$ Equipe $>$ 6 and Requisitos $\leq$ 6,5
TN_9	Planejamento e Controle $>$ 5,4 $\wedge$ Equipe $\leq$ 6,9 $\wedge$ Equipe $>$ 6 and Requisitos $>$ 6,5
TN_{10}	Planejamento e Controle $>$ 5,4 e Equipe $>$ 6,9

5.5 Passo 4: Conclusões

Obviamente, tudo isso não seria muito útil se não fosse possível relacionar as observações feitas em um nó terminal com as dimensões de performance e assim tirar conclusões desta relação. Espera-se que, ao examinarmos as observações em cada nó terminal, seja possível estabelecer esse tipo de relação. A Tabela 5.5 apresenta as estatísticas relevantes em todos os nós terminais apresentados na Figura 5.4.

Tabela 5.5: Estatísticas das dimensões de performance por nó terminal.

			Dimensões de Performance de Projeto							
Nó	**Observações**		**Técnica**		**Usuário**		**Custo**		**Tempo**	
Terminal	**Qtd**	**%**	**Média**	**D.P.**	**Média**	**D.P.**	**Média**	**D.P.**	**Média**	**D.P.**
TN_1	42	14,0%	4,9	0,84	5,7	0,87	6,1	1,16	6,1	1,15
TN_2	35	11,7%	5,6	1,03	6,2	0,37	6,4	0,55	6,4	0,55
TN_3	4	1,3%	5,1	1,18	3,5	0,38	4,8	0,50	4,0	1,41
TN_4	26	8,8%	6,7	0,81	6,8	0,59	6,6	0,86	6,5	1,03
TN_5	3	1,0%	5,3	0,76	3,9	0,81	4,3	0,58	4,3	0,58
TN_6	81	27,0%	5,3	0,90	5,0	1,17	5,1	1,00	5,1	1,05
TN_7	39	13,0%	3,9	1,24	3,8	1,26	3,7	1,37	4,0	1,12
TN_8	53	17,7%	3,8	1,08	3,6	1,20	3,8	1,27	3,8	1,34
TN_9	9	3,0%	1,3	0,66	1,4	0,69	1,4	0,53	1,6	0,54
TN_{10}	8	2,7%	1,0	0,01	1,1	0,35	1,2	0,05	1,4	0,03
Total	300	100,0%								

Antes de prosseguir, é importante mencionar que devido à forma como as declarações no questionário de Linda Wallace *et al.* foram concebidas, um fator de risco

com uma pontuação pequena indica uma pequena chance daquele fator em particular possuir um efeito negativo no projeto. Por outro lado, uma dimensão de performance com pontuação alta indica uma grande satisfação da organização cliente para com um determinado aspecto do projeto.

Por exemplo, de acordo com a Tabela 5.4, TN_1 contém os projetos nos quais *Planejamento e Controle* $\leq 5.4 \wedge$ *Equipe* ≤ 1.1. Não chega portanto a surpreender que a Tabela 5.5 indique alta satisfação do cliente com os resultados gerados por estes projetos na percepção técnica, percepção de usuário, custo e tempo, com médias de 4.9, 5.7, 6.1 e 6.1 respectivamente.

Contudo, projetos no nó TN_{10} vão pelo caminho contrário. De acordo com a Tabela 5.4 os projetos neste nó são aqueles que *Planejamento e Controle* $> 5{,}4 \wedge$ *Equipe* $> 6{,}9$, isto é, pontuaram perigosamente alto na dimensão de planejamento e controle, e também na dimensão de gerenciamento de equipe. Conseqüentemente, de acordo com a informação disposta na Tabela IV, esses projetos pontuaram igualmente mal em todas as dimensões de performance, com média de 1,0, 1,1, 1,2 e 1,4 respectivamente.

5.6 Passo 5: Tomando ações corretivas

Como resultado da análise feita no Passo 3, $\Delta\Sigma$ CORP sentiu-se compelida a agir. Inicialmente ela estabeleceu um nível mínimo de satisfação do cliente no qual todos os projetos feitos pela organização deve-se espelhar. Por conta da pressão exercida pela concorrência, a alta gerência concordou que um nível mínimo de satisfação do cliente (medidos pelas dimensões de performance de projeto) teria que estar acima da média, sendo esta arbitrariamente estabelecida em 4,0 para todas as dimensões de performance compiladas por Linda Wallace *et al.*. Ver Figura 2.9.

Para ter certeza de que este nível de serviço será alcançado e superado, os projetos em curso serão auditados regularmente. Se o diagrama na Figura 5.4 indica que um projeto em particular tende a ficar abaixo do nível mínimo aceitável, o gerente de projeto é responsável por tomar as ações corretivas. Um fundo adicional comum a

todos os projetos será disponibilizado para dar suporte a estas ações. Além disso, dado que as informações contidas na Figura 5.4 tendem a ficar desatualizadas com o tempo, novas entrevistas devem ser conduzidas regularmente e novos diagramas serão produzidos como resultado.

5.7 Sumarização do Método

O método que acabamos de apresentar nos permite aumentar as chances de sucesso de projetos de *software*. Nestes sentido, as organizações que desejam aumentar a percepção que seus clientes possuem da performance dos projetos de *software* que executam, podem se beneficiar substancialmente das seguinte ações:

0) *Contextualizando* - Estabeleça um período de tempo para coletar informações sobre projetos de *software*.

1) *Coletando as informações*

   (a) Utilize a primeira parte do questionário de Linda Wallace *et al.* para entrevistar os gerentes de projeto sobre os projetos de *software* que foram concluídos ao longo do período especificado.

   (b) Utilize a segunda parte do questionário para entrevistar os clientes destes projetos.

2) *Preparando os dados para análise* - Sumarize os dados coletados junto aos gerentes de projetos e clientes em dimensões de risco e performance.

3) *Analisando os dados* - Use o algoritmo Deolistico descrito na Section 4.3 para explicitar a relação entre as dimensões de risco e as de performance.

4) *Tirando conclusões* - Analise os clusters criados pelos Grafos Deolisticos usando as regras e estatísticas geradas pelo algoritmo.

5) *Tomando ações corretivas*

(a) Estabeleça um critério mínimo aceitável para a performance dos projetos que deve ser superado por todos os projetos de *software*.

(b) Auditore regularmente nos projetos de software em andamento.

(c) Os projetos que tenham probabilidade a não atingir os critérios mínimos de performance devem receber recursos extras para que ações corretivas possam ser realizadas.

(d) Utilize os dados coletados nas auditorias para realimentar o algoritmo Deolistico.

6 A FERRAMENTA

Para o desenvolvimento da ferramenta computacional que implementa o algoritmo Deolístico foi utilizada a linguagem de programação Delphi 7 da Borland Software Corporation (www.borland.com) (CANTU, 2003). A implementação possui cerca de 20.000 linhas de código. Esta ferramenta foi chamada de *Inteligent Segmentation Graphs*, ou ISG. As principais características da implementação estão descritas na Tabela 6.1

Tabela 6.1: Principais características da ferramenta.

Características	**Descrição**
Formato de leitura dos dados	Arquivo texto separado por ponto e vírgula. A primeira linha contém o nome das variáveis.
Número máximo de variáveis	Até 200 variáveis.
Número máximo de observações	Ilimitado. Testado até 10.000 observações para uma máquina com 1Gb de memória.
Sistema operacional	Windows 98, 2000, XP e Vista.
Variáveis Discretas	Transformadas automaticamente pelo programa em um conjunto de variáveis binárias. Por exemplo, uma variável discreta que possa assumir três valores é transformada em três variáveis binárias.
Variáveis Contínuas	São submetidas a um processo de normalização que transformam os valores apresentados em um novo conjunto de valores com média zero e desvio padrão 1. Este processo é conhecido na literatura por "transformação Z-Score" (HAIR et al., 2005).
Saída	Desenho gráfico da árvore que representa as relações existentes entre os diversos segmentos. Regras na forma de predicados, que diferenciam os diversos segmentos selecionados pelo usuário.

6.1 Apresentando a Tela Inicial

Uma tela de apresentação da ferramenta aparece toda vez que ela é utilizada. Essa tela permite que o usuário possa rapidamente abrir um arquivo de dados, assim como ver a lista de arquivos abertos recentemente. Caso o usuário não deseje que a esta tela seja exibida, ele pode marcar a opção *Don't show this window again.*

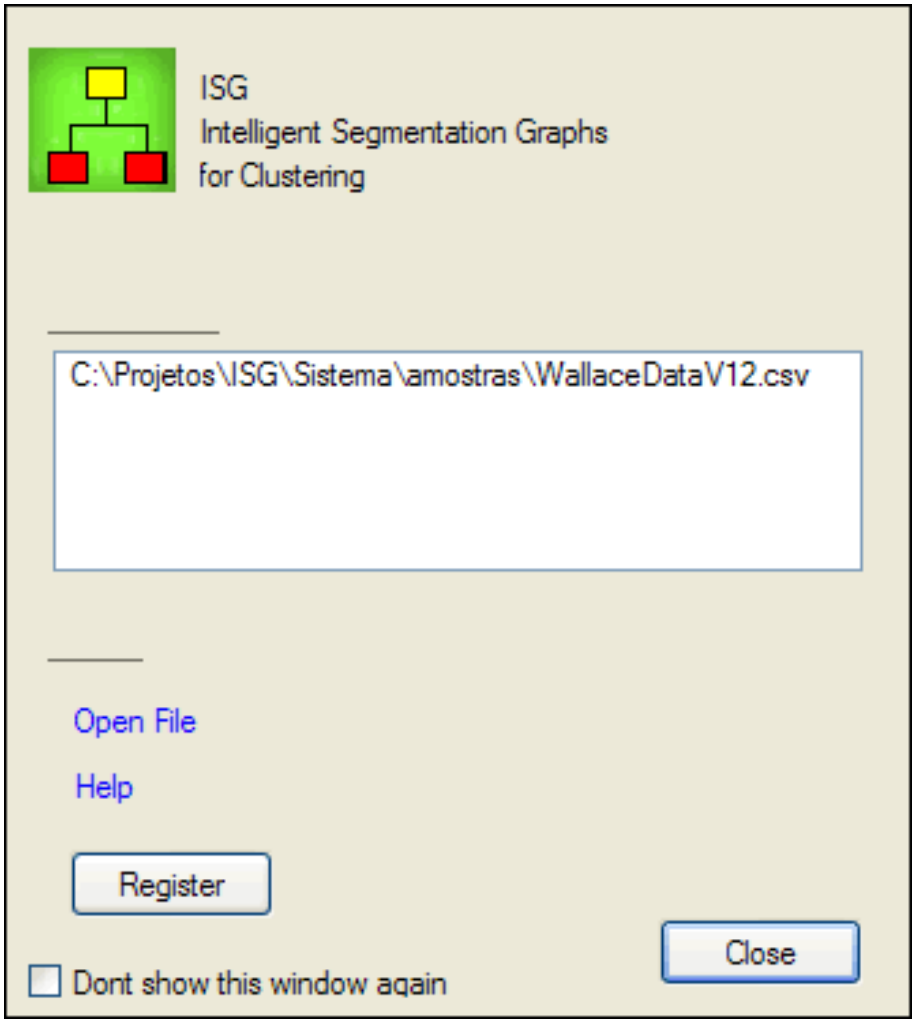


Figura 6.1: Tela Inicial.

6.2 Apresentando a Tela Principal

Na Figura 6.2 podemos visualizar a tela principal da ferramenta. Nesta tela o usuário pode observar a existência de uma janela chamada *Output* que é atualizada dinamicamente com informações do comportamento interno da ferramenta, tais

como: a descrição textual do modelo configurado, a sinalização da troca da medida de distância de Mahalanobis para Euclidiana quando Mahalanobis não é permitida, o passo atual da execução do algoritmo, o número de segmentos encontrados, etc.

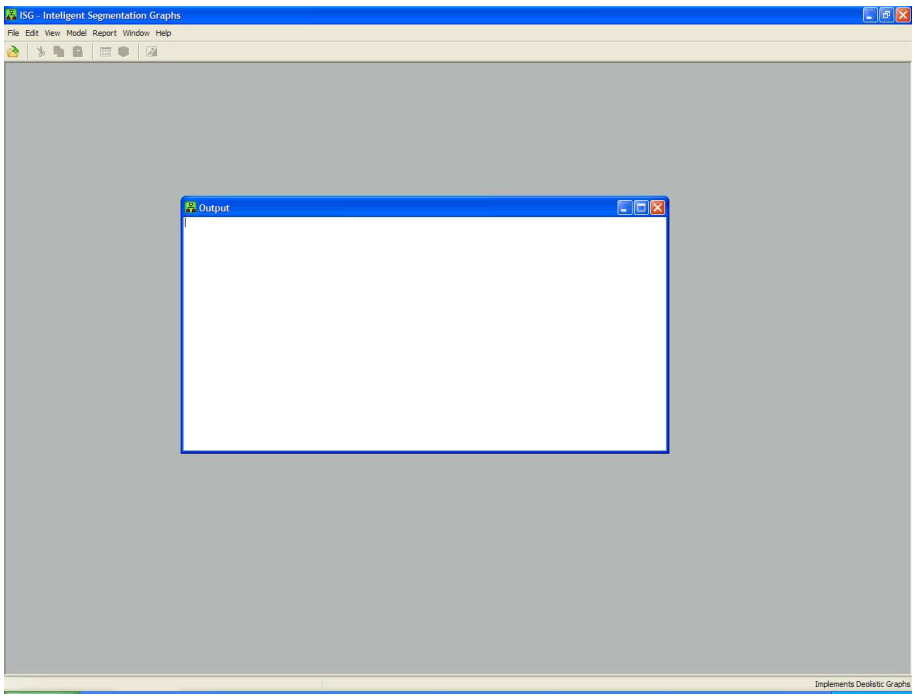


Figura 6.2: Tela Principal.

6.3 Apresentando as Opções de Menu

6.3.1 Menu *File* (Arquivo)

Este menu é utilizada para selecionar o arquivo de dados com as observações a serem segmentadas, acessar a configuração da impressora, imprimir e sair da ferramenta. A Figura 5.3 apresenta esta opção de menu.

São as seguintes as opções oferecidas pelo menu *File*:

- *Open* → *CSV File* - abre um arquivo de dados em formato CSV.
- *Recent Files* - apresenta a lista dos últimos 4 arquivos de dados utilizados pelo usuário. Estes arquivos estão disponíveis para serem abertos.
- *Close File* - fecha o arquivo de dados aberto, limpando todo os dados em *cache*.

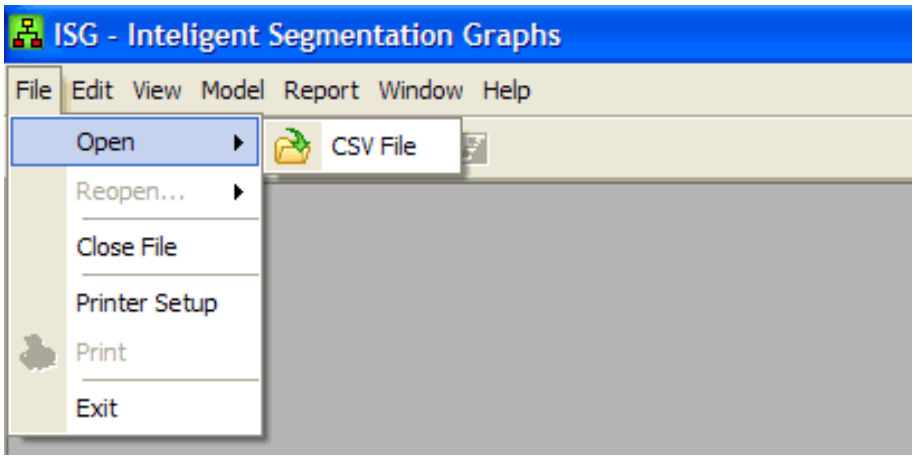


Figura 6.3: Menu File.

- *Printer Setup* - utilizado para configurar parâmetros da impressão, tais como: impressora, tipo de papel, cores, etc...
- *Print* - utilizado para imprimir as saídas produzidas pela execução do algoritimo Deolístico.

6.3.2 Menu *Edit* (Editar)

Este menu apresenta as operações padrões do Windows (PREPPERNAU, 2007) de interação com a área de transferência, sendo estas: recortar, copiar e colar, além de oferecer a opção de limpar a saída padrão da ferramenta, ou seja, a janela *Output*. A Figura 5.4 exibe o menu *Edit*.

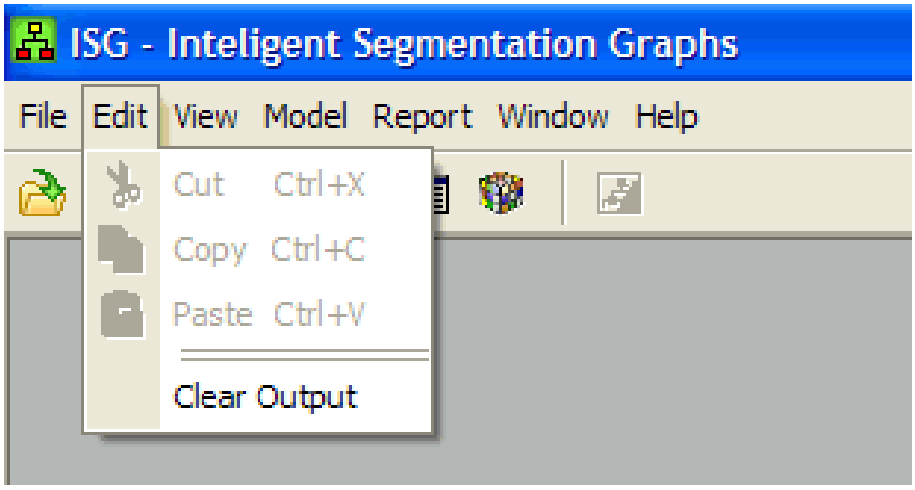


Figura 6.4: Menu Edit.

São as seguintes as opções oferecidas pelo menu *Edit*:

- *Cut* - operação padrão do Windows de recortar.
- *Copy* - operação padrão do Windows de copiar.
- *Past* - operação padrão do Windows de colar.
- *Clear Output* - utilizado para limpar a tela saída padrão, janela *Output* .

6.3.3 Menu *View* (Visualizar)

Este menu é utilizado para acessar as informações dos arquivos de dados de entrada do algoritmo e o *log* com informações do passo a passo da execução do algoritmo para aferição de seus resultados. Visualize este menu na Figura 5.5.

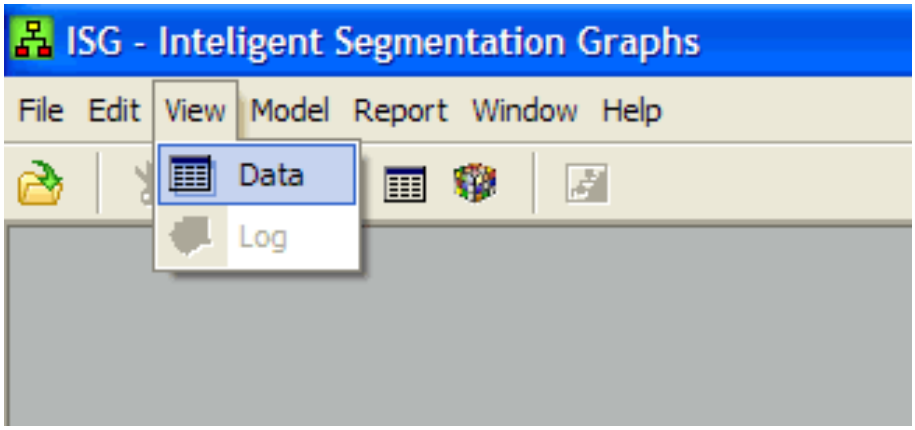


Figura 6.5: Menu View.

São as seguintes as opções oferecidas pelo menu *View*:

- *Data* - utilizado para apresentar a tela com as informações do arquivo de entrada que se encontra aberto.
- *Log* - esta opção apresenta uma tela com o *log* da execução do algoritmo.

6.3.4 Menu *Model* (Modelo)

Neste menu são feitas as principais operações da ferramenta. Dado que o usuário selecionou um arquivo de entrada, ele deve configurar a execução do algoritmo. A estrutura escolhida para armazenar os parâmetros do algoritmo, foi denominada nesta ferramenta de Modelo. Nele o usuário informa quais são as variáveis de entrada, seus

tipos, a medida de distância a ser utilizada, o número de segmentos desejados, etc. Veja a imagem deste menu na Figura 5.6.

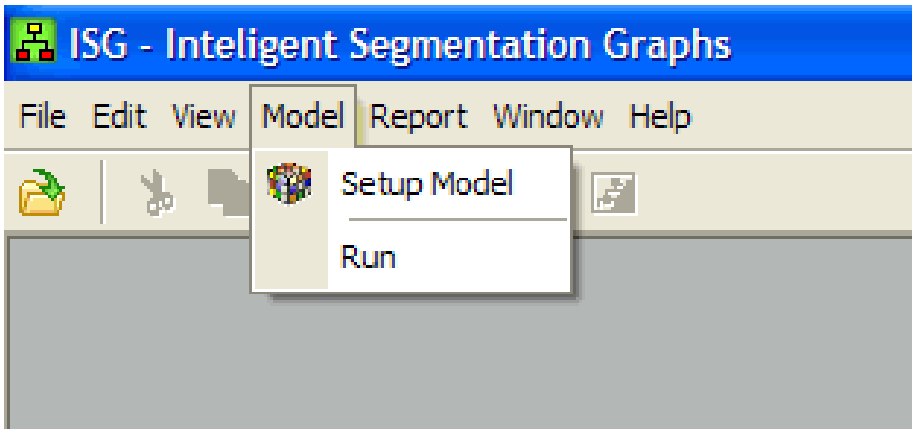


Figura 6.6: Menu Model.

São as seguintes as opções oferecidas pelo menu *Model*:

- *Setup* - utilizado para configurar o modelo. Ao selecionar este item de menu, a tela da Figura 5.7 será apresentada. Os números presentes em pequenos círculos laranjas são utilizados para apontar os elementos da tela que são explicados a seguir:

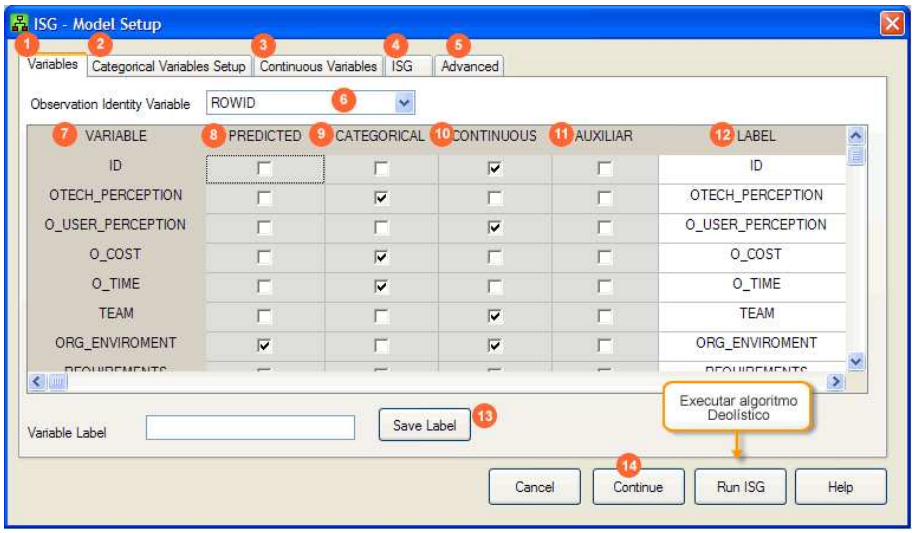


Figura 6.7: Configuração do Modelo.

(1) Pasta *Variables* - permite a configuração das variáveis de entrada do algoritmo.

(2) Pasta *Categorical Variables Setup* - seleciona a tela de configuração das variáveis categóricas. Nesta tela o usuário pode definir rótulos para os valores das variáveis categóricas. A Tabela 6.2 explica os elementos desta tela, que é apresentada na Figura 5.8.

Tabela 6.2: Tela de configuração dos rótulos para variáveis categóricas.

Item	**Descrição**
1	Escolhe a variável que terá rótulo definido para seus valores
2	Coluna que indica os valores da variável selecionada no item 1
3	Coluna que indica os rótulos da variável selecionada no item 1
4	Salva o valor do rótulo para o valor selecionado no *Grid* e digitado na caixa de texto *Value Label*

(3) Pasta *Continuos Variables* - seleciona a tela de configuração das variáveis contínuas. Esta opção esta desabilitada nesta versão. Por que o não oferecemos nenhuma parametrização para as variáveis contínuas, por exemplo, a normalização é feita através do método Z-score, e poderíamos ter outras opções.

(4) Pasta *ISG* - seleciona a tela de configuração dos parâmetros adicionais do algoritmo. Entre eles, o número de segmentos desejados, a medida de distância e a métrica utilizada para avaliar a dispersão das observações de um segmento. A Tabela 6.3 explica os elementos desta tela, que é apresentada na Figura 5.9.

Tabela 6.3: Configurando o algoritmo.

Item	**Descrição**
1	Informa o número máximo de segmentos desejado pelo usuário
2	O método informa qual a função de ganho na redução da diversidade que será utilizado: *Center Distance Measure* - Distância das observações ao centro do grupo e *Variance* - Variância da distância das observações do grupo entre si
3	Indica qual a medida de distância a ser utilizada. A mais conhecida é a Euclidiana, sendo que a de Mahalanobis lida melhor com variáveis correlacionadas (HAIR et al., 2005)

(5) Pasta *Advanced* - Configurações avançadas do algoritmo. Nesta tela o usuário informa o tamanho da amostra aleatória a ser utilizada pelo algoritmo caso o arquivo de entrada seja grande. Isso tem por objetivo per-

mitir a redução do esforço computacional e conseqüentemente de tempo de processamento. A Tabela 6.4 explica os elementos desta tela, que pode ser visualizada na Figura 5.10.

Tabela 6.4: Configurando parâmetros avançados.

Item	Descrição
1	Para um menor esforço do algoritmo podemos trabalhar com uma amostra do conjunto de observações quando calculamos a divisão de um segmento. Através desta caixa de texto o usuário informa o tamanho desta amostra.
2	As variáveis auxiliares são utilizadas para analisar o comportamento dos segmentos, mas pode ser que o usuário também queira ver as estatísticas das variáveis utilizadas para segmentação. Ao desmarcar esta opção, o usuário terá na análise de comportamento as estatísticas para todas as variáveis.

(6) Caixa *Observation Variable Identity* - este item serve para informar qual das variáveis do arquivo de entrada é identifica unicamente cada observação

(7) Coluna *Variable* - coluna que indica o nome das variáveis do arquivo de entrada.

(8) Coluna *Predicted* - coluna que indica quais são as variáveis preditoras, ou sejas, aquelas que serão usadas pelo algoritmo para encontrar os segmentos.

(9) Coluna *Categorical* - coluna que indica que a variável possui valores categóricos.

(10) Coluna *Continuos* - Coluna que indica que a variável possui valores contínuos.

(11) Coluna *Auxiliar* - coluna que indica quais as variáveis são consideradas auxiliares, ou seja, aquelas que serão utilizadas para derivar o comportamento dos segmentos encontrados. Esta variáveis não são utilizadas pelo algoritmo diretamente, são utilizadas pela ferramenta na produção das estatísticas de cada segmento.

(12) Coluna *Label* - coluna que informa o rótulo que será utilizado na apresen-

tação dos resultados do algoritmo quando a variável conectada ao rótulo for utilizada.

(13) Botão *Save Label* - Este botão é utilizado para gravar um novo rótulo para a variável selecionada no *grid*.

(14) *Continue* - este botão grava o modelo configurado e volta para tela principal do sistema.

(15) Botão *Run ISG* - esta opção de menu executa o algoritmo para o modelo que estiver configurado. Se o modelo ainda não estiver sido corretamente configurado, esta opção estará desabilitada.

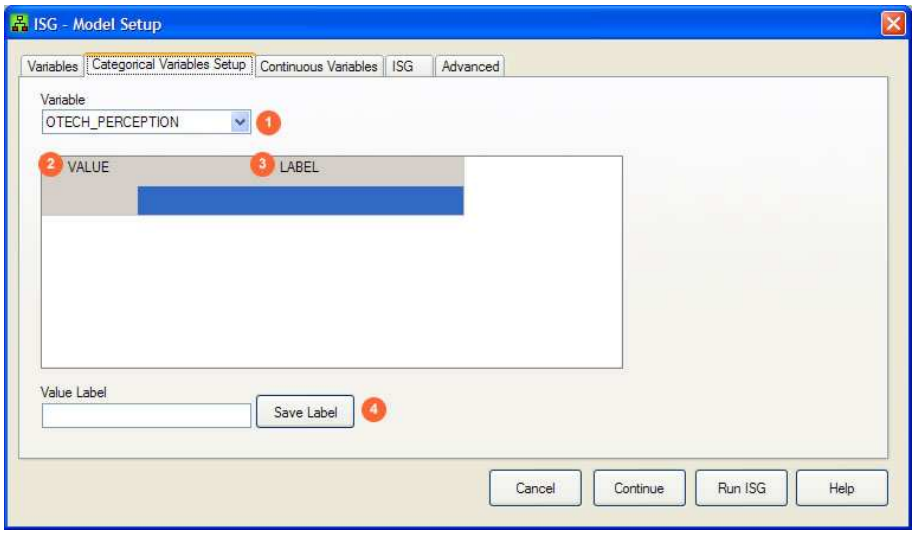


Figura 6.8: Configuração das Variáveis Categóricas.

6.3.5 Menu *Report* (Relatórios)

No menu *Report* o usuário visualiza as saídas da execução do algoritmo Deolístico, ou seja, a árvore gerada, as regras produzidas e as estatísticas das variáveis utilizadas para analisar o comportamento dos segmentos encontrados. As opções do menu *Report* são apresentada na Figura 5.10.

As seguintes as opções são oferecidas no menu *Report*:

- *Segmentation Tree* - Esta opção de menu apresenta a árvore gerada pela execução do algoritmo Deolísto pela ferramenta ISG. Veja o exemplo na Figura 5.11 e a explicação para os detalhes de cada nó da árvore na Figura 5.12.

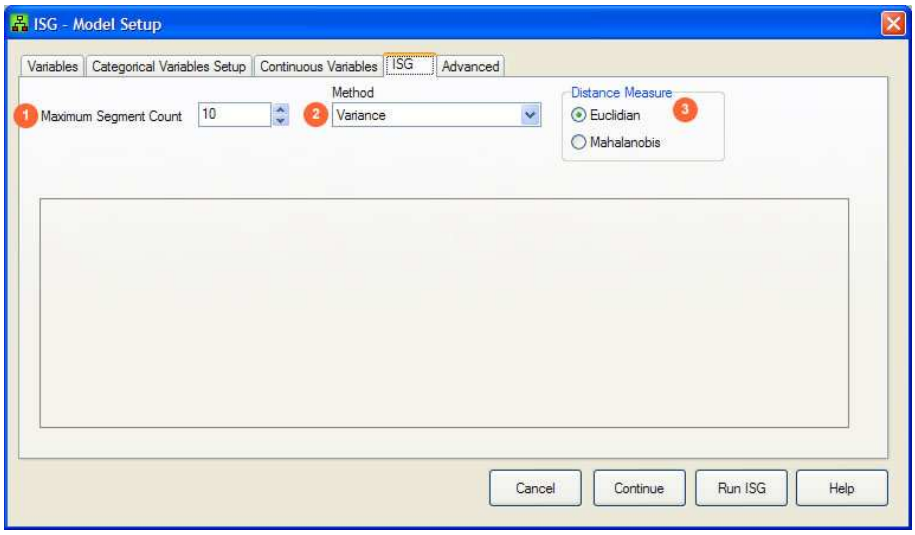


Figura 6.9: Configuração do ISG.

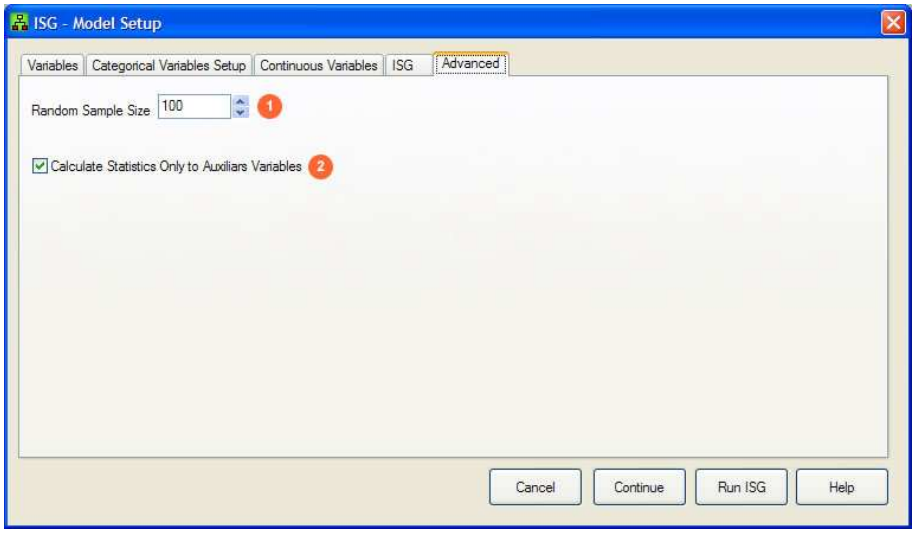


Figura 6.10: Configurações Avançadas.

- *Segmentation Rules* - As regras de segmentação encontradas pelo algoritmo Deolístico são acessadas por este item de menu. Veja a imagem da Figura 5.13 para um exemplo das regras produzidas..

- *Factual Rules* - A análise mais detalhada de cada segmento permite que o algoritmo refine as regras de segmentação, expondo com mais precisão as diferenças entre os diversos segmentos encontrados. Isto é possível substituindo as inequações e igualdades por intervalos de valores determinados pelo conteúdo das respectivas variáveis nos segmentos em questão. Com esta opção de menu

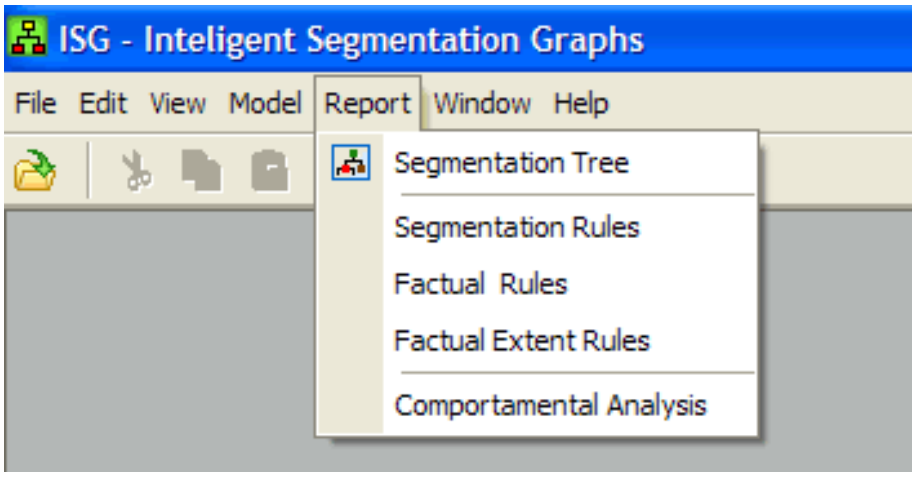


Figura 6.11: Menu Report.

o usuário poderá acessar esta opção. Veja a imagem da Figura 5.14 para um exemplo.

- *Factual Extent Rules* - Além das regras factuais, que procuram substituir as inequações e igualdades por intervalos de valores, determinados pelo conteúdo das respectivas variáveis nos segmentos em questão, as regras factuais extendidas aplicam esta mesma operação para as variáveis envolvidas no processo de segmentação que não estão presentes na regra de segmentação gerada. Veja a imagem da Figura 5.15 para um exemplo.

- *Comportamental Analysis* - Para que possamos extrair o comportamento dos segmentos encontrados, este comportamento pode ser derivado através das estatísticas apresentadas para as variáveis auxiliares escolhidas na configuração do modelo. Veja a imagem da Figura 5.16 para um exemplo.

6.3.6 Menu *Window* (Janelas)

Este menu permite manipular, organizar as janelas em aberto pela ferramenta atualmente. A Figura 5.17 apresenta as opções do Menu.

- *Close* - Fecha a janela atualmente ativa

- *Close All* - Fecha todas as janelas abertas.

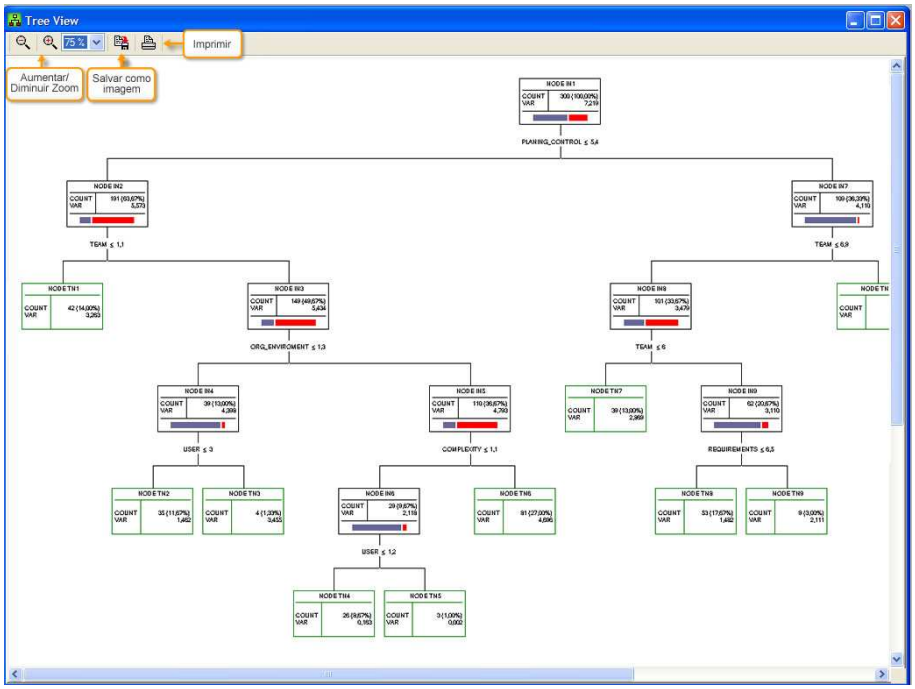


Figura 6.12: Árvore de Segmentação.

- *Tile Horizontally* - Organiza as janelas lado a lado horizontalmente
- *Tile Vertically* - Organiza as janelas lado a lado verticalmente
- *Minimize All* - Minimiza todas as janelas
- *Cascade* - Organiza as janelas em cascata
- *Arrange* - Organiza as janelas
- *Output* - Ativa a janela Output
- *Segment Tree* - Caso a janela da árvore de segmentação esteja aberta, ela aparece com uma entrada deste menu e pode ser ativada por aqui, este comportamento se repete para todas as janelas da ferramenta.

6.3.7 Menu *Help* (Ajuda)

A partir deste menu o usuário pode acessar a *home page* do produto, acessar o help da ferramenta e ver a tela com informações de contato e versão da ferramenta. Veja a Figura 5.18 com a imagem deste menu.

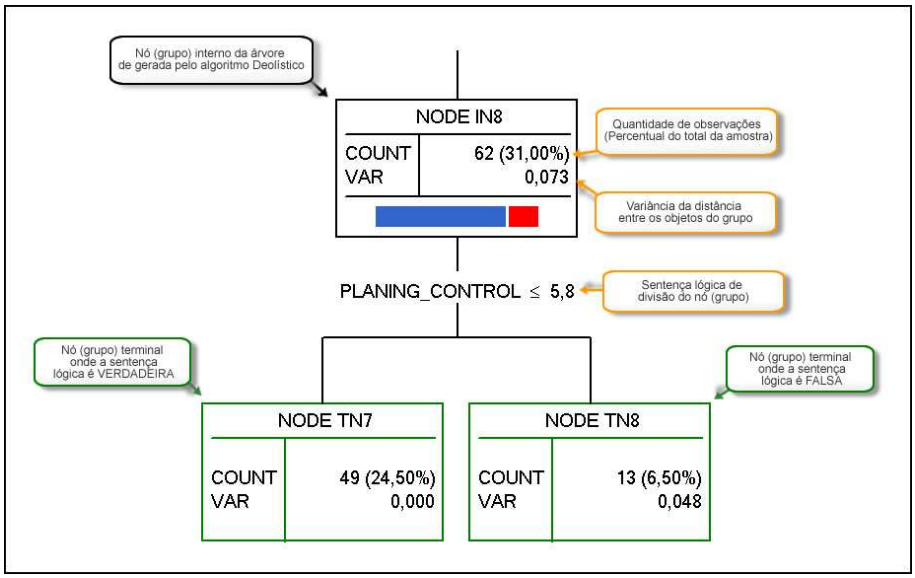
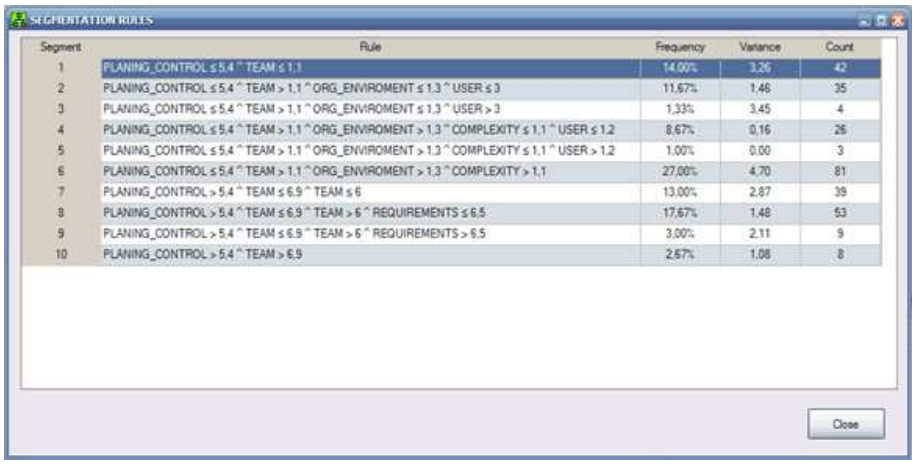


Figura 6.13: Detalhes de um nó da Árvore de Segmentação.



SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL ≤ 5,4 ^ TEAM ≤ 1,1	14,00%	3,26	42
2	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER ≤ 3	11,67%	1,46	35
3	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER > 3	1,33%	3,45	4
4	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER ≤ 1,2	8,67%	0,16	26
5	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER > 1,2	1,00%	0,00	3
6	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY > 1,1	27,00%	4,70	81
7	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM ≤ 6	13,00%	2,87	39
8	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM > 6 ^ REQUIREMENTS ≤ 6,5	17,67%	1,48	53
9	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM > 6 ^ REQUIREMENTS > 6,5	3,00%	2,11	9
10	PLANING_CONTROL > 5,4 ^ TEAM > 6,9	2,67%	1,08	8

Close

Figura 6.14: Tela Regras de Segmentação.

- *ISG Help* - acessa o *help* da ferramenta
- *ISG Home Page* - acessa o *site* da ferramenta na internet.
- *About* - apresenta informações de versão da ferramenta, contatos e copyright.

6.4 Apresentando a barra de ferramentas

A Figura 5.19 mostra uma opção mais rápida de acesso as funcionalidades da ferramenta.

FACTUAL SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL [1.2..2.3] ^ TEAM [1..1.1]	14,00%	3,26	42
2	PLANING_CONTROL [1,1..2] ^ TEAM [1.3..1.9] ^ ORG_ENVIROMENT [1..1.3] ^ USER [1.2..3]	11,67%	1,46	35
3	PLANING_CONTROL [1.1..2.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [1..1.3] ^ USER [6.7..7]	1,33%	3,45	4
4	PLANING_CONTROL [1.2..1.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [2..2.7] ^ COMPLEXITY [1.1..1.1] ^ USER [1..1.2]	8,67%	0,16	26
5	PLANING_CONTROL [1.1..5] ^ TEAM [1.3..6] ^ ORG_ENVIROMENT [2.3..4.5] ^ COMPLEXITY [1.1..1.1] ^ USER [1.5..6.8]	1,00%	0,00	3
6	PLANING_CONTROL [1..5.4] ^ TEAM [1.3..6.3] ^ ORG_ENVIROMENT [1.5..6.7] ^ COMPLEXITY [1.3..6.8]	27,00%	4,70	81
7	PLANING_CONTROL [5.7..6.8] ^ TEAM [4.7..6]	13,00%	2,87	39
8	PLANING_CONTROL [5.8..7] ^ TEAM [6.3..6.9] ^ REQUIREMENTS [1..6.5]	17,67%	1,48	53
9	PLANING_CONTROL [6.9..6.9] ^ TEAM [6.6..6.6] ^ REQUIREMENTS [6.6..6.6]	3,00%	2,11	9
10	PLANING_CONTROL [6.8..6.8] ^ TEAM [7..7]	2,67%	1,08	8

Figura 6.15: Tela Regras Factuais.

FACTUAL SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL [1.2..2.3] ^ TEAM [1..1.1]	14,00%	3,26	42
2	PLANING_CONTROL [1,1..2] ^ TEAM [1.3..1.9] ^ ORG_ENVIROMENT [1..1.3] ^ USER [1.2..3]	11,67%	1,46	35
3	PLANING_CONTROL [1.1..2.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [1..1.3] ^ USER [6.7..7]	1,33%	3,45	4
4	PLANING_CONTROL [1.2..1.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [2..2.7] ^ COMPLEXITY [1.1..1.1] ^ USER [1..1.2]	8,67%	0,16	26
5	PLANING_CONTROL [1.1..5] ^ TEAM [1.3..6] ^ ORG_ENVIROMENT [2.3..4.5] ^ COMPLEXITY [1.1..1.1] ^ USER [1.5..6.8]	1,00%	0,00	3
6	PLANING_CONTROL [1..5.4] ^ TEAM [1.3..6.3] ^ ORG_ENVIROMENT [1.5..6.7] ^ COMPLEXITY [1.3..6.8]	27,00%	4,70	81
7	PLANING_CONTROL [5.7..6.8] ^ TEAM [4.7..6]	13,00%	2,87	39
8	PLANING_CONTROL [5.8..7] ^ TEAM [6.3..6.9] ^ REQUIREMENTS [1..6.5]	17,67%	1,48	53
9	PLANING_CONTROL [6.9..6.9] ^ TEAM [6.6..6.6] ^ REQUIREMENTS [6.6..6.6]	3,00%	2,11	9
10	PLANING_CONTROL [6.8..6.8] ^ TEAM [7..7]	2,67%	1,08	8

Figura 6.16: Tela Regras Factuais Estendidas.

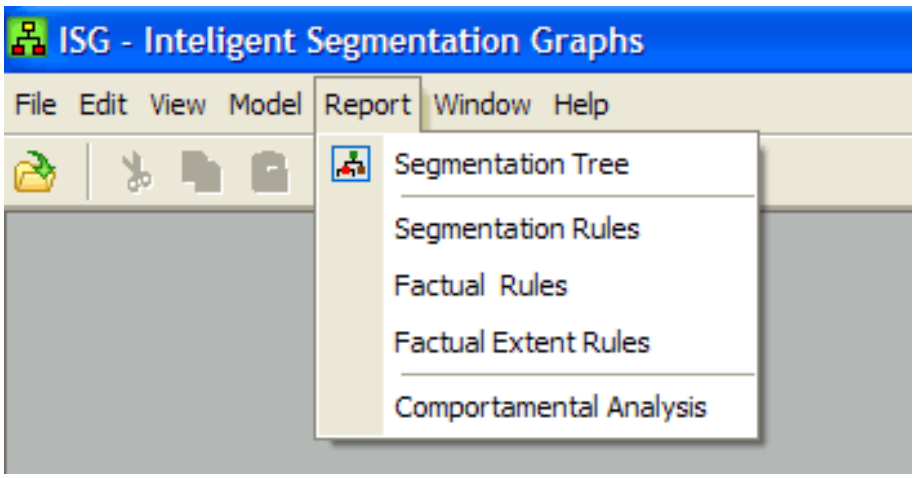


Figura 6.17: Tela Comportamental Analysis.

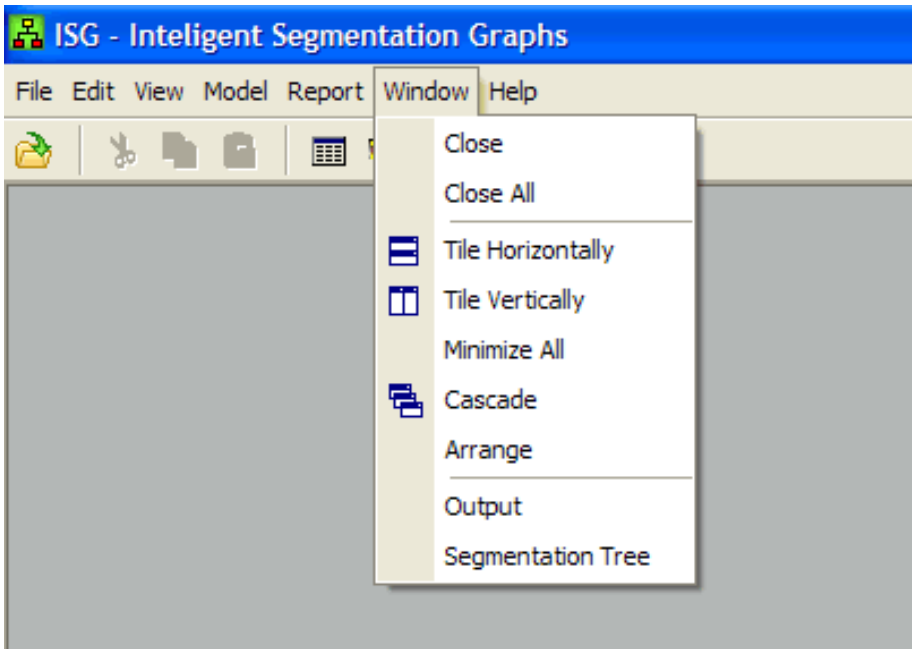


Figura 6.18: Menu Window.

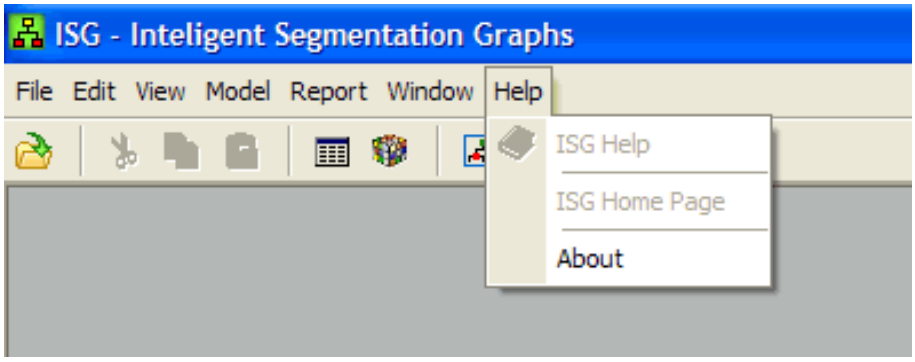


Figura 6.19: Menu Help.

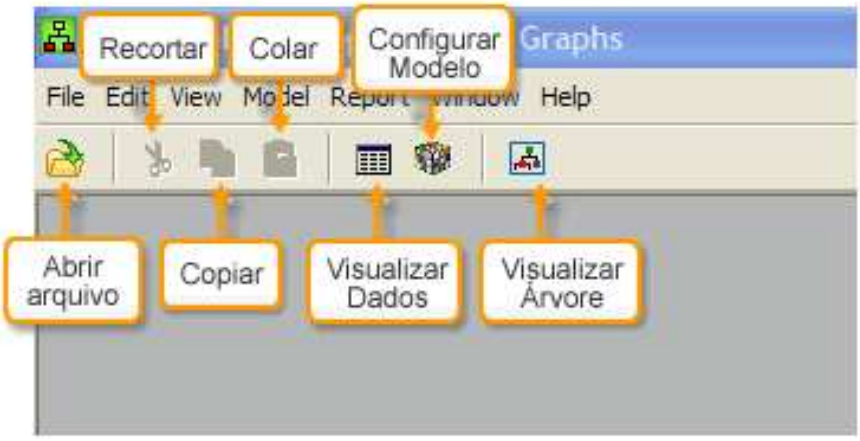


Figura 6.20: Barra de Ferramentas.

7 DISCUSSÃO E CONCLUSÕES

7.1 Perguntas e Respostas

No início deste trabalho nos comprometemos a apresentar um método, baseado em uma nova abordagem para análise de cluster, que permite que os gerentes de projeto de *software* identifiquem com maior facilidade os riscos a que seus projetos estão expostos. Abaixo apresentamos as respostas para algumas questões chaves sobre as implicações que os resultados providos pelo método traz tanto para o gerenciamento de projeto como para o desenvolvimento de estratégias de negócio.

Porque o questionário de Wallace *et al.* é utilizado como base para o método proposto nesta dissertação

O modelo desenvolvido por Wallace *et al.* (WALLACE; KEIL; RAI, 2004) possui vantagens importantes sobre as propostas anteriores de modelos de análise de risco que se tornaram populares tanto no mundo acadêmico quanto entre os desenvolvedores de software, especialmente os modelos desenvolvidos por Moynihan (MOYNIHAN, 1997), Karolak (KAROLAK, 1995) e Heemstra e Kusters (HEEMSTRA; KUSTERS, 1996):

- O modelo proposto por Wallace *et al.* foi construído sobre uma amostra de tamanho razoável (mais de 500 observações) de projetos recentes dos mais variados tipos, executados dentro e fora dos Estados Unidos, com o auxílio de análise fatorial, ou seja sobre um uma base sólida com a ajuda de um método estatístico largamente utilizado no estudo de casos do mundo real.

Veja (TABACHNICK; FIDELL, 2006) para uma introdução a análise fatorial;

- A segunda parte do questionário que faz parte do modelo permite que as condições de risco nos quais projetos são executados sejam facilmente conectados com indicadores de performance comumente utilizados no mundo real, tais como: tempo de projeto, custo, satisfação do usuário e atendimentos a requisitos previamente estabelecidos, entre outros.

  Essa conecção pode ser estabelecida através de uma variedade de métodos estatísticos e computacionais multivariados, tais como: árvores de classificação, redes neurais, regressão linear e análise de cluster. Veja (WITTEN; FRANK, 2005) e (TABACHNICK; FIDELL, 2006) para uma introdução a estes métodos. Tudo isso permite que o gerente de projetos tome ações corretivas antes que o pior aconteça.

- O modelo pode ser facilmente reaplicado a diferente situações, de acordo com necessidades especificas do desenvolvedor de software, permitindo que sejam feitas calibrações para diferentes realidades.

Como o método apresentado nesta dissertação se compara a outros métodos de gerenciamento de risco

O método proposto neste dissertação permite que ação dos fatores de risco sobre todos os indicadores de performance sejam considerados simultaneamente. Em adição o método conecta dimensões de risco a indicadores de performance utilizando regras que são fáceis de serem lidas e compreendidas. A conjunção dessas duas "features", que não está presente em nenhum outro método de análise de risco, permite que o gerente tenha uma visão mais holista de como risco de seus projetos afetam os indicadores de performance. Tudo isso propicia a tomada de ações corretivas mais eficazes e eficientes, antes que o pior aconteça.

Como as organizações e seus gerentes de projetos se beneficiam do método proposto neste trabalho?

Ao longo das últimas décadas a área TI tem experimentado avanços consideráveis na forma de uma grande variedade de novos conceitos, métodos e ferramentas, abrindo novas oportunidades para o desenvolvimento industrial, assim como da área de serviços. São jogos computacionais, sistemas embarcados para gerenciamento de negócios e entretenimento, sistemas inteligentes de segurança, telefonia móvel, televisão sobre demanda, educação a distância, etc.(HOSNI; KHALIL, 2004).

A despeito de todos os benefícios trazidos para os consumidores por estes novos produtos e serviços, para se manterem competitivos nos dias de hoje as organizações precisam inovar constantemente, criando produtos novos e mais sofisticados em um ritmo cada vez mais acelerado. Entretanto, a inovação freqüentemente exige a execução de projetos cada vez mais complexos, que requerem a coordenação dos esforços de equipes multidisciplinares, o estabelecimento de alianças estratégicas, a terceirização de atividades dos projetos, além da utilização de tecnologias desenvolvidas recentemente (ALENCAR et al., 2007).

Tudo isso impõe desafios enormes para os gerentes de projeto, mesmo sem levar em conta a necessidade de se produzirem resultados cada vez mais rápido. Quando a pressão fica muito grande, o senso comum costuma ser deixado de lado, e passos cruciais do desenvolvimento de projetos são ignorados. A qualidade dos produtos finais do projeto (*deliverables*) costuma ficar aquém do desejado e o re-trabalho que é necessário para consertar os erros tem um custo muitas vezes maior do que se as atividades de projetos tivessem sido executadas corretamente da primeira vez (CHARREL; GALARRETA, 2004).

Ao criar uma base de conhecimento a partir da qual é possível gerar regras que relacionam dimensões de risco com dimensões de performance de projetos, o método apresentado neste trabalho fornece aos gerentes de projeto o ferramental necessário para vislumbrar as conseqüências de suas decisões ao longo do tempo, favorecendo, dessa forma, que os projetos sejam entregues dentro do prazo, de acordo com o orçamento e em sintonia com os requisitos que ele se propuseram a cumprir.

O método proposto neste trabalho pode ser utilizado como um ferramenta

de negociação efetiva?

Existe um principio bem estabelecido no mundo dos negócios que diz que os clientes estão sempre buscando obter mais por menos. Entretanto, como nos lembra sabiamente a Senadora Norte Americana Barbara Mikulski (MIKULSKI; OATES, 1999):

> "Freqüentemente você não pode ter mais por menos, você pode ter somente aquilo pelo qual pagou".

A aparente contradição entre estas duas formas de pensar revela a batalha que muitas vezes é travada entre gerentes de projetos, empresas prestadoras de serviços de TI e as organizações clientes destas empresas. Embora os clientes possuam o desejo natural de pagar menos pelo *software* que eles precisam, os gerentes de projetos só conseguirão entregar o que prometeram se um orçamento justo for estabelecido entre as partes.

Dado que as regras geradas pelo algoritmo Deolístico são conjunções de predicados que são fáceis de serem lidas e compreendidas, e podem ser facilmente apresentadas a pessoas sem conhecimento técnico, estas regras podem ser utilizadas como uma ferramenta de conscientização da gerencia senior e dos clientes no que diz respeito a aspectos críticos dos projetos. Especialmente aqueles que requerem investimentos extras para se assegurar que o projeto vai entregar aquilo que foi combinado, em sintonia com os requisitos de performance de projeto que foram previamente estabelecido.

Como o método proposto neste trabalho favorece o ganho de vantagem competitiva nos negócios?

De acordo com Michael Porter existem apenas duas formas de se obter vantagem competitiva sobre os concorrentes em um mercado aberto: redução de custo e diferenciação do produto (PORTER, 2004). Com o advento do *e-commerce* e *e-business* e da globalização da economia mundial, um número cada vez maior de organizações tem voltado a sua atenção para o desenvolvimento de *software* como

uma forma de reduzir custos e prover uma grande quantidade de produtos e serviços inovadores que capturem a imaginação dos consumidores.

No entanto, ao favorecer a entrega de projetos de *software* de acordo com uma variedade de indicadores de performance (incluindo tempo de projeto, custo, aderência aos requisitos, aderência as expectativas dos usuários, facilidade de uso dos produtos finais do projeto, etc.), em muitas circunstâncias, o método proposto neste trabalho ajuda na obtenção de vantagens competitivas sobre os concorrentes, em particularmente quando a TI é peça central das estratégias de negócio.

Como o método proposto neste artigo favorece o gerenciamento de projetos em geral?

Apesar de todo o treinamento e experiência que um gerente de projeto possa ter acumulado ao longo da sua vida profissional e de todo o esforço que possa empregar no seu trabalho, projetos são atividades repletas de incertezas, tornado a atividade de planejamento um desafio de grande complexidade. Não somente os projetos dizem respeito a eventos e atividades que nunca foram realizadas antes, mas os eventos e atividades que fazem parte de um projeto jamais ocorrerão ou serão executadas da mesma forma. Portanto, dificilmente um projeto será executado da maneira que ele foi planejado, requerendo ajustes constantes durante a sua execução. Quanto mais complexo e maior for o projeto, mais ajustes ele deverá requerer (GREGG, 2006).

Ao longo do tempo, muitos conceitos e ferramentas foram desenvolvidas para ajudar os gerentes de projeto na condução dos projetos pelos quais são responsáveis (TURNER, 2008). No entanto, muito pouco tem sido feito para relacionar dimensões de risco com indicadores de performance de projeto na forma de regras de negócio que sejam fáceis de serem lidas e compreendidas e que podem mudar o destino de um projeto ao motivar a tomada de ações corretivas, no momento em que elas são mais necessárias. Este trabalho é uma passo adiante no esforço de assegurar que esta lacuna seja preenchida corretamente.

Podem os resultados apresentados neste trabalho ser obtido com o su-

porte de qualquer algoritmo de clusterização?

Não, de forma alguma. Os algoritmos de clusterização foram e continuam sendo concebidos para revelar segmentos dissimilares de objetos tão similares quanto possível. Até o momento, prover regras que sejam capazes de distinguir objetos em um segmento dos objetos nos outros segmento tem sido considerado um problema secundário na análise de cluster e não são apropriadamente abordados pelos outros algoritmos de clusterização. Veja (ROMESBURG, 2004; EVERITT; LANDAU; LEESE, 2001) para uma lista abrangente dos algoritmos de clusterização que estão a nossa disposição.

Quais são as principais limitações do método apresentado nesta dissertação?

Em linhas gerais o método proposto nesta dissertação sofre das seguintes limitações:

- As variáveis não devem conter *missing values*;
- Seguindo a orientação de Hair *et al.* (HAIR et al., 2005) o coeficiente mínimo entre o número de observações e o número de variáveis em um estudo de análise de cluster deve ser maior ou igual que 20. Veja também Hogarty *et al.* (HOGARTY et al., 2005) para uma discussão bastante ampla sobre o tamanho da mostra mínima para análise de cluster[1];
- O método não lida diretamente com variáveis ordinais e intervalares. Elas necessariamente tem que ser classificadas em variáveis nominais e de razão.

7.2 Conclusão

Este trabalho apresenta uma nova abordagem para análise de cluster ao mesmo tempo em que demonstra como ela pode ser utilizada para relacionar fatores de

[1] Ao consultar tanto Hair et al. quanto Hogarty et al. tenha em mente que análise de cluster também é conhecida como *Q factor analysis*

riscos em projetos de *software* com indicadores de performance de projetos. Esta relação é feita através de um conjunto de regras baseadas em sentenças lógicas que são fáceis de serem lidas, compreendidas e divulgada entre as partes interessadas.

O método proposto requer a construção de uma base de conhecimento com informações de projetos de *software*. Esta base pode ser facilmente desenvolvida com o auxílio de um questionário baseada na escala de *Likert*, utilizando o conhecimento dos gerentes e dos clientes dos projetos desenvolvidos por uma organização.

O método favorece a tomada de ações corretivas pelos gerentes de projeto, quando isso se faz necessário, e a entrega de projetos de acordo como um conjunto pré-estabelecido de indicadores de performance. Em adição, o método ajuda as organizações a obterem vantagens competitivas sobre os concorrentes.

7.3 Trabalhos Futuros

Como sugestões para aqueles que desejarem seguirem os passos delineados nesta dissertação apresentamos as seguintes sugestões:

- *Inclusão de variáveis substitutas* - para superar as limitações impostas pela falta de um ou mais valores em uma determinada variável alguns métodos de mineração usam o conceito de variável substituta, isto é eles procuram por uma variáveis que melhor represente o valor que está faltando dando prosseguimento a análise;

- *Estudo sobre tamanho mínimo da amostra em relação ao número de variáveis* - Um estudo poderia ser empreendido para se determinar com maior precisão o número mínimo de observações em relação ao número de variáveis que seriam necessárias para se levar adiante um processo de clusterização. Neste caso é importante alertar que trata-se de um problema em aberto em análise de cluster (SAUSEN; TOMCZAK; HERRMANN, 2005).

- *Comparação com outros métodos de análise multivariada* - Nesta dissertação provemos uma comparação do método aqui proposto com outros métodos de

análise de risco. Seria interessante comparar o método proposto com outros métodos de análise mumtivariada.

REFERÊNCIAS

ALENCAR, A. J.; RODRIGUES, G. G.; SCHMITZ, E. A.; FERREIRA, A. L. Combining Decorated Classification Trees with RCPS Stochastic Models to Gain New Valuable Insights into Software Project Management. In: INTERNATIONAL CONFERENCE ON SOFTWARE ENGINEERING & KNOWLEDGE ENGINEERING (SEKE 2007), 19., 2007, Boston, MA, USA. **Proceedings...** Knowledge Systems Institute Graduate School, 2007. p.151–155.

ALENCAR, A. J. S. M. de; CRUZ, L. T.; SCHMITZ, E. A.; FERREIRA, A. L. Using a Novel Approach to Cluster Analysis to Gain New Valuable Insights into Software-Project Risk Management. In: IEEE/IFIP INTERNATIONAL WORKSHOP ON BUSINESS-DRIVEN IT MANAGEMENT (BDIM), 2008, Salvador, BH. **Anais...** IEEE Press, 2008. p.40–48.

BACCARINI, D.; SALM, G.; LOVE, P. E. Management of Risks in Information Technology Projects. **Industrial Management & Data Systems**, Bingley, United Kingdom, v.104, n.4, p.286–295, 2004.

BARNES, S. **E-Commerce and V-Business**. 2nd.ed. Oxford, United Kingdom: Butterworth-Heinemann, 2007.

BEER, S. IT project failures up sharply according to US report. **ITWire**, Australia, November 26th 2004.

BERSON, A.; SMITH, S.; THEARLING, K. **Building data mining applications from CRM**. Kindle Edition.ed. Columbus, OH, USA: McGraw-Hill, 2002.

BHATT, G. D.; GROVER, V. Types of Information Technology Capabilities and Their Role in Competitive Advantage: an empirical study. **J. Manage. Inf. Syst.**, Armonk, NY, USA, v.22, n.2, p.253–277, 2005.

BONDY, J. A.; MURTY, U. S. R. **Graph Theory**. New York, NY, USA: Springer, 2008. (Graduate Texts in Mathematics).

BRACE, I. **Questionnaire Design**: how to plan, structure and write survey material for effective market research. 2nd.ed. London, UK: Kogan Page, 2008. (Market Research in Practice).

CANTU, M. **Mastering Delphi 7**. Berkeley, CA, USA: Sybex, 2003.

CHARREL, P.-J.; GALARRETA, D. **Project Management and Risk Management in Complex Projects**: studies in organizational semiotics. New York, NY, USA: Springer, 2004. (1th edition).

DANIELS, C. B.; LAMARS-II, W. J. Complexity as a Cause of Failure in Information Technology Project Management. In: IEEE INTERNATIONAL CONFERENCE ON SYSTEM OF SYSTEMS ENGINEERING, 2007, San Antonio, Texas, USA. **Anais...** IEEE, 2007. p.1–7.

DAVIS, J. C. **The Human Story** : our history, from the stone age to today. New York, NY, USA: HarperCollins, 2004.

DEER, P. J.; EKLUND, P. W.; NORMAN, B. D. A Mahalanobis Distance Fuzzy Classifier. In: AUSTRALIAN NEW ZEALAND CONFERENCE ON INTELLIGENT INFORMATION SYSTEMS, 1996, Adelaide, SA, Australia. **Proceedings...** IEEE Press, 1996. p.220–223.

DEO, N. **Graph Theory with Applications to Enginering and Computer Science**. New Delhi, India: Prentice-Hall of India, 2004.

DEY, P. K.; KINCH, J.; OGUNLANA, S. O. Managing risk in software development projects: a case study. **Industrial Management & Data Systems**, Howard

House,Wagon Lane,Bingley BD16 1WA,United Kingdom, v.107, n.2, p.284–303, 2007.

EVERITT, B. S.; LANDAU, S.; LEESE, M. **Cluster Analysis**. 4th.ed. London, UK: Arnold, 2001.

GLASS, R. L. The Standish report: does it really describe a software crisis? **Commun. ACM**, New York, NY, USA, v.49, n.8, p.15–16, 2006.

GLENDON, A. I.; CLARK, S. G.; MCKENNA, E. **Humman safety and risk management**. 2nd.ed. 6000 Broken Sound Parkway, NW: CRC Press, 2006. 59-62p.

GREGG, W. Is It Time for a Consultant. **Multichannel Merchant**, [S.l.], May 31, 2006.

GROSS, J. L.; YELLEN, J. **Handbook of Graph Theory**. Boca Raton, FL, USA: CRC, 2003. (Discrete Mathematics and Its Applications).

HAIR, J.; ANDERSON, R.; TATHAM, R.; BLACK, W. **Multivariate Data Analysis**. 6th.ed. Upper Saddle River, New Jersey 07458, USA: Prentice Hall, 2005.

HAN, J.; KAMBER, M. **Data Mining**: concepts and techniques. San Fransisco, CA, USA: Morgan Kaufmann, 2001. (The Morgan Kaufmann Series in Data Management Systems).

HEEMSTRA, F. J.; KUSTERS, R. J. Dealing with risk : a practical approach. **Journal of Information Technology**, Oxford, UK, v.11, n.4, p.333–346, 1996.

HOGARTY, K. Y.; HINES, C. V.; KROMREY, J. D.; FERRON, J. M.; MUMFORD, K. R. The Quality of Factor Solutions in Exploratory Factor Analysis: the influence of sample size. **Educational and Psychological Measurement**, New York, NY, USA, v.65, n.2, p.202–226, 2005.

HOSNI, Y. A.; KHALIL, T. **Management of Technology - Internet Economy**: opportunities and challenges for developed and developing regions of the world. Amsterdam, Netherlands: Elsevier Scienc, 2004.

KAROLAK, D. W. **Software Engineering Risk Management**. San Francisco, CA: John Wiley, 1995.

KAUFMAN, L.; ROUSSEEUW, P. J. **Finding Groups in Data**: an introduction to cluster analysis. Malden, MA, USA: Wiley-Interscience, 1990.

KEILL, M.; LI1, L.; MATHIASSEN1, L.; ZHENG1, G. The Influence of Checklists and Roles on Software Practitioner Risk Perception and Decision-Making. In: HAWAII INTERNATIONAL CONFERENCE ON SYSTEM SCIENCES, 39., 2006, Kauai, HI, USA. **Proceedings...** IEEE Computer Society, 2006. v.09, p.229–239.

KENDRICK, T. **Identifying and Managing Project Risk**: essential tools for failure proofiling your project. 2nd.ed. New York, New York: AMACOM, 2009. 1-2p.

LAAN, M. J. van der; POLLARD, K. S. A new algorithm for hybrid hierarchical clustering with visualization and the bootstrap. **Jornal of Statistical Planning and Inference**, Amsterdam, Netherlands, v.117, p.275–303, 2003.

LARSEN, R. J.; MARX, M. L. **An Introduction to Mathematical Statistics and Its Applications**. Upper Saddle River , New Jersey , USA: Prentice Hall, 1981. 785 pages.

LINNAEUS, C. **Genera plantarum eorumque characteres naturales secundum numerum, figuram, situm & proportionem omnium fructificationis partium**. Leiden, The Netherlands: [s.n.], 1737. In Latin.

MARCHI, S. Extrasolar planet taxonomy: a new statistical approach. **The Astrophysical Journal**, Padova, Italy, v.666, n.1, p.475–48, May 2007.

MASTICOLA, S. P. A Simple Estimate of the Cost of Software Project Failures and the Breakeven Effectiveness of Project Risk Management. In: FIRST INTERNA-

TIONAL WORKSHOP ON THE ECONOMICS OF SOFTWARE AND COMPUTATION, 2007, Minneapolis, MN, USA. **Proceedings...** IEEE Computer Society, 2007. p.6–9.

MIKULSKI, B.; OATES, M. **Capitol Venture**. New York , USA: Signet Books, 1999.

MILLIGAN, G. W.; COOPER, M. C. An examination of procedures for determining the number of clusters in a data set. **Psychometrika**, New York, NY, USA, v.50, n.2, p.159–179, June 1985.

MOJENA, R. Hierarchical grouping methods and stopping rules: an evaluation. **The Computer Journal**, Oxford, UK, v.20, n.4, p.359–363, 1977.

MOLØKKEN, K.; JØRGENSEN, M. A review of software surveys on software effort estimation. In: INTERNATIONAL SYMPOSIUM ON EMPIRICAL SOFTWARE ENGINEERING (ISESE), 2003, Roman Castles (Rome), Italy. **Proceedings...** IEEE Press, 2003. p.223–230.

MOYNIHAN, T. Riskman : a prototype tool for risk analysis for software. In: THE 3rd INTERNATIONAL CONFERENCE ON COMPUTER AIDED SOFTWARE ENGINEERING, 1989, London, England. **Anais...** ACM Press, 1989.

MOYNIHAN, T. How experienced project managers access risk. **IEEE Software**, Washington, USA, v.14, n.3, p.35–41, 1997.

MYERS, J. H. **Segmentation & Positioning for Strategic Marketing Decisions**. Florence, KY, USA: South-Western Educational Publisher, 1996.

PMBOK, G. **Project Management Body of Knowledge**. Newtown Square, Pennsylvania USA: Project Management Institute, 2000.

PORTER, M. E. **Competitive Strategy**. New York, NY, USA: Free Press, 2004.

PREPPERNAU, J. **Microsoft Windows Vista Step by Step**. Seattle, WA, USA: Microsoft Press, 2007.

ROBERT, M. **The Power of Strategic Thinking**. New Yourk, NY: MCGraw-Hill, 2000.

ROKACH, L.; MAIMON, O. Top-down induction of decision trees classifiers - a survey. **IEEE Transactions on Systems, Man and Cybernetics**, New York, NY, USA, v.35, n.4, p.476–487, 2005. Part C: Applications and Reviews.

ROMESBURG, C. **Cluster Analysis for Researchers**. NORTH CAROLINA, USA: Lulu.com, 2004.

SAUSEN, K.; TOMCZAK, T.; HERRMANN, A. Development of a taxonomy of strategic market segmentation: a framework for bridging the implementation gap between normative segmentation and business practice. **Journal of Strategic Marketing**, New York, NY, USA, n.13, p.151–178, 2005.

SCHAFFER, C. M.; GREEN, P. E. An Empirical Comparison of Variable Standardization Methods in Cluster Analysis. **Multivariate Behavioral Research**, New York, NY, USA, v.31, n.2, p.149–167, 1996.

SCHMIDT, R.; LYYTINEN, K.; KEIL, M.; CULE, P. Identifying software project risks: an international Delphi study. **Journal of Management Information Systems**, Boston, USA, v.17, n.4, p.5–36, Spring 2001.

SCHMITZ, E. A.; ALENCAR, A. J. **Análise de Risco em Gerência de Projetos**. 1^a.ed. Tijuca, Rio de Janeiro - Brasil: BRASPORT, 2007.

SCHMITZ, E. A.; ALENCAR, A. J.; VILAR, C. B. **Modelos Qualitativos de Análise de Risco**. 1^a.ed. Tijuca, Rio de Janeiro - Brasil: BRASPORT, 2007.

SLOVIC, P. **The Perception of Risk**. London, UK: Earthscan Publications, 2000.

STEVENS, S. S. On the theory of scales of measurement. **American Association for the Advancement of Science**, Washington, DC, USA, v.103, n.2684, p.677–680, 1946.

TABACHNICK, B. G.; FIDELL, L. S. **Using Multivariate Statistics**. 5th.ed. Toronto, Ontario: Allyn & Bacon, 2006.

THE-STANDISH-GROUP. **The CHAOS Report (1994)**. 60 State Street, Suite 700, Boston, MA, 02109, USA: The Standish Group International, Inc., 2004. Technical Report, Information available in the Internet at www.standishgroup.com. Site last visited on November 24th, 2007.

THOMSON, A. J.; MARTINET, A. V. **A Practical English Grammar**. Oxford, UK: Oxford University Press, 1986.

TURNER, J. R. **The Handbook of Project-based Management**. 3rd.ed. Dubuque, IA, USA: McGraw-Hill Professional, 2008.

TURNER, M. **MSF Risk Management Process**. Redmond, Washington: Microsoft Press, 2006. 110-139p.

VENABLES, W.; RIPLEY, B. **Modern Applied Statistics with S**. 4th.ed. [S.l.]: Springer, 2003.

VITHARANA, P. Risks and Challenges of Component-based Software Development. **Communications of the ACM**, New York, NY, USA, v.46, n.8, p.67–72, August 2003.

WALLACE, L.; KEIL, M.; RAI, A. Understanding software project risk: a cluster analysis. **Information & Management**, Amsterdam, Netherlands, v.42, n.1, p.115–125, December 2004.

WESTERMAN, G.; HUNTER, R. **IT Risk**: turning business threats into competitive advantage. Boston, USA: Harvard Business School Press, 2007.

WISHART, D. K-Means Clustering with Outlier Detection, Mixed Variables and Missing Values. In: EXPLORATORY DATA ANALYSIS IN EMPIRICAL RESEARCH: PROCEEDINGS OF 25th ANNUAL CONFERENCE OF THE GER-

MAN CLASSIFICATION SOCIETY, 2001, Munich, Germany. **Anais...** Springer, 2001.

WITTEN, I. H.; FRANK, E. **Data Mining**: practical machine learning tools and techniques. 2nd.ed. San Francisco, CA , USA: Morgan Kaufmann, 2005.

WONG, M. A.; LANE, T. A kth nearest neighbour clustering procedure. **Journal of the Royal Statistical Society.**, Oxford, UK, v.45, n.3, p.362–368, 1983.

YELLEN, J. **Handbook of Graph Theory**. Abingdon, UK: Taylor & Francis, 2007.

8 APÊNDICE

Abaixo mostramos passo a passo a utilização da ferramenta que implementa o algoritmo Deolístico na clusterização dos dados utilizados neste trabalho.

Passo 1: Selecionando o arquivo de dados - A Figura 7.1 mostra como o arquivo de dados deve ser selecionado para posterior análise de cluster.

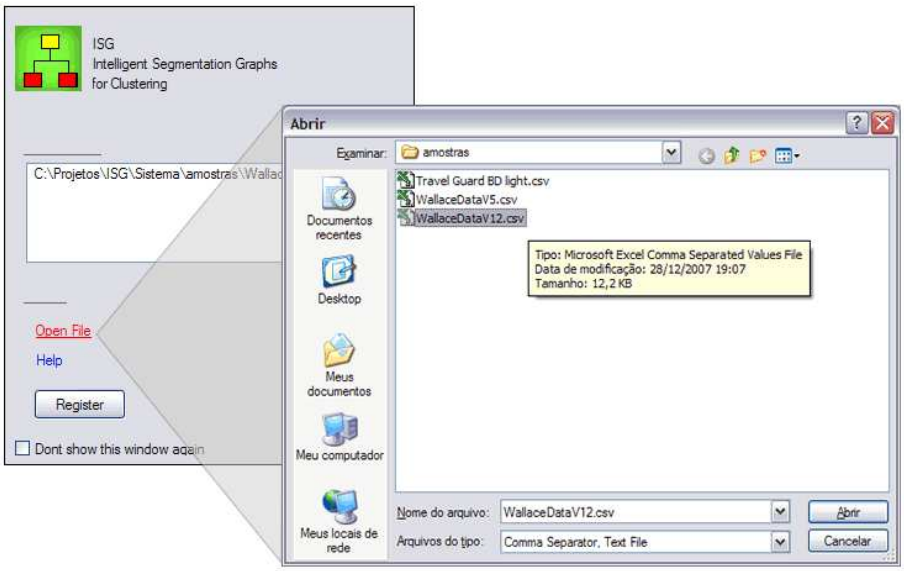


Figura 8.1: Abrindo o arquivo.

Passo 2: Informando os parâmetros do algoritmo - As Figuras 7.2, 7.3 e 7.4 mostram, respectivamente, como as variáveis, o método de clusterização e a medida e distância, e o tamanho da amostra foram selecionados.

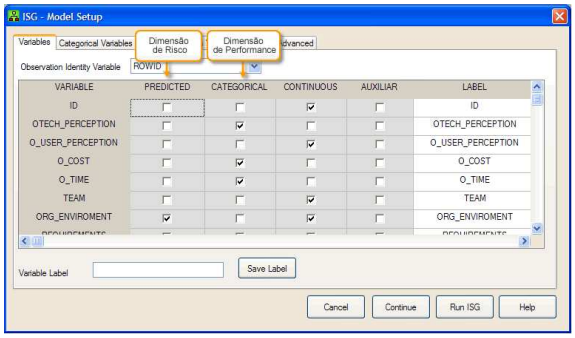


Figura 8.2: Configurando as variáveis.

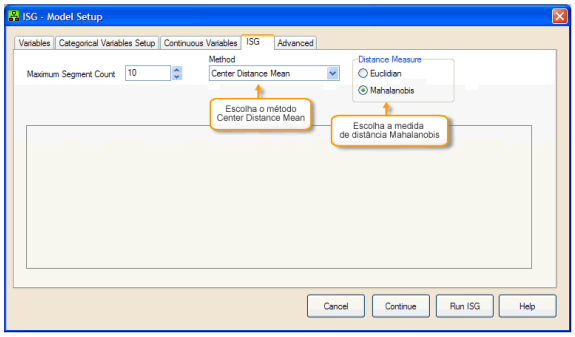


Figura 8.3: Configurando o método e a medida de distância.

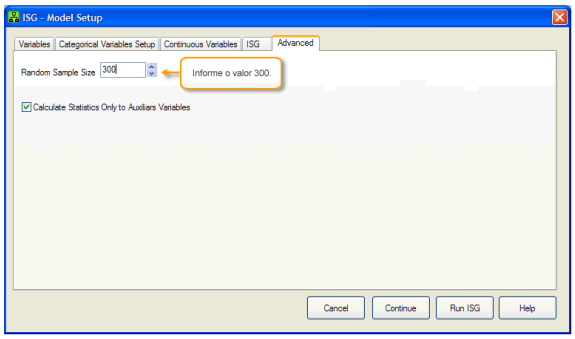


Figura 8.4: Configurando tamanho da amostra.

Passo 3: Execução do algoritmo Deolístico - Para iniciar a execução do algoritmo Deolístico basta pressionar o botão *RUN ISG*. Ver Figura 7.4.

Passo 4: Árvore de Segmentação - Após a execução do algoritmo Deolístico a árvore de segmentação produzido é apresentada automaticamente. Ver Figura 7.5.

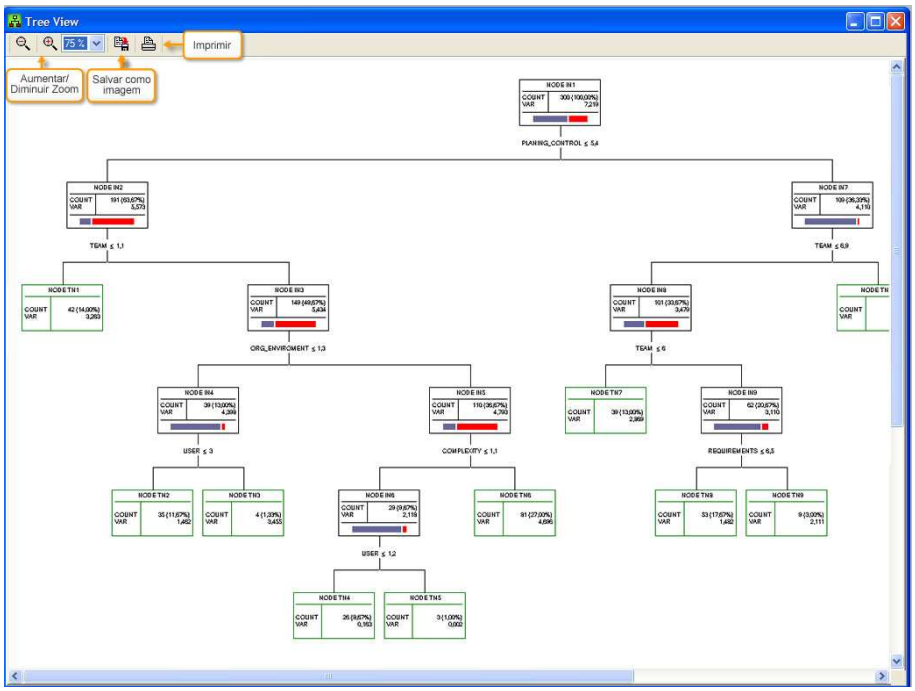


Figura 8.5: Árvore de segmentação.

Passo 5: Segmentos encontrados - Acionando-se a opção *Segmentation Rules* no menu *Report*, na tela principal do *software* (Ver Figura 7.6), o algoritmo encontrou as regras de segmentação descritas na Figura 7.7.

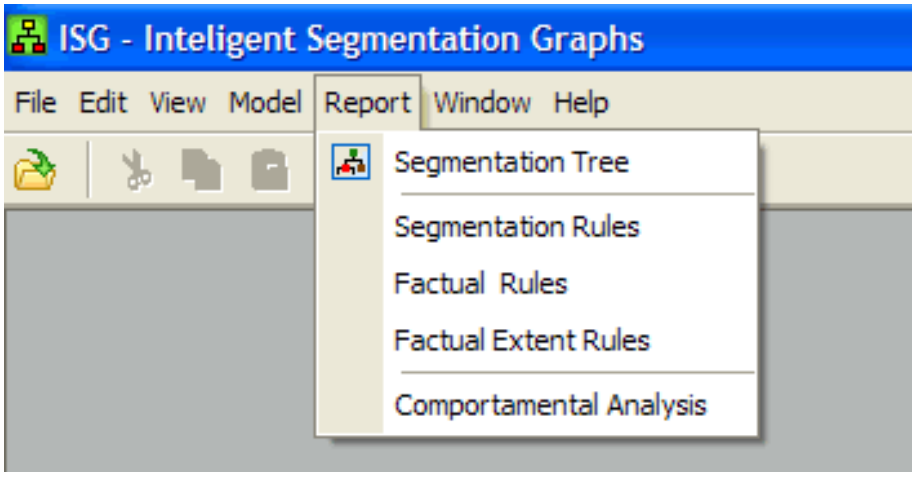


Figura 8.6: Menu Report.

SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL ≤ 5,4 ^ TEAM ≤ 1,1	14,00%	3,26	42
2	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER ≤ 3	11,67%	1,46	35
3	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER > 3	1,33%	3,45	4
4	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER ≤ 1,2	8,67%	0,16	26
5	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER > 1,2	1,00%	0,00	3
6	PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY > 1,1	27,00%	4,70	81
7	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM ≤ 6	13,00%	2,87	39
8	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM > 6 ^ REQUIREMENTS ≤ 6,5	17,67%	1,48	53
9	PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM > 6 ^ REQUIREMENTS > 6,5	3,00%	2,11	9
10	PLANING_CONTROL > 5,4 ^ TEAM > 6,9	2,67%	1,08	8

Close

Figura 8.7: Regras de segmentação.

Passo 6: Regras Factuais - A análise mais detalhada de cada segmento permite que o algoritmo refine as regras de segmentação, expondo com mais precisão as diferenças entre os diversos segmentos encontrados. Isto é possível substituindo as inequações e igualdades por intervalos de valores determinados pelo conteúdo das respectivas variáveis levando-se em consideração o conteúdo de cada segmento. Na Figura 7.8 encontram-se descritas as regras factuais geradas a partir dos dados coletados na Figura 7.1. As regras factuais são geradas a partir da opção *Factual Rules*, no menu *Report*. Ver Figura 7.6.

FACTUAL SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL [1,2..2,3] ^ TEAM [1..1,1]	14,00%	3,26	42
2	PLANING_CONTROL [1,1..2] ^ TEAM [1,3..1,9] ^ ORG_ENVIROMENT [1..1,3] ^ USER [1,2..3]	11,67%	1,46	35
3	PLANING_CONTROL [1,1..2,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [1..1,3] ^ USER [6,7..7]	1,33%	3,45	4
4	PLANING_CONTROL [1,2..1,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [2..2,7] ^ COMPLEXITY [1,1..1,1] ^ USER [1..1,2]	8,67%	0,16	26
5	PLANING_CONTROL [1,1..5] ^ TEAM [1,3..6] ^ ORG_ENVIROMENT [2,3..4,5] ^ COMPLEXITY [1,1..1,1] ^ USER [1,5..6,8]	1,00%	0,00	3
6	PLANING_CONTROL [1..5,4] ^ TEAM [1,3..6,3] ^ ORG_ENVIROMENT [1,5..6,7] ^ COMPLEXITY [1,3..6,8]	27,00%	4,70	81
7	PLANING_CONTROL [5,7..6,8] ^ TEAM [4,7..6]	13,00%	2,87	39
8	PLANING_CONTROL [5,8..7] ^ TEAM [6,3..6,9] ^ REQUIREMENTS [1..6,5]	17,67%	1,48	53
9	PLANING_CONTROL [6,9..6,9] ^ TEAM [6,6..6,6] ^ REQUIREMENTS [6,6..6,6]	3,00%	2,11	9
10	PLANING_CONTROL [6,8..6,8] ^ TEAM [7..7]	2,67%	1,08	8

Figura 8.8: Regras factuais.

Passo 7: Regras Factuais Extendidas - Além das regras factuais que procuram substituir as inequações e igualdades por intervalos de valores determinados pelo conteúdo das respectivas variáveis nos segmentos em questão, as regras factuais extendidas aplicam esta mesma operação para as variáveis envolvidas no processo de segmentação que não estão presentes na regra de segmentação gerada. Na Figura

7.9 encontram-se descritas as regras factuais extendidas geradas. As regras factuais extendidas são geradas a partir da opção *Factual Extended Rules* no menu *Report*. Ver Figura 7.6.

FACTUAL SEGMENTATION RULES

Segment	Rule	Frequency	Variance	Count
1	PLANING_CONTROL [1.2..2.3] ^ TEAM [1..1.1]	14,00%	3,26	42
2	PLANING_CONTROL [1,1..2] ^ TEAM [1.3..1.9] ^ ORG_ENVIROMENT [1..1.3] ^ USER [1.2..3]	11,67%	1,46	35
3	PLANING_CONTROL [1.1..2.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [1..1.3] ^ USER [6.7..7]	1,33%	3,45	4
4	PLANING_CONTROL [1.2..1.6] ^ TEAM [1.3..1.6] ^ ORG_ENVIROMENT [2..2.7] ^ COMPLEXITY [1.1..1.1] ^ USER [1..1.2]	8,67%	0,16	26
5	PLANING_CONTROL [1.1..5] ^ TEAM [1.3..6] ^ ORG_ENVIROMENT [2.3..4.5] ^ COMPLEXITY [1.1..1.1] ^ USER [1.5..6.8]	1,00%	0,00	3
6	PLANING_CONTROL [1..5.4] ^ TEAM [1.3..6.3] ^ ORG_ENVIROMENT [1.5..6.7] ^ COMPLEXITY [1.3..6.8]	27,00%	4,70	81
7	PLANING_CONTROL [5.7..6.8] ^ TEAM [4.7..6]	13,00%	2,87	39
8	PLANING_CONTROL [5.8..7] ^ TEAM [6.3..6.9] ^ REQUIREMENTS [1..6.5]	17,67%	1,48	53
9	PLANING_CONTROL [6.9..6.9] ^ TEAM [5.6..6.6] ^ REQUIREMENTS [6.6..6.6]	3,00%	2,11	9
10	PLANING_CONTROL [6.8..6.8] ^ TEAM [7..7]	2,67%	1,08	8

Figura 8.9: Regras factuais extendidas.

Passo 8: Análise Estatística dos Segmentos ou Análise Comportamental - Para analisarmos como se comportam os indicadores de performance em relação aos fatores de risco para os segmentos encontrados utilizamos a opção *Comportamental Analysis*, no menu Report. Ver Figura 7.6. Os resultados destas análises são apresentadas nas Figuras de 7.10 a 7.19.

Statistics

Segment 1

I - OVERVIEW

COUNT = 42 - 14,00% VARIANCE = 3,26

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM ≤ 1,1

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1.2..2.3] ^ TEAM [1..1.1]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1.2..2.3] ^ TEAM [1..1.1] ^ ORG_ENVIROMENT [1.2.2.5] ^ REQUIREMENTS [1.3.6.8] ^ USER [1.2.7] ^ COMPLEXITY [1.8.6.8]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	6,00	2,00	4,87	0,84	q1	q2	q3
O_USER_PERCEPTION	6,20	2,00	5,66	0,87	q1	q2	q3
O_COST	7,00	2,00	6,07	1,16	q1	q2	q3
O_TIME	7,00	1,00	6,12	1,15	q1	q2	q3

Figura 8.10: Análise comportamental do nó terminal 01.

Statistics

Segment 2

I - OVERVIEW

COUNT = 35 - 11,67% VARIANCE = 1,46

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER ≤ 3

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1,1..2] ^ TEAM [1,3..1,9] ^ ORG_ENVIROMENT [1..1,3] ^ USER [1,2..3]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1,1..2] ^ TEAM [1,3..1,9] ^ ORG_ENVIROMENT [1..1,3] ^ USER [1,2..3] ^ REQUIREMENTS [1,4,1,9] ^ COMPLEXITY [1,3,6,8]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	7,00	4,50	5,57	1,03	q1	q2	q3
O_USER_PERCEPTION	7,00	5,40	6,20	0,37	q1	q2	q3
O_COST	7,00	5,00	6,37	0,55	q1	q2	q3
O_TIME	7,00	5,00	6,37	0,55	q1	q2	q3

Figura 8.11: Análise comportamental do nó terminal 02.

Statistics

Segment 3

I - OVERVIEW

COUNT = 4 - 1,33% VARIANCE = 3,45

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT ≤ 1,3 ^ USER > 3

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1,1..2,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [1..1,3] ^ USER [6,7..7]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1,1..2,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [1..1,3] ^ USER [6,7..7] ^ REQUIREMENTS [6,4,6,9] ^ COMPLEXITY [1,6,8]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	6,00	3,50	5,13	1,18	q1	q2	q3
O_USER_PERCEPTION	3,80	3,00	3,50	0,38	q1	q2	q3
O_COST	5,00	4,00	4,75	0,50	q1	q2	q3
O_TIME	5,00	2,00	4,00	1,41	q1	q2	q3

Figura 8.12: Análise comportamental do nó terminal 03.

Statistics

Segment 4

I - OVERVIEW

COUNT = 26 - 8,67% VARIANCE = 0,16

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER ≤ 1,2

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1,2..1,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [2..2,7] ^ COMPLEXITY [1,1..1,1] ^ USER [1..1,2]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1,2..1,6] ^ TEAM [1,3..1,6] ^ ORG_ENVIROMENT [2..2,7] ^ COMPLEXITY [1,1..1,1] ^ USER [1..1,2] ^ REQUIREMENTS [1,1,1,4]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	7,00	3,00	6,71	0,81	q1	q2	q3
O_USER_PERCEPTION	7,00	4,00	6,78	0,59	q1	q2	q3
O_COST	7,00	3,00	6,58	0,86	q1	q2	q3
O_TIME	7,00	2,00	6,54	1,03	q1	q2	q3

Figura 8.13: Análise comportamental do nó terminal 04.

Statistics

Segment 5

I - OVERVIEW

COUNT = 3 - 1,00% VARIANCE = 0,00

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY ≤ 1,1 ^ USER > 1,2

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1,1..5] ^ TEAM [1,3..6] ^ ORG_ENVIROMENT [2,3..4,5] ^ COMPLEXITY [1,1..1,1] ^ USER [1,5..6,8]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1,1..5] ^ TEAM [1,3..6] ^ ORG_ENVIROMENT [2,3..4,5] ^ COMPLEXITY [1,1..1,1] ^ USER [1,5..6,8] ^ REQUIREMENTS [1,6,6,8]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	6,00	4,50	5,33	0,76	q1	q2	q3
O_USER_PERCEPTION	4,80	3,20	3,93	0,81	q1	q2	q3
O_COST	5,00	4,00	4,33	0,58	q1	q2	q3
O_TIME	5,00	4,00	4,33	0,58	q1	q2	q3

Figura 8.14: Análise Comportamental do Nó Terminal 05.

Statistics

Segment 6

I - OVERVIEW

COUNT = 81 - 27,00% VARIANCE = 4,70

II - SEGMENTATION RULES

PLANING_CONTROL ≤ 5,4 ^ TEAM > 1,1 ^ ORG_ENVIROMENT > 1,3 ^ COMPLEXITY > 1,1

III - FACTUAL EXTENT RULES

PLANING_CONTROL [1..5,4] ^ TEAM [1,3..6,3] ^ ORG_ENVIROMENT [1,5..6,7] ^ COMPLEXITY [1,3..6,8]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [1..5,4] ^ TEAM [1,3..6,3] ^ ORG_ENVIROMENT [1,5..6,7] ^ COMPLEXITY [1,3..6,8] ^ REQUIREMENTS [1,1,6,9] ^ USER [1,7]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	7,00	3,50	5,34	0,90	q1	q2	q3
O_USER_PERCEPTION	6,80	3,00	4,98	1,17	q1	q2	q3
O_COST	7,00	2,00	5,10	1,00	q1	q2	q3
O_TIME	7,00	2,00	5,07	1,05	q1	q2	q3

Figura 8.15: Análise comportamental do nó terminal 06.

Statistics

Segment 7

I - OVERVIEW

COUNT = 39 - 13,00% VARIANCE = 2,87

II - SEGMENTATION RULES

PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM ≤ 6

III - FACTUAL EXTENT RULES

PLANING_CONTROL [5,7..6,8] ^ TEAM [4,7..6]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [5,7..6,8] ^ TEAM [4,7..6] ^ ORG_ENVIROMENT [4,5,6,8] ^ REQUIREMENTS [1,6,8] ^ USER [1,6,8] ^ COMPLEXITY [1,1,6,6]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	5,50	1,00	3,94	1,24	q1	q2	q3
O_USER_PERCEPTION	5,80	1,00	3,76	1,26	q1	q2	q3
O_COST	5,00	1,00	3,74	1,37	q1	q2	q3
O_TIME	5,00	1,00	3,95	1,12	q1	q2	q3

Figura 8.16: Análise comportamental do nó terminal 07.

Statistics

Segment 7

I - OVERVIEW

COUNT = 39 - 13,00% VARIANCE = 2,87

II - SEGMENTATION RULES

PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM ≤ 6

III - FACTUAL EXTENT RULES

PLANING_CONTROL [5,7..6,8] ^ TEAM [4,7..6]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [5,7..6,8] ^ TEAM [4,7..6] ^ ORG_ENVIROMENT [4,5,6,8] ^ REQUIREMENTS [1,6,8] ^ USER [1,6,8] ^ COMPLEXITY [1,1,6,6]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	5,50	1,00	3,94	1,24	q1	q2	q3
O_USER_PERCEPTION	5,80	1,00	3,76	1,26	q1	q2	q3
O_COST	5,00	1,00	3,74	1,37	q1	q2	q3
O_TIME	5,00	1,00	3,95	1,12	q1	q2	q3

Figura 8.17: Análise comportamental do nó terminal 08.

Statistics

Segment 8

I - OVERVIEW

COUNT = 53 - 17,67% VARIANCE = 1,48

II - SEGMENTATION RULES

PLANING_CONTROL > 5,4 ^ TEAM ≤ 6,9 ^ TEAM > 6 ^ REQUIREMENTS ≤ 6,5

III - FACTUAL EXTENT RULES

PLANING_CONTROL [5,8..7] ^ TEAM [6,3..6,9] ^ REQUIREMENTS [1..6,5]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [5,8..7] ^ TEAM [6,3..6,9] ^ REQUIREMENTS [1..6,5] ^ ORG_ENVIROMENT [4,3,6,8] ^ USER [1,7] ^ COMPLEXITY [1,1,7]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	5,50	1,00	3,75	1,08	q1	q2	q3
O_USER_PERCEPTION	5,20	1,00	3,57	1,20	q1	q2	q3
O_COST	5,00	1,00	3,82	1,27	q1	q2	q3
O_TIME	5,00	1,00	3,77	1,34	q1	q2	q3

Figura 8.18: Análise comportamental do nó terminal 09.

Statistics

Segment 10

I - OVERVIEW

COUNT = 8 - 2,67% VARIANCE = 1,08

II - SEGMENTATION RULES

PLANING_CONTROL > 5,4 ^ TEAM > 6,9

III - FACTUAL EXTENT RULES

PLANING_CONTROL [6,8..6,8] ^ TEAM [7..7]

IV - EXTENT FACTUAL SEGMENTATION RULES

PLANING_CONTROL [6,8..6,8] ^ TEAM [7..7] ^ ORG_ENVIROMENT [7,7] ^ REQUIREMENTS [6,9,6,9] ^ USER [6,8,6,8] ^ COMPLEXITY [7,7]

V - CONTINUOUS VARIABLES

VARIABLE	STATISTICS						
	MAX	MIN	MEAN	STDDEV	QT1	QT2	QT3
OTECH_PERCEPTION	1,00	1,00	1,00	0,00	q1	q2	q3
O_USER_PERCEPTION	2,00	1,00	1,13	0,35	q1	q2	q3
O_COST	1,00	1,00	1,00	0,00	q1	q2	q3
O_TIME	1,00	1,00	1,00	0,00	q1	q2	q3

Figura 8.19: Análise comportamental do nó terminal 10.