PONTIFÍCIA UNIVERSIDADE CATÓLICA DO RIO GRANDE DO SUL
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

PROCESSO DE INDUÇÃO E RANQUEAMENTO DE ÁRVORES DE DECISÃO SOBRE MODELOS OLAP

Peterson Fernandes Colares

Dissertação apresentada como requisito parcial à obtenção do grau de Mestre em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul.

Orientador: Prof. Dr. Duncan Dubugras Alcoba Ruiz

Porto Alegre
2011

Dados Internacionais de Catalogação na Publicação (CIP)

C683p Colares, Peterson Fernandes
Processo de indução e ranqueamento de árvores de decisão sobre modelos OLAP / Peterson Fernandes Colares. – Porto Alegre, 2011.
109 p.

Diss. (Mestrado) – Fac. de Informática, PUCRS.
Orientador: Prof. Dr. Duncan Dubugras Alcoba Ruiz.

1. Informática. 2. Mineração de Dados (Informática). 3. Sistema de Apoio à Decisão (Informática). 4. Data Warehouse. 5. Sistemas de Informação. I. Ruiz, Duncan Dubugras Alcoba. II. Título.

CDD 005.72

Ficha Catalográfica elaborada pelo
Setor de Tratamento da Informação da BC-PUCRS

Pontifícia Universidade Católica do Rio Grande do Sul
FACULDADE DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

TERMO DE APRESENTAÇÃO DE DISSERTAÇÃO DE MESTRADO

Dissertação intitulada "**Processo de Indução e Ranqueamento de Árvores de Decisão Sobre Modelos Olap**", apresentada por Peterson Fernandes Colares, como parte dos requisitos para obtenção do grau de Mestre em Ciência da Computação, Sistemas de Informação, aprovada em 30/03/2010 pela Comissão Examinadora:

Prof. Dr. Duncan Dubugras Alcoba Ruiz – Orientador — PPGCC/PUCRS

Profa. Dra. Renata Vieira - — PPGCC/PUCRS

Prof. Dr. Carlos Alberto Heuser – — UFRGS

Homologada em 10./01./2012, conforme Ata No. 002. pela Comissão Coordenadora.

Prof. Dr. Fernando Gehm Moraes
Coordenador.

PUCRS
Campus Central
Av. Ipiranga, 6681 – P32– sala 507 – CEP: 90619-900
Fone: (51) 3320-3611 – Fax (51) 3320-3621
E-mail: ppgcc@pucrs.br
www.pucrs.br/facin/pos

“Se você acha que a instrução é cara,
experimente a ignorância.”

Benjamin Franklin (1706 - 1790)

AGRADECIMENTOS

Agradeço a minha família, em especial a minha esposa Juliana, pelo apoio e incentivo incondicional e principalmente por ter compreendido minhas ausências nestes últimos dois anos.

Agradeço também ao meu orientador, Prof. Duncan, por ter acreditado no meu potencial e colaborado inúmeras vezes para que eu não perdesse o foco na pesquisa. Ao Duncan também meu muito obrigado pelo conhecimento e experiência adquirida no estágio de docência.

Aos colegas do grupo de pesquisa GPIN, em especial o Christian, a Ana, a Patrícia e o Luciano que, sempre com a maior boa vontade, me deram o apoio necessário para que eu conseguisse concluir o mestrado.

Por fim, as demais pessoas que de alguma forma colaboraram com esta tão importante etapa, meu muito obrigado.

RESUMO

Organizações atuantes nos mais diferentes mercados, têm utilizado os benefícios oferecidos pela utilização de técnicas de *Data Mining* – DM como atividades complementares a seus sistemas de apoio a decisão estratégica. Porém, para a grande maioria das organizações, a implantação de um projeto de DM acaba sendo inviabilizada em função de diferentes fatores como: duração do projeto, custos elevados e principalmente pela incerteza quanto à obtenção de resultados que possam auxiliar de fato a organização a melhorar seus processos de negócio. Neste contexto, este trabalho apresenta um processo, baseado no processo de *Knowledge Discovery in Database* – KDD*,* que visa identificar oportunidades para aplicação de técnicas de DM através da indução e ranqueamento de árvores de decisão geradas pela exploração semiautomática de modelos *On-Line Analytical Processing* - OLAP. O processo construído utiliza informações armazenadas em um modelo OLAP preparado com base nas informações utilizadas por sistemas de *Customer Relationship Management* - CRM e *Business Intelligence* – BI, tipicamente utilizados por organizações no apoio a tomada de decisão estratégica. Neste trabalho é apresentada uma série de experimentos, gerados de forma semiautomática, utilizando técnicas de DM, cujos resultados são coletados e armazenados para posterior avaliação e ranqueamento. O processo foi construído e testado com um conjunto significativo de experimentos e posteriormente avaliado por especialistas de negócio em uma instituição financeira de grande porte onde esta pesquisa foi desenvolvida.

Palavras chave: Descoberta de Conhecimento em Banco de Dados, Mineração de Dados, OLAP, Gestão de Relacionamento com Cliente, Inteligência de Negócios.

ABSTRACT

Organizations acting on several markets have been using the benefits offered by the use of Data Mining - DM techniques as a complementary activity to their support systems to the strategic decision. However, to the great majority of the organizations, the deployment of a DM Project ends up not being feasible due to different factors, such as: Project duration, high costs and mainly by the uncertainty as to getting results that may effectively help the organization to improve their business processes. In this context, this paper presents a process based on the process of knowledge Discovery in Database - KDD which aims to identify opportunities to the application of DM techniques through the induction and ranking of decisions generated by the exploration of semi automatic Online Analytical Processing Models-OLAP. The built process uses stored information in a OLAP model prepared on the basis of used information by Customer Relationship Management - CRM and Business Intelligence - BI typically used by the organization to support strategic decision making. In relation to the information selected for this research, it has been carried out in a semi automatic way, a series of experiments using DM techniques which the results are collected and stored for later evaluation and ranking. The process was built and tested with a significant number of experiments and later evaluated by business experts in a large financial institution where this research was developed.

Keywords: Knowledge Discovery in Database, Data Mining, OLAP, Customer Relationship Management, Business Intelligence.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ARFF	*Attribute-Relation File Format*
BI	*Business Intelligence*
CRM	*Customer Relationship Management*
CSV	*Comma-separated values*
DM	*Data Mining*
DW	*Data Warehouse*
ETL	*Extract Transform Load*
FN	Falso Negativo
FP	Falso Positivo
KDD	*Knowledge Discovery in Database*
OLAP	*On-Line Analytical Processing*
OLTP	*Online Transaction Processing*
SAD	Sistemas de Apoio a Decisão
SQL	*Structured Query Language*
VN	Verdadeiro Negativo
VP	Verdadeiro Positivo
UML	*Unified Modeling Language*

SUMÁRIO

1- INTRODUÇÃO

A utilização de Sistemas de Apoio a Decisão – SAD é uma prática comum em organizações de grande porte. Dentre estes sistemas destacamos o uso dos Sistemas de *Customer Relationship Management* - CRM [SWI01] e *Business Intelligence* – BI [TUR09] por parte de instituições financeiras para obtenção de vantagens em um mercado cada vez mais competitivo. Sistemas de CRM, em especial, são utilizados pelas organizações na gestão de relacionamento com seus clientes permitindo uma maior assertividade na oferta de produtos e serviços. Utilizados também como fonte de informação para sistemas de CRM e BI, os modelos *On-Line Analytical Processing* - OLAP [KIM02] são amplamente utilizados no apoio a tomada de decisão estratégica nas organizações.

Na busca por informações que possam gerar diferenciais competitivos, organizações atuantes nos mais diferentes mercados têm utilizado técnicas de *Data Mining* – DM, que é parte integrante de um processo mais abrangente comumente conhecido como *Knowledge Discovery in Database* - KDD [FAY96] [HAN01], como complementares aos tradicionais SADs. Entretanto, investimentos em projetos de KDD ainda não são amplamente realizados principalmente em função de fatores como: custos elevados, longa duração do projeto e principalmente pela incerteza quanto à obtenção de resultados.

Neste contexto, este trabalho apresenta um processo, baseado no processo de KDD, que visa identificar oportunidades para aplicação de técnicas de DM através da indução e ranqueamento de árvores de decisão geradas pela exploração semiautomática de modelos *On-Line Analytical Processing* – OLAP. O processo construído é executado com o apoio de um ferramental construído para automatizar a execução dos experimentos com algoritmos de mineração de dados sobre uma base de dados modelada e populada com informações utilizadas por sistemas de CRM e BI da organização. O processo foi construído, testado com um conjunto significativo de experimentos e avaliado por especialistas de negócio em uma instituição financeira de grande porte onde esta pesquisa foi desenvolvida.

Para uma melhor compreensão, este trabalho está organizado da seguinte forma: no Capítulo 2, apresentamos uma revisão sobre o referencial teórico onde abordamos questões relacionadas aos dados, suas formas de apresentação e tratamento. Abordamos ainda o armazenamento de dados em grandes repositórios bem como o processo de KDD. No Capítulo 3, detalhamos o cenário onde esta pesquisa foi desenvolvida, seus objetivos e características, bem como o processo de exploração semiautomática proposto. No Capítulo 4, apresentamos o processo de KDD realizado, o ferramental desenvolvido, os experimentos executados e as avaliações dos especialistas de negócio. No Capítulo 5, discorremos sobre os trabalhos relacionados com esta pesquisa. Por fim, no Capítulo 6, apresentamos as conclusões e os trabalhos futuros e, em seguida, as referências bibliográficas.

2- REFERENCIAL TEÓRICO

Neste capítulo é realizada uma revisão bibliográfica dividida em duas partes. Na primeira, apresentamos questões referentes aos dados, formas de representação, tratamento e armazenamento. Abordamos uma técnica de armazenamento de dados conhecida como *Data Warehouse* – DW, amplamente utilizada no processo de *Knowledge Discovery in Database* - KDD. Na segunda parte o processo de KDD é apresentado, são mostradas as principais etapas deste processo e destacadas quatro das principais técnicas de mineração de dados.

2.1 Dados

Os conjuntos de dados diferem de diversas formas. Por exemplo, os atributos usados para descrever objetos podem ser de diferentes tipos – quantitativos ou qualitativos – e os conjuntos de dados podem ter características especiais como séries temporais ou objetos com relacionamentos explícitos entre si. É comum que o tipo de dado determine quais ferramentas e técnicas devam ser usadas no processo de análise de dados [TAN06]. No tocante a qualidade de dados, cabe observar que a maioria das técnicas de mineração de dados tolera algum nível de imperfeição nos dados. Entretanto, operações que visam melhorar a qualidade dos dados colaboram para análises mais qualificadas. Nas próximas seções são descritos alguns dos principais tipos de dados bem como destacadas questões relacionadas à qualidade destes.

2.1.1 Tipo de Dados

Um conjunto de dados pode ser visto como uma coleção de objetos de dados, registros, entidades, etc. Objetos de dados são representados por um determinado número de atributos que descrevem as características básicas como tamanho, altura, idade, peso, etc.

Um atributo é uma propriedade ou característica de um objeto que pode variar de um objeto para outro ou em função do tempo [TAN06]. Dois exemplos clássicos de atributos são a cor dos olhos que varia de pessoa para pessoa e o peso que, além de variar de pessoa para pessoa, pode variar em função do tempo para uma mesma pessoa. Estes atributos podem representar informações de diferentes tipos e podem ser classificados como ordinal, nominal, intervalo e proporção. As operações que podem ser utilizadas sobre estes atributos são:

- Distinção: = e ≠
- Ordenação: <, ≤, > e ≥
- Adição: + e –
- Multiplicação: * e /

A Tabela 1 descreve estes quatro tipos de atributos bem como as operações que podem ser realizadas com estes.

Tabela 1 - Tipos de Atributos. Adaptado de [TAN06]

Tipo do Atributo		**Descrição**	**Exemplos**	**Operações**
Categorização (Qualitativos)	Nominal	Os valores de um atributo nominal são apenas nomes diferentes que fornece informação suficiente para distinguir um objeto do outro.	Códigos postais, números de matrículas, cor dos olhos, sexo.	Modo, entropia, correlação de contingência, teste x^2.
	Ordinal	Os valores de um atributo ordinal fornecem informações suficientes para ordenar objetos. (>, <)	Qualidade {bom, melhor, pior}, notas, números de ruas.	Medianas, porcentagens, testes de execução, testes de assinatura
Numéricos (Quantitativos)	Intervalar	Para atributos intervalares, as diferenças entre os valores são significativas e existe uma unidade de medida. (+, -)	Datas de calendário, temperatura.	Média, desvio padrão, correlação de Pearson, testes T e F.
	Proporcional	Para variáveis proporcionais, tanto as diferenças quanto as proporções são significativas. (*, /)	Quantidades monetárias, contadores, idades, comprimento.	Média geométrica, média harmônica, variação percentual.

Atributos nominais e ordinais são chamados de atributos categorizados ou qualitativos. Já os atributos intervalares e proporcionais são chamados de quantitativos ou numéricos. Os tipos de atributos também podem ser descritos em termos de transformações que não alteram o significado dos mesmos

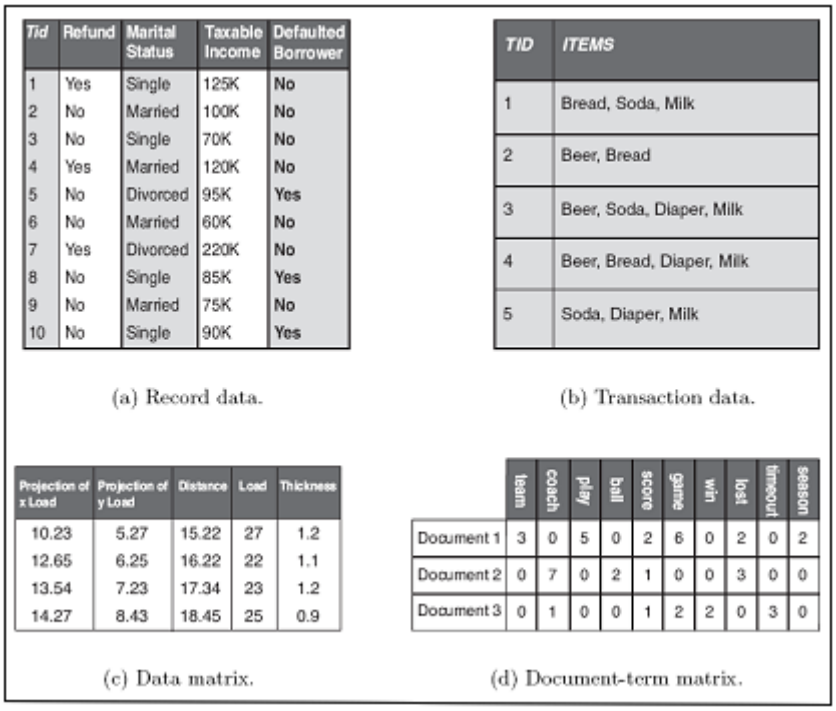
[TAN06]. Uma forma independente de distinguir atributos é pelo número de valores que eles podem receber tais como:

- **Discretos -** um atributo discreto possui um conjunto de valores finito. São exemplos de atributos discretos o CEP, matrícula, ID de um produto, etc.
- **Contínuos -** um atributo contínuo é representado por um número real. São exemplos de atributos contínuos a temperatura, altura, peso, etc.

2.1.1.1 Tipo de Conjuntos de Dados

Existem muitos tipos de conjuntos de dados que são utilizados no processo de mineração nas mais variadas áreas. Porém, no escopo desta pesquisa, destacamos apenas o tipo de dados de registros e suas características gerais de dimensão, dispersão e resolução.

Grande parte do trabalho de mineração de dados supõe que o conjunto de dados seja uma coleção de registros, cada registro formado por um conjunto fixo de atributos, como ilustrado na Figura 1(a). Na forma mais básica de um dado em registro, não há relacionamento explícito entre registros ou atributos. Dados em registro são geralmente armazenados em arquivos ou bancos de dados relacionais. A Figura 1 ilustra diferentes tipos de dados em registros que são detalhados na sequência.



Tid	Refund	Marital Status	Taxable Income	Defaulted Borrower
1	Yes	Single	125K	No
2	No	Married	100K	No
3	No	Single	70K	No
4	Yes	Married	120K	No
5	No	Divorced	95K	Yes
6	No	Married	60K	No
7	Yes	Divorced	220K	No
8	No	Single	85K	Yes
9	No	Married	75K	No
10	No	Single	90K	Yes

(a) Record data.

TID	ITEMS
1	Bread, Soda, Milk
2	Beer, Bread
3	Beer, Soda, Diaper, Milk
4	Beer, Bread, Diaper, Milk
5	Soda, Diaper, Milk

(b) Transaction data.

Projection of x Load	Projection of y Load	Distance	Load	Thickness
10.23	5.27	15.22	27	1.2
12.65	6.25	16.22	22	1.1
13.54	7.23	17.34	23	1.2
14.27	8.43	18.45	25	0.9

(c) Data matrix.

	team	coach	play	ball	score	game	win	lost	timeout	season
Document 1	3	0	5	0	2	6	0	2	0	2
Document 2	0	7	0	2	1	0	0	3	0	0
Document 3	0	1	0	0	1	2	2	0	3	0

(d) Document-term matrix.

Figura 1 - Tipos de dados em registros
Fonte: [TAN06]

- **Dados de transação** – dados de transação são um tipo especial de dados em registros, onde cada registro (transação) envolve um conjunto de itens. Um exemplo adequado para representar este tipo de dado é um cupom fiscal de compras realizadas em um estabelecimento comercial. Este tipo de dado está representado na Figura 1(b).

- **Matriz de dados** – se os objetos de dados em uma coleção possuem o mesmo conjunto de atributos numéricos, então estes dados podem ser representados por uma matriz *m* por *n*, onde existem *m* linhas, uma para cada objeto, e *n* colunas, uma para cada atributo. Esta matriz é chamada de matriz de dados que, por permitir operações de matrizes aplicadas nas transformações de dados, são utilizadas como formato padrão de dados para

a maioria dos dados estatísticos. Este tipo de dado está representado na Figura 1(c).

- **Matriz de dados dispersos** – uma matriz de dados dispersos é um caso especial de matriz de dados no qual os atributos possuem o mesmo tipo e são assimétricos, onde somente os valores diferentes de zero são significativos. Este tipo de matriz é muito utilizado em mineração de textos onde as linhas da matriz representam os documentos e as colunas representam as palavras com o número de ocorrências das mesmas em cada documento. Este tipo de dado está representado na Figura 1(d).

2.1.2 Qualidade dos Dados

Algoritmos de mineração de dados são muitas vezes aplicados a dados que foram coletados para outros propósitos. Por este motivo, a mineração de dados envolve etapas de detecção e correção de problemas relacionados à qualidade dos dados além de permitir a utilização de algoritmos que toleram dados com baixa qualidade. A etapa de detecção e correção é conhecida como limpeza dos dados. Neste tópico abordamos questões referentes à qualidade de dados que podem ser identificadas já na coleta das informações. Também destacamos problemas que podem ser encontrados em dados previamente armazenados.

2.1.2.1 Erros de Medição e Coleta

O termo erro de medição refere-se a qualquer problema resultante do processo de medição, como no caso onde um valor registrado diferente do valor real. Para atributos contínuos, a diferença numérica entre o valor medido e o valor real é chamada de erro. O termo erro de coleção de dados refere-se a erros com a omissão de objetos de dados ou valores de atributos, ou a inclusão

inapropriada de um objeto de dados. Existem certos tipos de erros de dados que são encontrados com frequência e, para estes, existem técnicas desenvolvidas para detecção e correção.

2.1.2.2 Precisão, Foco e Exatidão

A precisão é muitas vezes medida pelo desvio padrão de um conjunto de valores, enquanto que o foco é medido utilizando a diferença da média do conjunto de valores e o valor conhecido da quantidade que está sendo medida. O foco só pode ser determinado para objetos cuja quantidade medida é conhecida através de meios externos a situação corrente. A exatidão depende da precisão e do foco e, para esta, deve-se utilizar dígitos significativos. O objetivo é usar apenas tantos dígitos para representar o resultado de uma medição ou cálculo quanto for justificado pela precisão de dados. Cabe ainda observar que questões como dígitos significativos, precisão, foco e exatidão são às vezes negligenciados, mas muito importantes para a mineração de dados. Muitas vezes conjuntos de dados não apresentam informações sobre a precisão dos mesmos. Porém, sem uma compreensão sobre a exatidão dos dados e dos resultados, analistas correm o risco de cometer grandes erros na análise de dados.

2.1.2.3 Problemas com a Qualidade dos Dados

Segundo [TAN06], diferentes problemas com a qualidade dos dados podem ser identificados em dados previamente armazenados para posterior mineração. Para tratar estas questões, destacamos as características destes e algumas técnicas que podem ser úteis no tratamento e correção destas informações.

2.1.2.3.1 *Outliers*

Outliers são objetos que, de alguma forma, apresentam características diferentes da maioria ou ainda atributos que apresentam valores

incomuns com relação aos valores típicos para este. Diferentes definições para *outliers* são apresentadas pelas comunidades de estatística e mineração de dados. No entanto, ambas afirmam que estes objetos não devem ser confundidos com erros e que, dependendo do foco da mineração, estas informações são muito importantes como no caso de detecção de anomalias.

2.1.2.3.2 Valores Ausentes

Diversas estratégias podem ser utilizadas para lidar com valores ausentes em um conjunto de dados sendo que cada uma pode ser melhor indicada em determinadas circunstâncias. Na sequência, destacamos algumas destas estratégias juntamente com suas vantagens e desvantagens.

- Eliminar objetos ou atributos - uma estratégia simples e eficaz é a eliminação de objetos com valores ausentes. Entretanto, deve-se levar em conta que estes objetos, mesmo com valores ausentes, possuem informações que podem ser relevantes. Neste caso, se muitos objetos apresentam estas características, uma análise confiável pode ser comprometida. No caso de conjunto de dados que apresentam apenas alguns objetos com valores ausentes uma boa estratégia seria a remoção destes objetos observando que esta deve ser adotada com cautela em função de que os objetos eliminados podem ser significativos para a análise.

- Eliminar valores ausentes - em muitos casos dados ausentes podem ser estimados confiavelmente. Como exemplo, podemos supor um conjunto de dados com muitos valores semelhantes. Nesta situação, valores dos atributos mais próximos do ponto com valores ausentes são utilizados para estimar o valor ausente. No caso de atributos contínuos, utiliza-se a média dos valores dos

atributos mais próximos e, para o caso de atributos categorizados, utiliza-se o valor com maior frequência.

- Valores inconsistentes - dados com valores inconsistentes são normalmente inseridos no conjunto de dados durante a aquisição, principalmente quando digitados. Estes, por sua vez, são mais fáceis de identificar como no caso onde, no cadastro de um cliente, o CEP não pertencer à cidade informada pelo mesmo ou no caso de pessoas cuja altura, peso ou idade apresentam valores negativos. Estas inconsistências, quando identificadas, devem ser corrigidas mesmo que, para isso, seja necessário a utilização de informações adicionais.

2.1.2.3.3 Dados Duplicados

Um conjunto de dados pode incluir objetos duplos e, para detectar e eliminar tais objetos, duas questões dever ser observadas. Primeiro, se houver dois objetos que realmente representam um único, então os valores dos atributos correspondentes podem diferir e estes, se diferentes, devem ser ajustados. Segundo, deve-se tomar cuidado para não combinar acidentalmente objetos semelhantes, como no caso de pessoas distintas com o mesmo nome.

2.1.3 Armazenamento de Dados

Data Warehouse - DW é uma base de dados projetada e modelada para dar suporte à tomada de decisão estratégica. Nesta, são armazenadas informações de diferentes sistemas e bases de dados. [HAN01]. Algumas características importantes diferenciam um DW de uma base de dados transacional:

- Utiliza modelagem multidimensional conhecida como “cubo”, construído para armazenamento de séries históricas.

- Projetado para armazenar um grande volume de informações.
- Agrega informações de diversas origens, de maneira uniforme e consistente.
- Tende a ser não-volátil, ou seja, seus dados não serão perdidos e/ou atualizados ao longo do tempo.
- Não se destina ao armazenamento de informações transacionais diárias.
- Oferece formas extremamente flexíveis de visualização de suas informações.

Na modelagem multidimensional [KIM98], os dados são armazenados em tabelas identificadas como fato e dimensão. As tabelas do tipo fato armazenam valores e medidas que representam um fato da organização. Nas tabelas do tipo dimensão, são armazenados os pontos de observação utilizados pela organização nas análises realizadas sobre os fatos armazenados. Cabe observar que nas dimensões é comum utilizar informações que, quando associadas ao fato, respondem questões como: “O quê?”, ”Quem?”, “Quando?” e “Onde?”. Ainda sobre a modelagem multidimensional, esta pode ser construída de três formas:

- **Modelo Estrela:** apresentado na Figura 2, este modelo caracteriza-se por conter uma única tabela fato ligada a várias tabelas dimensão.
- **Modelo Floco de Neve:** difere-se do modelo estrela por permitir a existência de relacionamentos entre as tabelas dimensão.
- **Modelo Constelação de Fatos:** neste modelo, podem-se encontrar diversas tabelas fato relacionadas com suas

próprias dimensões e, assim como no modelo floco de neve, entre estas dimensões podem existir relacionamentos.

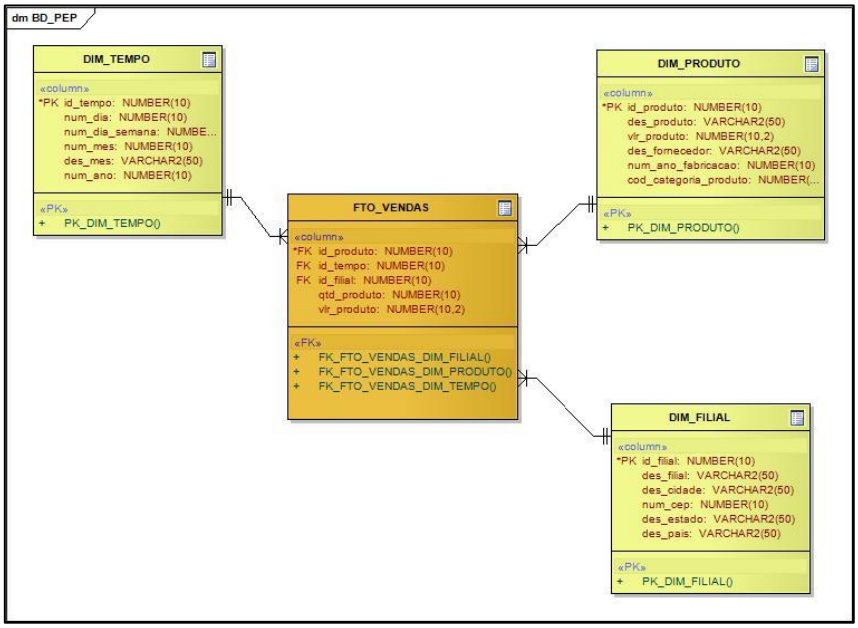


Figura 2 - Modelo Estrela
Fonte: Adaptado de [KIM98]

2.2 Mineração de Dados

Mineração de Dados é parte integrante de um processo mais abrangente comumente conhecido como *Knowledge Discovery in Database* - KDD [FAY96] [HAN01], amplamente utilizado pelas organizações no intuito de descobrir informações relevantes, até então armazenadas, porém desconhecidas, em grandes bases de dados que podem auxiliar seus processos de tomada de decisão. O processo de KDD, segundo [HAN01], está dividido em sete etapas distintas identificadas como:

- **Limpeza:** Esta etapa é responsável pela correção de informações e eliminação de dados inconsistentes.
- **Integração:** Nesta fase é realizada a unificação de informações disponíveis em diferentes fontes de dados. É comum a execução destas duas primeiras etapas em uma

fase de pré-processamento cujo resultado é o armazenamento dos dados em um *Data Warehouse* - DW.

- **Seleção:** É a fase na qual os dados relevantes para o processo de análise são selecionados na base de dados.
- **Transformação:** Etapa onde os dados são transformados e/ou consolidados de acordo com o modelo adequado para o processo de mineração. No caso da utilização de *Data Warehouse*, esta etapa constitui um processamento posterior ao armazenamento dos dados de diferentes fontes.
- **Mineração de Dados:** Basicamente é o processo onde métodos inteligentes de processo de software são aplicados sobre os dados no intuito de identificar determinados padrões de comportamento.
- **Validação:** Nesta etapa são analisados os resultados obtidos com o processo de mineração buscando identificar informações realmente relevantes.
- **Representação:** Etapa responsável por disponibilizar os resultados obtidos de forma adequada para os usuários.

Os resultados obtidos com o processo de KDD dependem, dentre outros fatores, da qualidade dos dados utilizados na etapa de mineração. Segundo [LIR07], estas etapas, que compõem basicamente uma fase de preparação de dados, consomem em torno de 85% de todo o processo de KDD. A Figura 3 representa as etapas do processo bem como a sequência de execução destas dentro do processo de KDD. Tanto as técnicas utilizadas especificamente no processo de mineração quanto à fonte de dados adequada para tal estão detalhadas nas próximas seções.

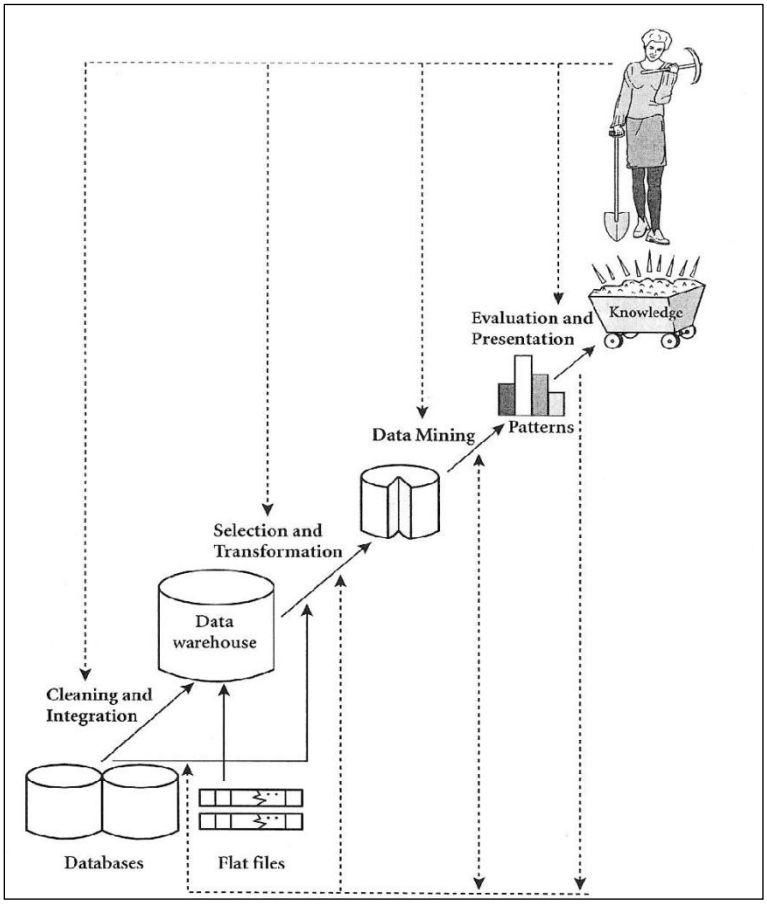


Figura 3 - Etapas do Processo de KDD
Fonte: [HAN01]

2.2.1 Técnicas de Mineração

As técnicas de Mineração de Dados podem ser classificadas em dois grandes grupos: técnicas preditivas e técnicas descritivas [TAN06]. As técnicas preditivas possuem como objetivo prever o valor de uma informação em especial em função do conjunto das demais informações disponíveis para análise. Os termos comumente utilizados para identificar estas variáveis são respectivamente “variável dependente” e “variáveis independentes”. Já as técnicas descritivas possuem o objetivo de encontrar padrões de comportamento ou situações em que se observa fuga de um determinado padrão. Estas são frequentemente utilizadas e associadas a processos de validação e interpretação de seus resultados. A Figura 4 exibe quatro das principais técnicas de mineração detalhadas a seguir.

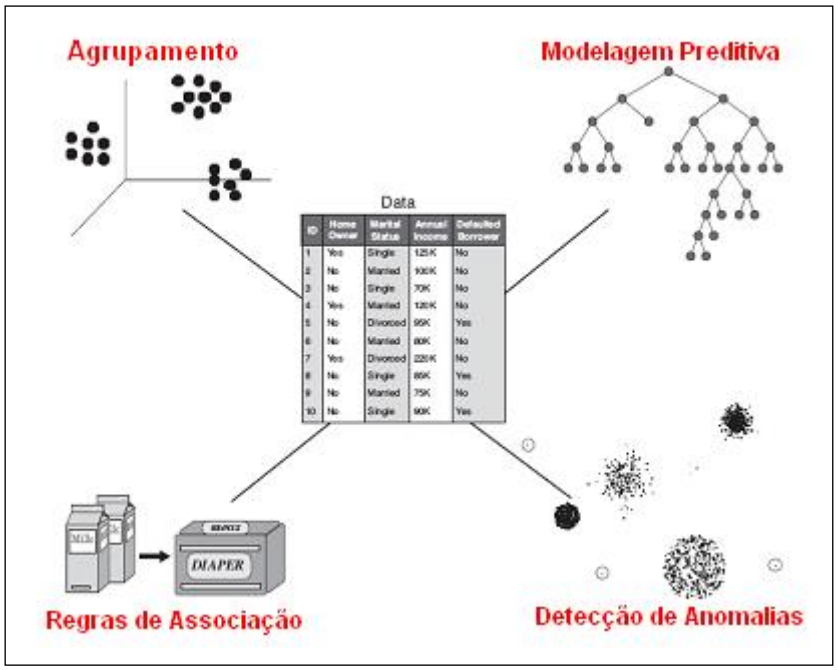


Figura 4 – Técnicas de Mineração de Dados
Fonte: Adaptado de [TAN06]

2.2.1.1 Agrupamento

Esta técnica busca realizar agrupamentos de objetos em função da proximidade dos valores de seus atributos. Esta proximidade é avaliada em função da comparação de atributos de um objeto “A” com os mesmos atributos do objeto “B”. Supondo que um atributo de “A” descreva a qualidade do produto em uma escala de 0 a 5, podemos afirmar que o produto “A”, qualificado como 4 está mais próximo de “B” qualificado como 5 do que de “C” qualificado como 2.

A análise de agrupamentos trabalha sobre dados em que os valores das classes não estão definidos. A tarefa consiste em identificar agrupamentos de objetos, agrupamentos estes que identificam uma classe.

2.2.1.2 Regras de Associação

Esta técnica é utilizada para descobrir condições que ocorrem com frequência em um determinado conjunto de dados. Os padrões encontrados são tipicamente representados sob a forma de regras de implicação.

Regras de Associação são amplamente utilizadas na análise de comportamento de clientes que realizam compras via internet utilizando cestas de compras. Em uma definição mais formal, utiliza associações do tipo “X → Y”, no caso do comportamento de clientes em compras, quando identificado um determinado padrão, a regra de associação pode ser destacada como “se comprou X, então comprou Y”.

2.2.1.3 Detecção de Anomalias

Atributos que indicam anomalias são aqueles que, dentro de um determinado grupo de objetos, fogem totalmente ao padrão de comportamento. A maioria das técnicas de mineração classifica estes objetos como *outliers* e desconsidera os mesmos. Entretanto, quando se está buscando detectar anomalias, os *outliers* são justamente os objetos que se deseja identificar. *Outliers* são identificados através da utilização de testes estatísticos que determinam a distribuição ou a probabilidade do modelo de dados.

2.2.1.4 Modelagem Preditiva

Modelagem preditiva compreende a construção de um modelo para uma determinada variável dependente em função dos valores encontrados nas variáveis exploratórias. A modelagem preditiva é dividida em dois tipos de tarefas:

- **Classificação:** usada quando se está trabalhando com variáveis discretas.
- **Regressão:** utilizada quando se está trabalhando com variáveis contínuas.

Segundo [HAN01], algoritmos utilizados em tarefas de classificação buscam a predição de valores desconhecidos em função de um atributo de interesse, denominado atributo classe, levando a construção de um modelo preditivo. Para que este modelo seja construído, um classificador passa uma etapa conhecida como treinamento onde, para este, é fornecido um conjunto de dados onde o atributo classe é conhecido. Após o treinamento, o classificador é submetido ao conjunto de testes que é constituído por atributos semelhantes aos do conjunto de treino, porém, neste conjunto, o atributo classe não é conhecido. Como resultado destas etapas, temos a construção de um modelo preditivo.

A partir do treinamento, um classificador produz um modelo preditivo que pode ser representado de forma adequada por uma árvore de decisão. Uma árvore de decisão é um grafo cujos nodos internos são compostos pelos atributos explanatórios e os nodos folha representam os valores do atributo classe. O algoritmo C4.5 [QUI96] é um popular algoritmo que implementa árvores de decisão. O software Weka [WEK10], que será utilizado para realização dos experimentos propostos nesta pesquisa, disponibiliza uma implementação do C4.5 identificada como J48. A Figura 5 e a Figura 6 detalham o funcionamento de um classificador baseado em árvore de decisão e a representação dos resultados.

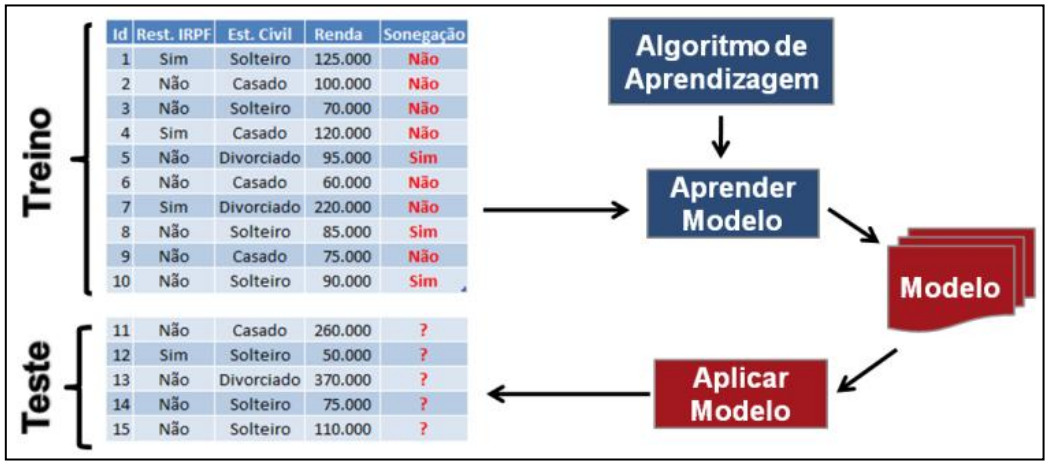
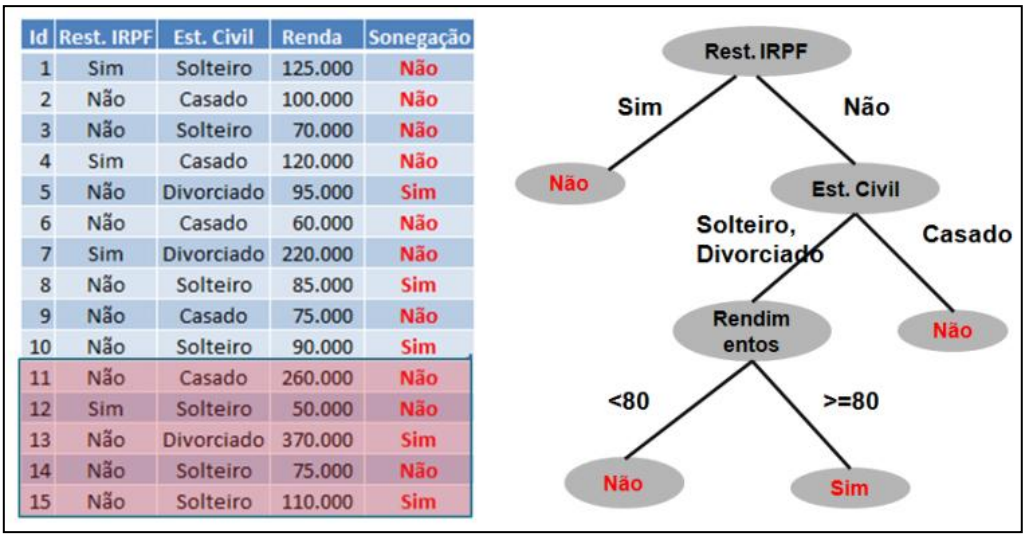


Figura 5 – Funcionamento de um Classificador.

Fonte: Adaptado de [TAN06]



Id	Rest. IRPF	Est. Civil	Renda	Sonegação
1	Sim	Solteiro	125.000	Não
2	Não	Casado	100.000	Não
3	Não	Solteiro	70.000	Não
4	Sim	Casado	120.000	Não
5	Não	Divorciado	95.000	Sim
6	Não	Casado	60.000	Não
7	Sim	Divorciado	220.000	Não
8	Não	Solteiro	85.000	Sim
9	Não	Casado	75.000	Não
10	Não	Solteiro	90.000	Sim
11	Não	Casado	260.000	Não
12	Sim	Solteiro	50.000	Não
13	Não	Divorciado	370.000	Sim
14	Não	Solteiro	75.000	Não
15	Não	Solteiro	110.000	Sim

Figura 6 - Representação dos resultados
Fonte: Adaptado de [TAN06]

O desempenho de um classificador pode ser avaliado por diferentes critérios e, segundo [TAN06] [WIT05], uma das formas mais tradicionais de avaliação é através da matriz de confusão. A matriz de confusão apresenta o resultado para um modelo de classificação binária onde cada entrada da tabela representa o número de registros previstos correta ou incorretamente para cada classe. A

Tabela 2 detalha os resultados apresentados para um problema de duas classes onde temos:

- (VP) – número de instâncias classificadas como verdadeiros positivos;
- (VN) – número de instâncias classificadas como verdadeiros negativos;
- (FP) – número de instâncias classificadas como falso positivo;
- (FN) – número de instâncias classificadas como falso negativo.

Tabela 2 – Matriz de confusão. Adaptado de [WIT05]

Atributo Preditivo		Classe Prevista	
		Classe = 1	Classe = 0
Classe Real	Classe = 1	VP	FN
	Classe = 0	FP	VN

Com base nas informações de uma matriz de confusão, algumas métricas de desempenho podem ser utilizadas para a avaliação de resultados. Neste contexto, apresentamos na Tabela 6, sete métricas que podem ser calculadas com base nas informações da matriz.

Tabela 3 – Métricas utilizadas. Adaptado de [WIT05]

2.2.2 Software para Mineração de Dados

Uma variada gama de ferramentas de mineração de dados podem ser encontradas com facilidade no mercado. Dentre estas, destacamos soluções oferecidas por reconhecidos fornecedores como IBM SPSS Modeler [SPS10],

SAS Enterprise Miner [SAS10], Oracle Data Mining [ODM10], entre outros. Como soluções *open source* destacamos softwares como *o* RapidMiner [RDM10] e, em especial, software Weka [WEK10] utilizado nos experimentos realizados nesta pesquisa.

Embora o software Weka seja capaz de buscar informações diretamente do banco de dados, o formato mais utilizado é um arquivo com a extensão *arff.* A Figura 7 apresenta um exemplo de um arquivo no formato *arff*. Esse arquivo é formado por atributos e instâncias destes onde, a primeira entrada deste arquivo define seu nome (A). Em seguida, são declarados os atributos (B) que compõe o arquivo que podem ser de dois tipos: contínuos ou categóricos. Quando contínuo, deve-se informar o tipo de valor aceito: no exemplo, real. Quando categóricos, é necessário informar todas as entradas válidas para cada atributo: no exemplo, {Solteiro, Casado, Divorciado}. Após a declaração, são inseridas as instâncias (C) do arquivo, obedecendo à ordem declarada dos atributos e separando os mesmos por vírgula. Todas as entradas precedidas do sinal “@” são utilizadas para a correta formatação do arquivo.

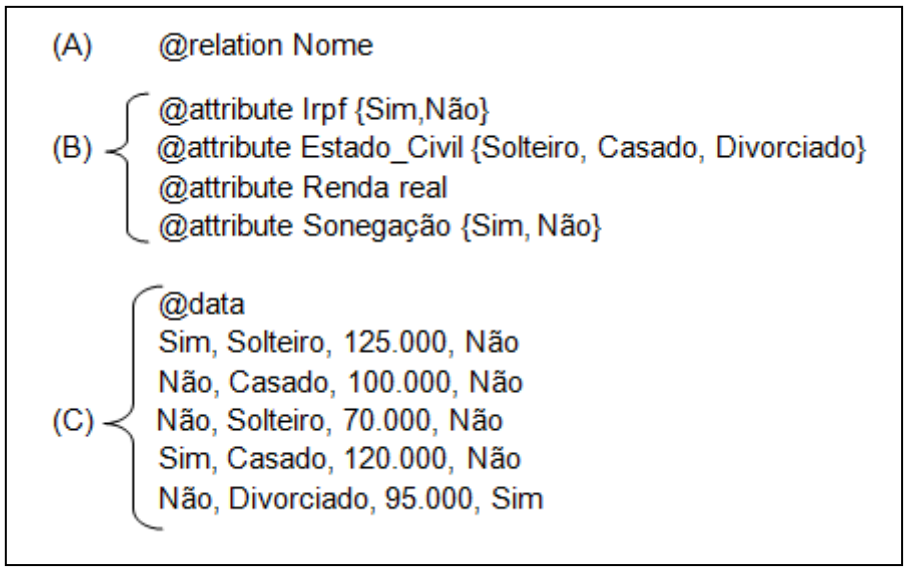


Figura 7 - Formato do arquivo de entrada

2.3 Considerações do Capítulo

Este capítulo apresentou um estudo sobre os assuntos tratados nesta pesquisa. Na seção 2.1, tratamos da caracterização dos dados discorrendo sobre os tipos de dados existentes, as operações possíveis sobre estes e, no tocante a qualidade dos dados, abordamos os problemas que a baixa qualidade dos dados pode trazer para o processo bem como algumas técnicas de tratamento. Ainda nesta seção abordamos o armazenamento de dados em um modelo dimensional, destacamos algumas características das diferentes formas de modelagem multidimensional.

Na seção 2.2, abordamos o processo de mineração de dados onde detalhamos as etapas envolvidas neste processo e discorremos sobre as principais técnicas utilizadas. Ainda nesta seção, apresentamos o formato de entrada para um algoritmo de mineração utilizado pelo software com o qual foram realizados os experimentos detalhados na seção 4.

Cabe observar que os assuntos abordados até então são utilizados como fundamentação para as questões abordadas nos próximos capítulos onde, apresentamos o detalhamento do cenário onde esta pesquisa foi realizada, a caracterização do problema, a contribuição científica e o desenvolvimento desta pesquisa.

3- DESCRIÇÃO DO CENÁRIO

O cenário onde esta pesquisa foi desenvolvida contempla uma organização financeira de grande porte, atuante no mercado nacional, onde um *Data Warehouse* – DW composto por um conjunto de modelos OLAP, é utilizado como fonte de informações para um sistema de *Customer Relationship Management* - CRM. Este sistema, implantado em meados de julho de 2006, é utilizado como apoio à gestão de clientes, carteiras de clientes e cooperativas no tocante a oferta de produtos e serviços. As informações do DW utilizado são atualizadas semanalmente com base nos sistemas OLTP [INM97] como: conta corrente, crédito, investimentos, seguros, cartões e consórcios. Com relação ao modelo multidimensional utilizado na construção dos modelos OLAP, foram identificadas características dos modelos constelação de fatos e floco de neve. A Tabela 4 apresenta o conjunto de objetos utilizados pelo sistema de CRM da organização, sua classificação quanto ao tipo e o volume de dados de cada objeto coletados em dezembro de 2009.

Tabela 4 – Tabelas de origem

Tabela	Classificação	Volume
FATO_CONTA_CORRENTE	Fato	174.501.244
FATO_CAPTACAO	Fato	155.614.747
FATO_CREDITO	Fato	45.580.321
FATO_CARTAO_CREDITO	Fato	10.632.162
FATO_CARTAO_DEBITO	Fato	40.593.791
FATO_COBRANCA	Fato	1.701.465
FATO_RISCO	Fato	65.166.196
FATO_SEGUROS	Fato	38.525.785
FATO_SERVICO	Fato	190.388.354
CONTA	Dimensão	2.942.491
TEMPO	Dimensão	16.741
ENTIDADE	Dimensão	28.639
PESSOA	Dimensão	2.556.229
UNIDADE_ATENDIMENTO	Auxiliar	1.525
CREDIS	Auxiliar	153
UNIDADE_FEDERATIVA	Auxiliar	27
REGIONAL	Auxiliar	15

Na organização onde esta pesquisa foi desenvolvida, informações gerenciais são extraídas com base em cruzamento das informações geradas pelos relatórios disponibilizados no sistema de CRM ou através da preparação

e execução de consultas *ad-hoc* realizadas diretamente na base de dados por profissionais especializados e com amplo domínio sobre o modelo de dados utilizado. Este cenário motivou a realização desta pesquisa visando à possibilidade de, desenvolver-se um processo semiautomático que potencializasse a obtenção de melhores resultados na utilização de técnicas de mineração de dados, trazendo assim, vantagens estratégicas para a organização.

3.1 Caracterização do Problema

Partindo do pressuposto que as organizações estão constantemente buscando melhorias de seus processos de negócio, visando obter vantagens em um mercado cada vez mais competitivo, projetos de KDD têm sido adotados como diferenciais para organizações de grande porte atuantes nos mais diferentes mercados. Mesmo adotando o processo CRISP-DM [CRI99] que define uma abordagem sobre o ciclo de vida de um projeto de KDD, alguns problemas como: custos elevados, duração do projeto e principalmente a não-garantia de sucesso reduzem bastante o número de organizações que utilizam-se dos benefícios oferecidos pela implantação destes projetos.

Neste contexto, seja através de uma avaliação prévia das informações armazenadas no OLAP da organização ou, através da realização de uma etapa preliminar do projeto que possa organizar os dados e executar uma série de experimentos, visando à identificação de informações que possam sugerir tarefas de mineração de dados, pode-se reduzir os riscos de insucesso e, por consequência, ampliar as chances de uma organização investir em projetos de KDD.

3.2 Caracterização da Contribuição

O objetivo desta pesquisa é propor um processo de exploração semiautomática de modelos OLAP que possibilite a indicação de oportunidades para utilização técnicas de mineração de dados e, desta forma, viabilize a implantação de um projeto de KDD. Este processo, executado em uma base

disponibilizada pela empresa onde esta pesquisa foi realizada, foi desenvolvido em oito etapas. Primeiramente uma amostra (a) das informações da organização foi transferida para uma base de dados apropriada para a pesquisa. Na sequência, as informações desta amostra foram avaliadas (b) e os tratamentos (c) necessários foram realizados com base em informações coletadas de uma *Staging Area* utilizada por outros sistemas de informação da organização. Depois dos ajustes realizados na etapa (c), com o objetivo de consolidar as informações em uma fonte única de pesquisa, foi definido e populado um modelo OLAP (d). Deste modelo, foi extraído um conjunto de consultas (e) formadas por conjuntos de atributos que podem ser utilizados para responder diferentes questões de negócio. Por sua vez, estas consultas foram submetidas a uma ferramenta desenvolvida para executar experimentos (f) assistidos de mineração de dados, cujos resultados foram armazenados (g) para uma posterior avaliação. Por fim, os resultados dos experimentos foram avaliados e ranqueados sendo que, os 25 melhores modelos, foram apresentados e avaliados (h) por especialistas de negócio da organização. A Figura 8 ilustra o conjunto de etapas realizadas no processo proposto nesta pesquisa.

Figura 8 – Etapas do processo

3.2.1 A Amostra de Dados Utilizada

A amostra de dados, selecionada e extraída para a realização desta pesquisa, contemplou informações de 10 cooperativas distribuídas nos estados do RS, SC, PR, MT e SP onde, em cada estado, selecionamos as duas cooperativas com o maior número de clientes. Para armazenar estas informações, foi construído um DW com a mesma estrutura do utilizado pela organização e, para este modelo, foram importadas as informações das tabelas relacionadas na Tabela 4 que compreendem o período de jan/2009 à dez/2009.

3.2.2 Tratamento das Informações

Ao analisar as informações da amostra de dados, observamos algumas inconsistências geradas por falhas no processo de *Extract Transform Load* – ETL [KIM02] da organização. Estas inconsistências, identificadas nas dimensões Pessoa e Conta, destacavam a falta de informações nos atributos data de nascimento, estado civil e tipo de conta para alguns registros destas tabelas. Estas inconsistências foram ajustadas com base em informações coletadas de uma *Staging Area* utilizada pelos sistemas de BI da organização.

3.2.3 Modelo de Dados Preparado para a Pesquisa

Para armazenar as informações utilizadas nos experimentos realizados nesta pesquisa, foi construído um modelo estrela composto por cinco dimensões e uma tabela fato. Este modelo, apresentado em detalhes no Capítulo 4, foi construído e populado no intuito de consolidar as informações da amostra de dados utilizadas nos experimentos realizados nesta pesquisa.

3.2.4 Preparação de Consultas

As consultas preparadas e utilizadas nos experimentos foram construídas com base nas informações armazenadas nos diferentes níveis das dimensões. Procurou-se construir um número de consultas suficientes para responder um variado conjunto de questões de negócio utilizando, em cada consulta, diferentes conjuntos de atributos.

3.2.5 Execução dos Experimentos

A execução dos experimentos foi realizada com o apoio de uma ferramenta construída para automatizar o máximo de operações possíveis. Esta ferramenta, apresentada em detalhes no Capítulo 4, utiliza as consultas construídas com os diferentes conjuntos de dados para coletar as informações armazenadas no DW construído, prepara os arquivos de acordo com o formato utilizado pela ferramenta de mineração de dados e executa, utilizando um conjunto de parâmetros definidos e bibliotecas auxiliares, os experimentos de mineração.

3.2.6 Avaliação dos Resultados

A avaliação dos resultados pode ser realizada utilizando as diferentes informações coletadas dos logs de execução dos experimentos. A ferramenta de apoio, construída para execução dos experimentos, coleta e armazena na

base de dados as diferentes informações apresentadas nos logs após a execução de cada experimento.

3.3 Considerações do Capítulo

Neste capítulo foi apresentado o cenário de desenvolvimento da pesquisa, destacadas questões que podem levar uma organização a não investir em projetos de KDD e, para tratar destas, foi apresentado um processo que discorre sobre a exploração de modelos OLAP e a execução assistida de experimentos de mineração de dados com o objetivo de identificar previamente informações relevantes que possam ser utilizadas para viabilizar a implantação de um projeto de KDD em uma organização. No capítulo seguinte, são apresentados maiores detalhes do desenvolvimento deste trabalho e os resultados obtidos.

4- O PROCESSO DE KDD DESENVOLVIDO

Este capítulo apresenta o processo de KDD desenvolvido detalhando as atividades realizadas no que diz respeito a integração de dados, o modelo OLAP construído, o ferramental desenvolvido utilizado na execução dos experimentos e, por fim, apresenta uma avaliação dos resultados obtidos.

4.1 Processo de integração de Dados

Para a definição do modelo analítico utilizado nos experimentos realizados nesta pesquisa, tomamos como ponto de partida a construção de um modelo estrela capaz de consolidar as informações originais da organização possibilitando uma visão abrangente e centralizada dos diferentes produtos comercializados. Neste contexto, definimos que os objetos fundamentais seriam as dimensões Pessoa, Conta, Tempo, Entidade e Produto associadas a uma tabela Fato. A Figura 9 e a Figura 10 apresentam a estrutura e hierarquias das dimensões bem como o modelo analítico por completo.

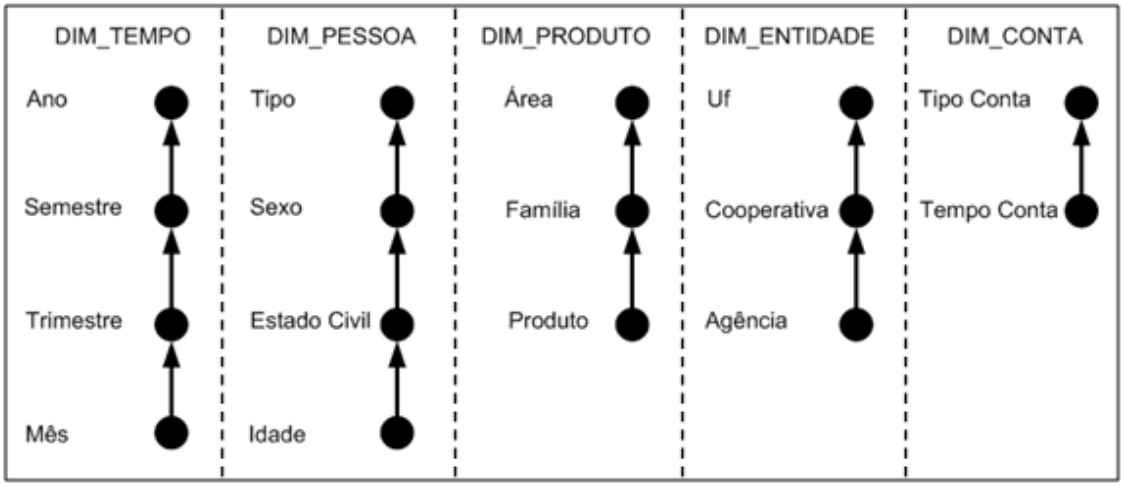


Figura 9 – Hierarquia das dimensões

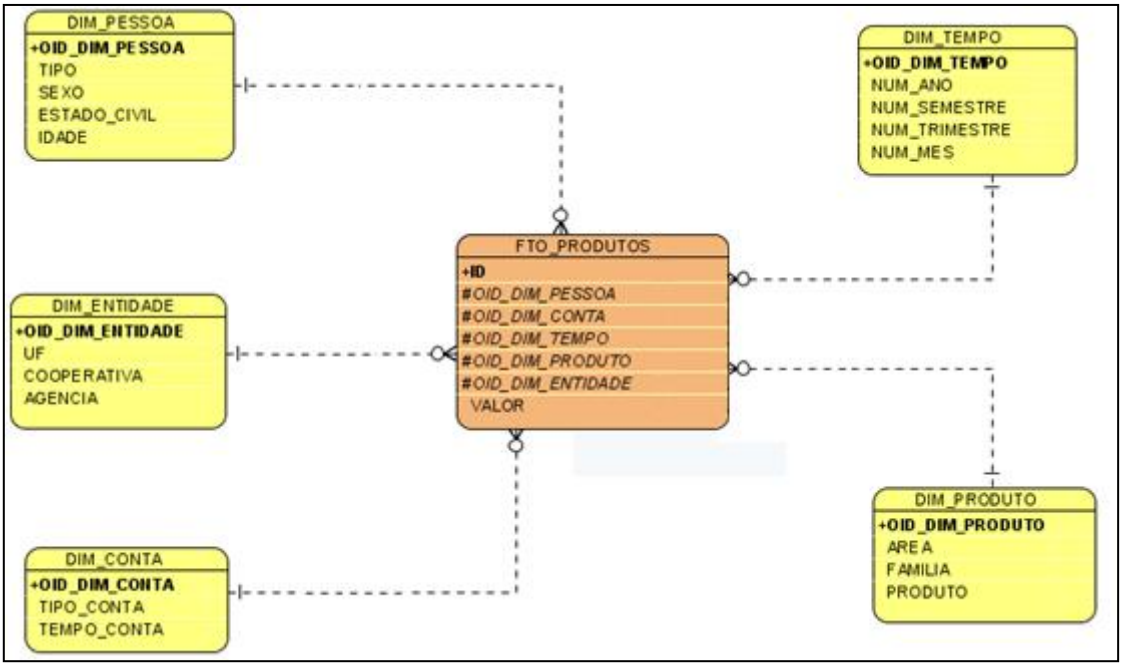


Figura 10 – Modelo Estrela Construído

No intuito de detalhar o modelo construído apresentamos os diferentes objetivos e características pertinentes a cada tabela:

- **DIM_PESSOA**: esta dimensão armazena as informações de Tipo, Sexo, Estado Civil (Solteiro, Casado, Divorciado, Viúvo, União Estável, Não Informado) e, para o atributo Idade, foram criadas as classes A, B, C, D e E que contemplam os seguintes agrupamentos:
  - **A**: até 20 anos;
  - **B**: de 20 a 30 anos;
  - **C**: de 30 a 40 anos;
  - **D**: de 40 a 50 anos;
  - **E**: acima de 50 anos;
- **DIM_ENTIDADE**: esta dimensão armazena as informações de UF, Cooperativa e as Agências que pertencem a cada cooperativa.

- **DIM_CONTA**: esta dimensão de Tipo de Conta (Individual, Conjunta solidária, conjunta não Solidária, Menor) e Tempo de Conta que foi dividido nas classes A, B, C, D, E e F com os seguintes agrupamentos:
    - **A**: até 2 anos;
    - **B**: de 2 a 4 anos;
    - **C**: de 4 a 6 anos;
    - **D**: de 6 a 8 anos;
    - **E**: de 8 a 10 anos;
    - **F**: acima de 10 anos;
- **DIM_TEMPO**: esta dimensão apresenta as informações de Ano, Semestre, Trimestre e Mês.
- **DIM_PRODUTO**: apresenta as informações dos Produtos considerados no modelo, suas Famílias e respectivas Áreas de Negócio.
- **FTO_PRODUTOS**: esta é a tabela Fato do modelo analítico nela, são armazenadas as chaves para cada uma das dimensões e a informação de Valor do produto.

Com relação ao volume de dados destacamos que, mesmo que tenhamos selecionado apenas 10 cooperativas e os registros de 12 meses, o volume da amostra mostrou-se satisfatório para a realização dos experimentos desta pesquisa. A Tabela 5 apresenta o volume de cada tabela do modelo gerado.

Tabela 5 – Volume de dados

Tabela	Volume
DIM_PESSOA	281.589
DIM_TEMPO	12
DIM_ENTIDADE	115
DIM_PRODUTO	23
DIM_CONTA	284.183
FTO_PRODUTOS	6.009.988

4.2 Ferramenta Desenvolvida

A ferramenta desenvolvida para preparação e execução assistida dos experimentos é composta basicamente por três classes Java e um arquivo *properties* onde são definidas algumas configurações fundamentais. Cabe observar que esta ferramenta é auxiliada também por uma biblioteca *open source* utilizada para geração de arquivos CSV [CSV10]. O código fonte da ferramenta pode ser observado em detalhes no Apêndice C.

No arquivo de configurações, cujo conteúdo é apresentado pela Figura 11, são definidas informações vitais para a execução da ferramenta como: o caminho de instalação dos softwares Weka e Java, a configuração de memória que pode ser alocada para a execução da ferramenta, o algoritmo de mineração que será utilizado, a pasta onde são disponibilizadas as consultas que serão executadas pela ferramenta, a identificação do experimento, o atributo classe e um parâmetro de balanceamento.

```
#Configurações
weka.home=C:\\softwares\\Weka-3-6\\weka.jar
url =jdbc:oracle:thin:@peterson-pc:1521:dblocal
Xmx = -D64 -Xmx2000m
java.path =C:\\java\\64\\jdk1.6.0_21\\bin\\java
weka.algoritmo = weka.classifiers.trees.J48 -C 0.25 -M 2 -t

#geral
diretorio.origem =C:\\_experimentos\\1
experimento.nome = EXPERIMENTO_16

#Parametrizações do Balancer
#definicao do atributo classe do arquivo original
atributo.classe=AREA

#definicao do conjunto de classes desejadas no arquivo de saída
#importante: a) quando nenhuma classe for informada, o algoritmo deverá balancear todas
#               as classes encontradas no arquivo original
#            b) quando for informada somente uma classe, deve-se gerar "classe" e "não_classe"
#               como o conjunto de classes no arquivo de saída
#classes.alvo=CRED,INVEST
#classes.alvo=CART,CONS,SEG,CRED,INVEST
classes.alvo=INVEST
```

Figura 11 – Arquivo de configuração – config.properties

A classe *PrepareFile* pode ser considerada a principal classe da ferramenta. Dentre suas funcionalidades destacamos a leitura dos arquivos SQL, a conexão com o banco de dados, a execução das consultas e a exportação para CSV do resultado de cada consulta executada. Como última atividade, esta classe prepara um script que deve ser executado para conversão dos arquivos CSV em arquivos ARFF. A Figura 12 apresenta um diagrama de atividades UML [ARL09] que detalha o funcionamento da classe.

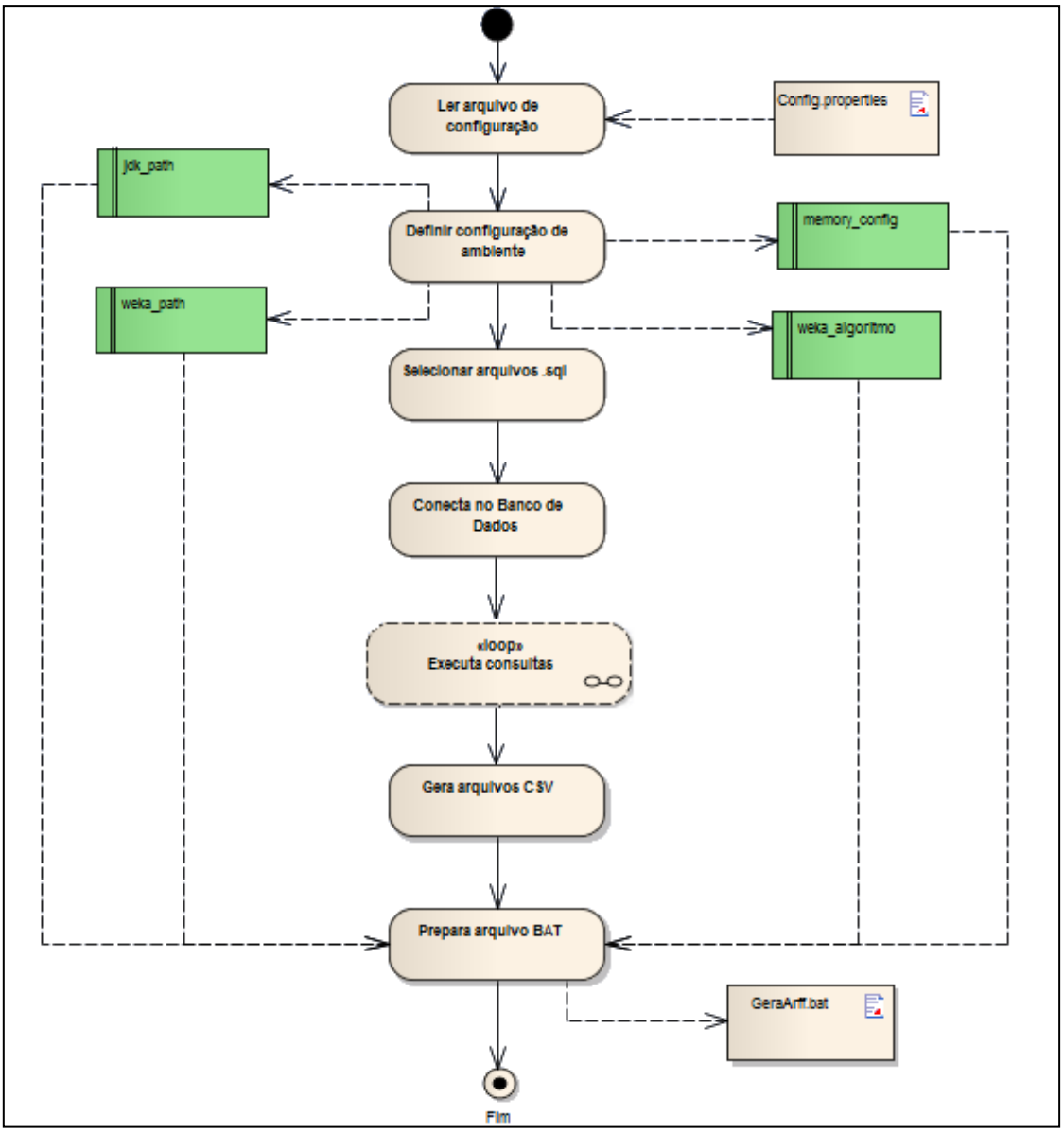


Figura 12 – Diagrama de Atividades – *PrepareFile.class*

As outras duas classes utilizadas pela ferramenta desempenham um papel menos complexo, mas de igual importância, para execução dos experimentos. A classe *BalanceFilter* realiza o balanceamento de classes em cada arquivo ARFF gerado obedecendo o parâmetro "*classe.alvo"* definido no arquivo de configuração. Após a realização do balanceamento, que se resume em manter no arquivo ARFF o mesmo número de instâncias de cada classe definida no parâmetro, esta classe prepara um script para execução dos experimentos utilizando as bibliotecas disponibilizadas com o Weka. Uma pequena amostra deste script pode ser observada na Figura 13.

```
C:\java\64\jdk1.6.0 21\bin\java -D64 -Xmx2000m -cp C:\softwares\Weka-3-6\weka.jar
weka.classifiers.trees.J48 -C 0.25 -M 2 -t consulta_1.arff -d  consulta_1.model >
consulta_1.log

C:\java\64\jdk1.6.0 21\bin\java -D64 -Xmx2000m -cp C:\softwares\Weka-3-6\weka.jar
weka.classifiers.trees.J48 -C 0.25 -M 2 -t consulta_10.arff -d  consulta_10.model >
 consulta_10.log
```

Figura 13 – Script para execução dos experimentos

A classe *GetLog* é utilizada após a realização dos experimentos. Esta classe realiza a leitura do log de execução de cada experimento coletando as informações necessárias para a avaliação dos resultados. O armazenamento destas informações é realizado no banco de dados em uma tabela identificada como experimentos_log. A Figura 14 apresenta um trecho do log de execução de um experimento com um algoritmo de classificação onde são destacadas as informações armazenadas na tabela experimentos_log.

```
Number of Leaves  :   776
Size of the tree :  975

Time taken to build model: 16.94 seconds
Time taken to test model on training data: 3.9 seconds

.........

=== Stratified cross-validation ===

Correctly Classified Instances      584989               59.5445 %
Incorrectly Classified Instances    397451               40.4555 %
Kappa statistic                          0.1909
Mean absolute error                      0.472
Root mean squared error                  0.486
Relative absolute error                 94.395  %
Root relative squared error             97.2026 %
Total Number of Instances           982440


=== Confusion Matrix ===

      a      b   <-- classified as
 259639 231581 |      a = INVEST
 165870 325350 |      b = NAO_INVEST
```

Figura 14 – Log de execução de um classificador

4.3 Execução dos Experimentos

Com o objetivo de atender os propósitos desta pesquisa e ainda, em função das características das informações disponíveis no OLAP construído, optamos por limitar o escopo a utilização do algoritmo classificador J48 disponibilizado pelo Weka. Isso posto, a execução dos experimentos é realizada em quatro etapas. Inicialmente, deve-se preparar as consultas, que possam ser utilizadas para responder determinadas questões de negócio, utilizando diferentes conjuntos de atributos do modelo de dados. Nesta etapa, pode-se relacionar informações das diferentes dimensões em seus diferentes níveis de hierarquia. O conjunto de consultas gerado será utilizado pela etapa seguinte do processo. A Figura 15 apresenta o exemplo de uma consulta construída utilizando atributos de diferentes dimensões. A Tabela 6 apresenta o conjunto de consultas preparadas para os experimentos realizados onde podemos observar os atributos utilizados.

```
SELECT PE.IDADE AS IDADE
     , CTA.FLG_TIPO_CONTA AS TIPO_CONTA
     , CTA.CLASSE_TEMPO_CONTA AS TEMPO_CONTA
     , PE.DES_SEXO AS SEXO
     , PE.FLG_ESTADO_CIVIL AS ESTADO_CIVIL
, ENTI.UF AS UF
     , PROD.DES_AREA  AS AREA
FROM CRM_DIM_PESSOA PE
   , CRM_DIM_PRODUTO PROD
   , CRM_FTO_PRODUTOS FTO
   , CRM_DIM_CONTA CTA
, CRM_DIM_ENTIDADE ENTI
WHERE PE.OID_PESSOA = FTO.OID_DIM_PESSOA
AND PROD.OID_PRODUTO = FTO.OID_DIM_PRODUTO
AND FTO.OID_DIM_CONTA = CTA.OID_CONTA
AND FTO.OID_DIM_ENTIDADE = ENTI.OID_DIM_ENTIDADE
AND PE.DES_TIPOPESSOA = 'F'
AND FTO.OID_DIM_TEMPO IN (200903, 200906, 200909, 200912)
```

Figura 15 – Exemplo de uma consulta preparada

Tabela 6 – Consultas preparadas

Consulta	Pessoa				Conta		Entidade	Produto	Tempo
	Tipo	Sexo	Estado Civil	Idade	Tipo Conta	Tempo Conta	Uf	Área	Trimestre
Arq_1.sql	x			x			x	x	x
Arq_2.sql		x	x	x			x	x	x
Arq_3.sql			x	x			x	x	x
Arq_4.sql		x	x				x	x	x
Arq_5.sql	x			x	x		x	x	x
Arq_6.sql		x	x	x	x		x	x	x
Arq_7.sql			x	x	x		x	x	x
Arq_8.sql		x	x		x		x	x	x
Arq_9.sql	x			x	x	x	x	x	x
Arq_10.sql		x	x	x	x	x	x	x	x
Arq_11.sql			x	x	x	x	x	x	x
Arq_12.sql		x	x		x	x	x	x	x
Arq_13.sql	x			x		x	x	x	x
Arq_14.sql		x	x	x		x	x	x	x
Arq_15.sql			x	x		x	x	x	x
Arq_16.sql		x	x			x	x	x	x
Arq_17.sql	x			x				x	x
Arq_18.sql		x	x	x				x	x
Arq_19.sql			x	x				x	x
Arq_20.sql		x	x					x	x
Arq_21.sql	x			x	x			x	x
Arq_22.sql		x	x	x	x			x	x
Arq_23.sql			x	x	x			x	x
Arq_24.sql		x	x		x			x	x
Arq_25.sql	x			x	x	x		x	x
Arq_26.sql		x	x	x	x	x		x	x
Arq_27.sql			x	x	x	x		x	x
Arq_28.sql		x	x		x	x		x	x
Arq_29.sql	x			x		x		x	x
Arq_30.sql		x	x	x		x		x	x
Arq_31.sql			x	x		x		x	x
Arq_32.sql		x	x			x		x	x

Após a construção das consultas, os arquivos sql precisam ser disponibilizados em uma pasta, indicada juntamente com os demais

parâmetros do arquivo *Config.properties* para que as classes, detalhadas no item 4.2, possam ser executadas. Ao final do processo de execução, as informações coletadas diretamente dos *logs* de execução dos experimentos poderão ser observadas e avaliadas diretamente na base de dados onde os resultados são armazenados.

Cabe observar que para cada consulta relacionada na Tabela 6, foram preparados 32 arquivos ARFF contendo diferentes conjuntos de classes alvo. A Tabela 7 apresenta os conjuntos de classes utilizados.

Tabela 7 – Conjuntos de classes utilizados

Conjunto	**Classes Utilizadas**
Conjunto_1	CART, CONS, SEG, CRED e INVEST
Conjunto_2	CART e CONS
Conjunto_3	CART e SEG
Conjunto_4	CART e CRED
Conjunto_5	CART e INVEST
Conjunto_6	CONS e SEG
Conjunto_7	CONS e CRED
Conjunto_8	CONS e INVEST
Conjunto_9	SEG e CRED
Conjunto_10	SEG e INVEST
Conjunto_11	CRED e INVEST
Conjunto_12	CART e NAO_CART
Conjunto_13	CONS e NAO_CONS
Conjunto_14	SEG e NAO_SEG
Conjunto_15	CRED e NAO_CRED
Conjunto_16	INVEST e NAO_INVEST

4.4 Avaliação dos Modelos

Considerando que foram utilizados 16 conjuntos de classes e, para cada conjunto, foram preparados 32 arquivos ARFF resultantes das consultas apresentadas na Tabela 6, foram realizados 512 experimentos distintos. Os resultados destes experimentos foram armazenados na tabela de logs cuja estrutura é apresentada pela Figura 16.

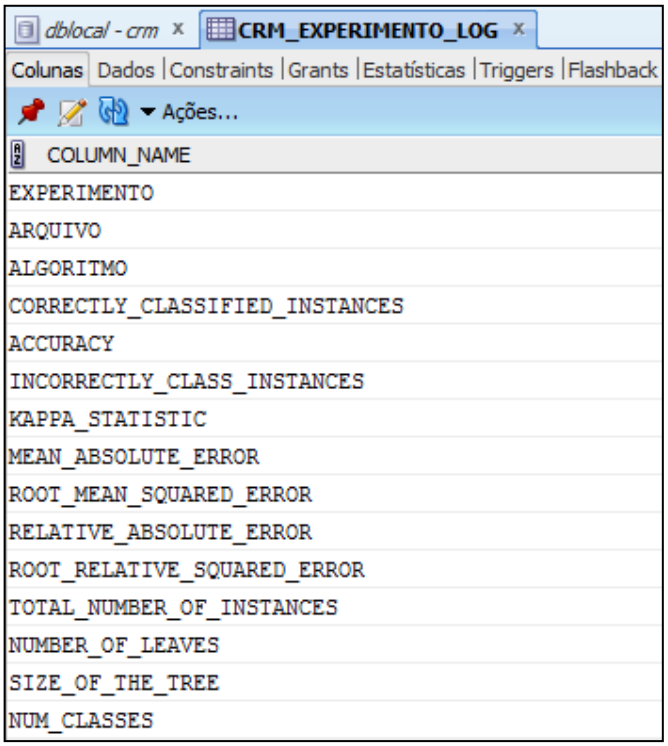


Figura 16 – Tabela de logs

Para possibilitar a avaliação por parte dos especialistas de negócio da organização, selecionamos os 25 experimentos utilizando dois critérios de seleção. O primeiro critério utilizado diz respeito ao número de nodos da árvore de decisão gerada. Neste, selecionamos experimentos em que o número de nodos estivesse no intervalo entre 10 e 40. Estes limites foram estabelecidos após termos verificado na tabela de resultados experimentos com árvores extremamente pequenas, que não possuem um significado relevante, e árvores com centenas de níveis que dificultariam bastante a avaliação dos modelos gerados. O segundo critério utilizado foi a acurácia, selecionamos dentre os experimentos classificados pelo primeiro critério, aqueles que apresentaram a maior acurácia. Um resumo dos logs de execução dos experimentos selecionados para avaliação é apresentado na Tabela 8.

Tabela 8 – Resultados dos experimentos classificados

Ranque	Experimento	Consulta	Acurácia	Nº Instâncias	Classes		Confiabilidade	
					Classe A	Classe B	Classe A	Classe B
1º	8	15	83,71%	24002	CONS	INVEST	74,85%	92,58%
2º	8	9	83,10%	27790	CONS	INVEST	73,12%	93,08%
3º	8	13	83,02%	27790	CONS	INVEST	72,99%	93,04%
4º	7	11	82,34%	24002	CONS	CRED	73,34%	91,33%
5º	4	13	77,59%	65816	CART	CRED	63,17%	92,01%
6º	4	9	77,57%	65816	CART	CRED	63,98%	92,06%
7º	5	1	76,78%	65816	CART	INVEST	61,12%	92,44%
8º	5	5	76,73%	65816	CART	INVEST	62,07%	91,39%
9º	4	5	76,38%	65816	CART	CRED	64,05%	88,70%
10º	5	3	75,88%	70310	CART	INVEST	60,14%	91,62%
11º	4	1	75,77%	65816	CART	CRED	67,69%	83,86%
12º	4	7	75,66%	70310	CART	CRED	52,35%	98,97%
13º	4	6	75,66%	70310	CART	CRED	52,36%	98,95%
14º	4	2	75,65%	70310	CART	CRED	52,48%	98,83%
15º	4	3	75,65%	70310	CART	CRED	52,49%	98,81%
16º	5	8	75,29%	70310	CART	INVEST	56,34%	94,23%
17º	5	4	74,91%	70310	CART	INVEST	51,75%	98,07%
18º	6	9	74,67%	27790	CONS	SEG	53,16%	96,17%
19º	6	13	74,53%	27790	CONS	SEG	57,09%	91,98%
20º	6	5	73,33%	27790	CONS	SEG	53,16%	93,49%
21º	6	1	72,64%	27790	CONS	SEG	57,93%	87,35%
22º	6	15	71,21%	24002	CONS	SEG	50,93%	91,50%
23º	13	5	70,21%	27790	CONS	NAO_CONS	57,55%	82,88%
24º	6	16	70,19%	24002	CONS	SEG	42,67%	97,70%
25º	13	1	69,97%	27790	CONS	NAO_CONS	59,62%	80,33%

Dentre os experimentos selecionados podemos observar que um mesmo conjunto de classes foi utilizado em diferentes experimentos e estes utilizaram diferentes conjuntos de atributos. Utilizando a abordagem selecionada para ranquear os resultados, podemos observar que nos primeiros experimentos obtivemos uma acurácia acima de 80%. Esta acurácia pode ser considerada plenamente aceitável considerando que, por não ser o propósito específico desta pesquisa, os experimentos foram realizados utilizando as configurações padrão do algoritmo. Outra observação relevante diz respeito ao conjunto de atributos utilizados em cada um dos experimentos ranqueados. A definição dos melhores conjuntos de atributos para cada experimento foi

simplificada pelo processo de execução assistida de experimentos desenvolvido nesta pesquisa.

Uma métrica que também pode ser utilizada para avaliar o resultado de um experimento é a confiabilidade de uma determinada classe. Esta métrica leva em consideração o percentual de acertos realizados pelo algoritmo para cada classe. Neste contexto, destacamos a ocorrência de 6 experimentos onde podemos observar uma confiabilidade acima de 96% para uma das classes utilizadas.

Ao analisar as árvores de decisão geradas pelos experimentos observamos que, na grande maioria, foram apresentados resultados em um nível de detalhe adequados para interpretação e análise dos modelos. Porém, em função de termos utilizado as configurações padrão do algoritmo classificador, processos de poda foram realizados eliminando em praticamente todos os experimentos as ramificações das árvores que apresentariam informações de registros dos estados de MT, SC e SP. Observamos que as cooperativas destes estados apresentam um número muito inferior de associados quando comparadas com as selecionadas nos estados do RS e PR. Esta diferença significante quanto ao número de registros é refletida nos arquivos preparados e utilizados em todos os experimentos realizados.

4.5 Avaliação dos Especialistas de Negócio

Para possibilitar a avaliação dos experimentos selecionados por parte dos especialistas de negócio da organização foi realizada uma reunião com cinco analistas de negócios onde foram apresentadas as características da pesquisa realizada, seus objetivos, detalhamento da amostra de dados e experimentos realizados. No intuito de simplificar o processo de avaliação foram disponibilizados arquivos com a representação gráfica de cada árvore de decisão, coletadas do software weka, que facilitam a interpretação das árvores detalhadas nos logs dos experimentos realizados apresentados no Apêndice A.

Após a apresentação da pesquisa e do material para avaliação, foi

entregue aos especialistas um questionário, apresentado em detalhes no Apêndice B. Neste, os especialistas poderiam sugerir, após terem avaliado o material entregue, um ranqueamento dos experimentos realizados de acordo com suas percepções sobre o problema tratado, definir um grau de relevância do processo apresentado nesta pesquisa no contexto da organização e, para complementar suas observações, foi disponibilizado um espaço para que pudessem descrever seus comentários gerais. Observamos que o perfil técnico, a formação e a experiência na área de atuação foram fatores relevantes para a seleção dos especialistas que participaram da avaliação sendo que o grupo selecionado é composto de: dois estatísticos, dois analistas de *business intelligence* e um analista de planejamento comercial, ambos desempenham suas atividade na área de inteligência de negócios da organização.

Um resumo das avaliações realizadas é apresentado na sequência sendo que todos os questionários respondidos são apresentados em detalhes no Apêndice B.

- **Primeira questão**: Analisando as árvores geradas, como você ranquearia os experimentos realizados?

  Todos os especialistas indicaram que a acurácia, utilizada no ranqueamento realizado pelo processo apresentado nesta pesquisa, é adequado para ranquear os experimentos realizados sem que, para isso, fosse necessário avaliar cada árvore de decisão.

- **Segunda questão**: Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha? Indique um valor de 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

A Figura 17 apresenta o gráfico gerado com as respostas dos especialistas para esta questão.

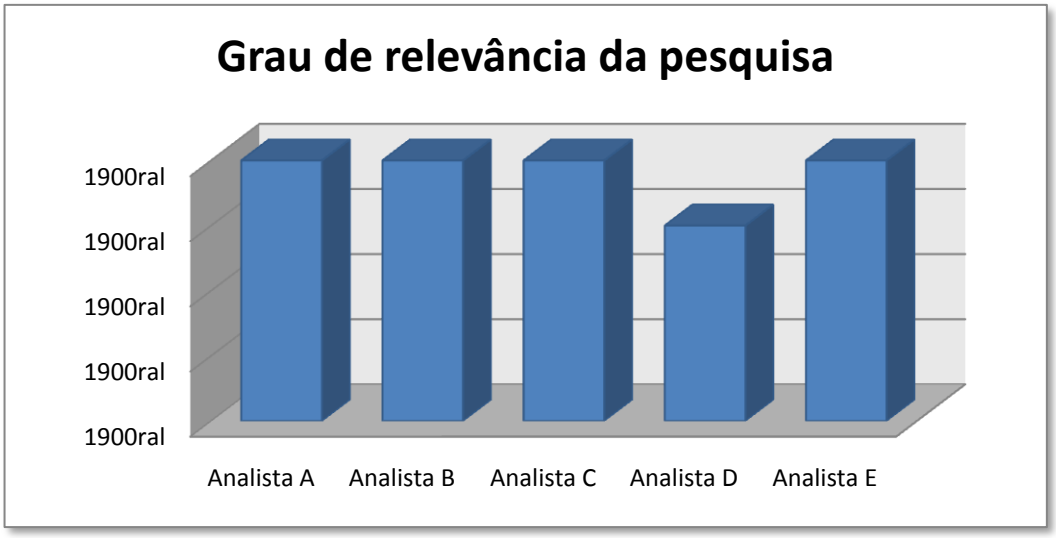


Figura 17 – Grau de relevância da pesquisa

- **Terceira questão**: Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

  As percepções indicadas pelos especialistas foram as seguintes:

  a. Analista A: “A atual estrutura de negócios da organização está vivendo um momento ímpar com relação ao seu potencial futuro. A vontade de crescer é grande, e os recursos para investimentos também existem. Porém, algo que ainda o sistema carece é de uma informação acurada, que funcione como “bússola”; driver de negócios. Enxergo no trabalho do colega Peterson Colares um início da construção desta “bússola”. Um “pontapé inicial” que muito poderá nos ajudar a navegar de forma mais certeira neste “oceano do mercado financeiro”. Os resultados por ele apresentados são extremamente significativos, e terão grande relevância para organização, se implantados a nível sistêmico”;

b. Analista B: “O trabalho vem resolver de uma forma bem estruturada e simplificada o processo de modelagem através do uso de ferramentas de mineração de dados. De uma forma bem interessante foi proposto uma escolha do melhor modelo através da acurácia. Trata-se de uma proposta bem interessante e que vale a pena validá-la o quanto antes. Inicialmente deve-se propor um corte e testar os grupos formados. Recomendo a ‘leitura’ dos grupos, a fim de verificar se faz sentido o agrupamento de variáveis que se ‘fundiram’. Também, a partir desta proposta de modelagem será interessante aplicar a metodologia em outras áreas, como por exemplo estudos em geomarketing. A escolha da melhor localização para a abertura de pontos comerciais é o grande desafio das redes de varejo”;

c. Analista C: “Acredito que a mineração de dados é de extrema relevância, pois transforma dados brutos em inteligência mercadológica. A automatização do processo de mineração de dados potencializa a viabilidade de estudos via data mining em tempo real, ou muito próximo disso, o que sob a ótica do usuário final dessa ferramenta é um diferencial competitivo extremamente valioso. Outro ponto relevante é a robustez de um data mining automatizado, que permite tanto consultas para os mais variados temas quanto uma rápida e eficiente imputação de novos dados, assim como um repositório único de informações, garantindo com isso a integridade dessas informações, uma vez que diferentes fontes de dados geram um alto risco de incongruências nos dados extraídos. Por fim vale

citar que a aplicabilidade deste trabalho vai desde a eficaz prospecção de novos clientes ou vendas de novos produtos, até a análise de cenários geográficos com alto potencial para a expansão de novos negócios, ou até mesmo um relacionamento mais preciso e valoroso para com os atuais clientes";

d. Analista D: "O trabalho é extremamente útil para identificação de um possível ramo de resposta inicial, com isso indicando um possível norte para o trabalho. Além de já indicar algumas respostas mais claras. Eduardo Berno – Estatístico – Analista de BI";

e. Analista E: "Na questão (a), relativo ao ranqueamento, fica o entendimento de que a acurácia é um excelente critério a ser utilizado. Mesmo não avaliando a árvore de decisão. Atualmente, o processo de Data Mining além de preparar e integrar dados estruturados, pode também:

   • Construir e validar modelos, utilizando-se das mais avançadas técnicas de estatística;

   • Disponibilizar eficientemente o conhecimento e aplicar os modelos preditivos, para os tomadores de decisão de sua empresa e os sistemas que os apoiam.

   Este processo é visto com custo as organizações, que através de uma lente míope, esperam que tal modelo complexo gere resultados no curtíssimo prazo. Apoiando-se sobre esta visão curta, as organizações não compreendem o quão ótimo é o

> retorno sobre o investimento em mineração de dados, mas, no longo prazo. Apesar de, globalizado o mercado dificilmente dará retorno de imediato. Sendo assim, o modelo proposto neste trabalho evidencia sua total aplicabilidade no mercado atual. Desta forma, as organizações poderão antecipar as necessidades dos seus mercados consumidores".

4.6 Considerações do Capítulo

Este capítulo apresentou o processo de KDD desenvolvido nesta pesquisa detalhando desde a preparação do modelo, o processo de integração e tratamento dos dados até o ferramental desenvolvido para automatização dos experimentos. Foi apresentado também o processo de definição, preparação e execução dos experimentos, o processo de armazenamento de resultados, a avaliação dos resultados obtidos realizada pelo autor, bem como um resumo das avaliações realizadas por especialistas de negócio da organização.

No que diz respeito às avaliações realizadas pelos especialistas de negócio podemos afirmar que o objetivo foi parcialmente atingido. Nenhum dos especialistas avaliou o conjunto de árvores geradas limitando seus pareceres a uma avaliação dos benefícios que o processo apresentado pode trazer para a organização.

5- TRABALHOS RELACIONADOS

Este capítulo apresenta alguns trabalhos que estão relacionados com o tema desta pesquisa. Estes trabalhos são relatados a seguir e, após a descrição destes, discorremos sobre a comparação com o processo desenvolvido nesta pesquisa.

A identificação de trabalhos relacionados com o tema pesquisado foi uma das maiores dificuldades enfrentadas no desenvolvimento desta pesquisa. Foram realizadas inúmeras buscas utilizando diferentes *strings* relacionadas com o tema e, dentre os trabalhos que de alguma forma apresentam semelhanças com o processo apresentado nesta pesquisa, destacamos o CRISP-DM [CRI99] e o processo de KDD apresentado por [FAY96] e [HAN01].

O CRISP-DM apresenta um processo sobre o ciclo de vida de um projeto de mineração de dados cujo objetivo é definir e controlar as diferentes fases do projeto visando à qualidade do mesmo. A abordagem utilizada divide o projeto de mineração de dados em seis fases distintas que são utilizadas como um guia para a execução e controle das etapas do projeto. O CRISP-DM diferenciasse do processo apresentado nesta pesquisa na medida em que este, mesmo utilizando determinadas etapas do processo de KDD busca a identificação de cenários promissores que motivem investimentos nestes projetos. Entretanto, destacamos que o processo apresentado pelo CRISP-DM pode ser utilizado com complementar ao proposto nesta pesquisa. A Figura 18 apresenta as fases do processo e a relação entre estas.

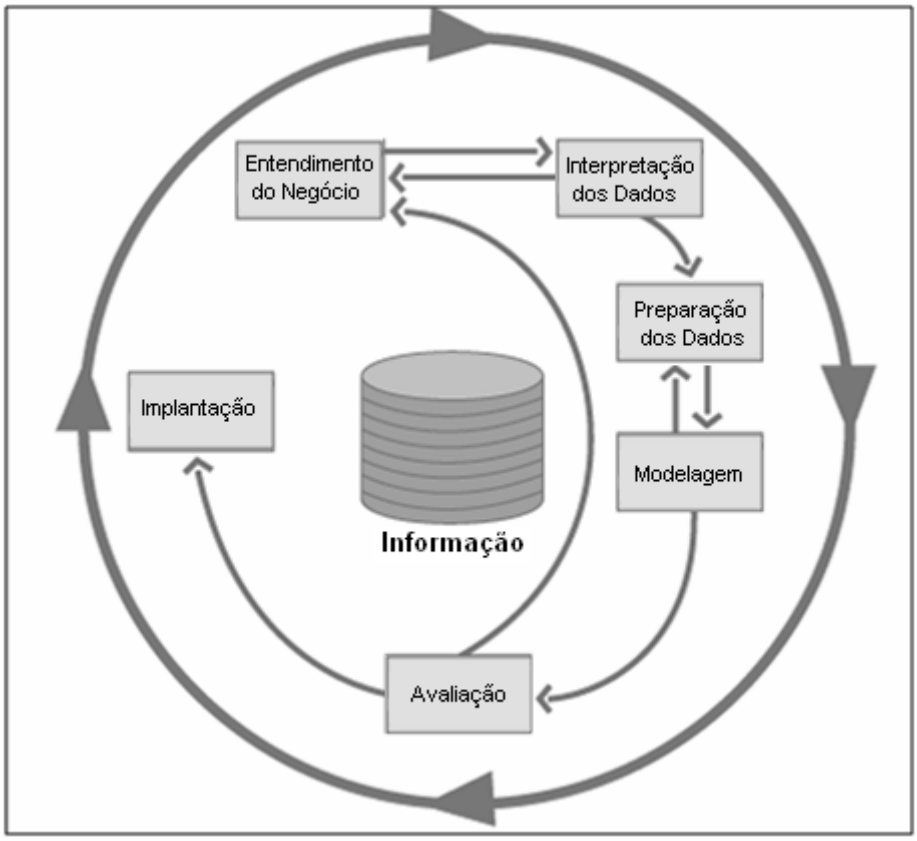


Figura 18 – Fases do CRISP-DM
Fonte: Adaptado de [CRI99]

No que diz respeito ao processo de KDD apresentado por [FAY96] e adaptado por [HAN01] com a inclusão de uma etapa para armazenamento de dados em um *Data Warehouse,* apresentado em detalhes no Capítulo 2, observamos que o processo proposto por esta pesquisa está diretamente relacionado em função de que utiliza as mesmas etapas do processo de KDD. Porém, em função de características apresentadas no cenário onde esta pesquisa foi realizada, foi necessário realizar uma etapa de tratamento e integração de informações coletadas de um DW formado por um conjunto de modelos OLAP previamente modelados e utilizados pela organização.

Destacamos ainda que, em função da dificuldade em identificar trabalhos semelhantes ao processo proposto nesta pesquisa, realizamos um contato com o Prof. Dr. Jiawei Han, autor de inúmeras publicações desta área bem como de livros amplamente utilizados em universidades que abordam mineração de dados em disciplinas de cursos de graduação e pós-graduação. Este, por sua vez, retornou nosso contato indicando dois grupos de pesquisa

onde poderíamos, pesquisando em seus trabalhos publicados, encontrar alguma abordagem que se aproximasse ao tema que estávamos desenvolvendo. Foi realizada então uma busca nos artigos publicados pelos grupos sugeridos onde novamente não encontramos materiais especificamente relacionados. A grande maioria dos trabalhos publicados pelos grupos indicados abordava processos de armazenamento ou coleta de informações de modelos OLAP focados no atendimento de requisitos de determinados algoritmos de mineração.

6- CONSIDERAÇÕES FINAIS E TRABALHOS FUTUROS

Este trabalho apresenta um processo de indução e ranqueamento de árvores de decisão baseado em informações extraídas de modelos OLAP com o objetivo de identificar, de forma semiautomatizada, modelos que apresentem oportunidades para aplicação de técnicas de mineração de dados. Um dos principais objetivos do processo desenvolvido é fazer com que mais organizações venham a investir em projetos de mineração de dados no momento em que estas, utilizando o processo apresentado, possam visualizar indícios de retorno sobre os investimentos realizados nestes projetos.

O cenário onde esta pesquisa foi desenvolvida apresentou características favoráveis para a execução de um processo de KDD completo. Questões que envolvem tipo e qualidade dos dados, limpeza, integração, seleção, transformação e armazenamento, bem como a execução de uma série de experimentos, executados com o apoio de um ferramental desenvolvido, foram exploradas e detalhadas nos Capítulos 2, 3 e 4 desta pesquisa. O processo desenvolvido foi executado com o apoio de um conjunto de programas de código aberto, possibilitando que estes sejam melhorados, otimizados e adaptados para diferentes necessidades das organizações. Para disponibilizar informações utilizadas nos experimentos realizados foi preparada uma base de dados e construído um modelo estrela que foi populado com informações coletadas de diferentes modelos OLAP utilizados pela organização. O volume de informações coletadas possibilitou a realização de um considerável número de experimentos. Este fato destacou a capacidade e flexibilidade do processo desenvolvido em trabalhar com grandes volumes de dados, gerar e armazenar um diferente conjunto de resultados e, principalmente, a simplificação do processo de avaliação dos resultados na medida em que estes são coletados e armazenados em uma tabela do banco de dados.

Por se tratar de um tema consideravelmente abrangente, identificamos que algumas melhorias podem ser realizadas no processo apresentado, dentre estas destacamos:

- A adaptação do processo para coleta e armazenamento de experimentos realizados com outras técnicas de mineração de dados;
- A automatização da construção de consultas geradas com diferentes atributos do modelo de dados;
- A otimização do código para que se possam realizar experimentos de classificação com um volume maior de informações em cada experimento;
- A utilização deste processo em diferentes organizações e segmentos de negócios;

Por fim, considerando as avaliações realizadas pelos especialistas de negócio, detalhadas no Capítulo 4, o trabalho desenvolvido possibilita uma maior segurança para as organizações investirem em projetos de mineração de dados na medida em que se consegue, através da identificação de modelos professores, minimizar os riscos de insucesso em projetos realizados nesta área.

REFERÊNCIAS

[ARL09] ARLOW, J.; NEUSTADT, I. “UML 2 and the unified process: practical object-oriented analysis and design”. Upper Saddle River: Addison-Wesley, 2009, 592p.

[CRI99] CROSS Industry Standard Process for data mining. Capturado em: http://www.crisp-dm.org, Junho 2010.

[CSV10] A Simple CSV Parser for Java. Capturado em: http://sourceforge.net/projects/opencsv/, Agosto 2010.

[FAY96] FAYYAD, U.; SHAPIRO, G. P.; SMYTH, P. “Advances in knowledge discovery and Data Mining”. Menlo Park: AAAI PRESS, 1996, 611p.

[HAN01] HAN, J.; KAMBER, M. "Data Mining: concepts and techniques". San Francisco: Morgan Kaufmann Publishers, 2001, 550p.

[INM97] INMON, W. H. "Como construir o Data Warehouse ". Rio de Janeiro: Campus, 1997, 388p.

[KIM98] KIMBALL, R. “The Data Warehouse Lifecycle Toolkit: Expert Methods for Designing, Developing, and Deploying Data Warehouses”. New York: John Wiley & Sons, 1998, 771p.

[KIM02] KIMBALL, R.; ROSS, M. “The Data Warehouse toolkit : the complete guide to dimensional modeling”. New York: John Wiley & Sons, 2002, 421p.

[LIR07] LI, T.; RUAN, D. “An extended process model of knowledge discovery in databases”. *Journal of Enterprise Information Management*, vol. 20, Fev 2007, pp.169-177.

[ODM10] Oracle Data Mining. Capturado em: http://www.oracle.com/us/products/database/options/data-mining/index.htm/, Outubro de 2010.

[QUI96] QUINLAN, J. R. “C4.5: programs for machine learning”. San Francisco: Morgan Kaufmann Publishers, 1996, 302 p.

[RDM10] RapidMiner. Capturado em: http://rapid-i.com/content/view/181/190/, Outubro 2010.

[SAS10] SAS Enterprise Miner. Capturado em: http://www.sas.com/technologies/analytics/datamining/miner/, Outubro 2010.

[SPS10] IBM SPSS Modeler. Capturado em: http://www.spss.com/software/modeler/, Outubro 2010.

[SWI01] SWIFT, Ronald. “CRM, customer relationship management”. Rio de Janeiro: Elsevier, 2001, 493p.

[TAN06] TAN, P. N.; STEINBACH, M.; KUMAR, V. “Introduction to Data Mining“. Boston: Addison-Wesley, 2006, 796p.

[TUR09] TURBAN, E.; SHARDA, R.; ARONSON, J. E.; KING, D. “Business Intelligence: um enfoque gerencial para a inteligência de negócio“. Porto Alegre: Bookman, 2009, 253p.

[WEK10] WEKA: Waikato environment for knowledge analysis. Capturado em: http://www.cs.waikato.ac.nz/ml/weka/, Outubro 2010.

[WIT05] WITTEN, I.; FRANK, E. “Data mining: pratical machine learning tools and techniques”. San Francisco: Morgan Kaufmann, 2005, 525 p.

APÊNDICE A

Log do 1º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------
UF = RS: INVEST (17058.0/4024.0)
UF = PR: CONS (5807.0/861.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :          5
Size of the tree :      6

Time taken to build model: 0.16 seconds
Time taken to test model on training data: 0.2 seconds
=== Error on training data ===
Correctly Classified Instances       22905               82.4217 %
Incorrectly Classified Instances      4885               17.5783 %
Kappa statistic                          0.6484
Mean absolute error                      0.2741
Root mean squared error                  0.3702
Relative absolute error                 54.8122 %
Root relative squared error             74.0352 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 9871  4024 |     a = CONS
  861 13034 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       22905               82.4217 %
Incorrectly Classified Instances      4885               17.5783 %
Kappa statistic                          0.6484
Mean absolute error                      0.2741
Root mean squared error                  0.3702
Relative absolute error                 54.8135 %
Root relative squared error             74.0403 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 9871  4024 |     a = CONS
  861 13034 |     b = INVEST
```

Log do 2º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------
UF = RS
|   IDADE = D: INVEST (3270.0/804.0)
|   IDADE = A: INVEST (1493.0/283.0)
|   IDADE = C
|   |   TEMPO_CONTA = F
|   |   |   TIPO = F: CONS (522.0/192.0)
|   |   |   TIPO = J: INVEST (157.0/60.0)
|   |   TEMPO_CONTA = A: INVEST (653.0/104.0)
|   |   TEMPO_CONTA = C: INVEST (645.0/181.0)
|   |   TEMPO_CONTA = B: INVEST (704.0/193.0)
|   |   TEMPO_CONTA = E: INVEST (788.0/251.0)
|   |   TEMPO_CONTA = D: INVEST (585.0/142.0)
|   IDADE = E: INVEST (4751.0/601.0)
|   IDADE = B: INVEST (3490.0/1075.0)
UF = PR
|   TEMPO_CONTA = F: CONS (2334.0/101.0)
|   TEMPO_CONTA = A
|   |   IDADE = D: CONS (165.0/58.0)
|   |   IDADE = A: CONS (133.0/47.0)
|   |   IDADE = C: CONS (190.0/68.0)
```

```
|   |   IDADE = E
|   |   |   TIPO_CONTA = C: CONS (39.0/16.0)
|   |   |   TIPO_CONTA = I: INVEST (122.0/34.0)
|   |   |   TIPO_CONTA = M: INVEST (0.0)
|   |   |   TIPO_CONTA = S: INVEST (0.0)
|   |   IDADE = B: CONS (286.0/83.0)
|   TEMPO_CONTA = C: CONS (771.0/90.0)
|   TEMPO_CONTA = B: CONS (899.0/152.0)
|   TEMPO_CONTA = E: CONS (391.0/63.0)
|   TEMPO_CONTA = D: CONS (477.0/95.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :             27
Size of the tree :      34

Time taken to build model: 0.23 seconds
Time taken to test model on training data: 0.22 seconds

=== Error on training data ===
Correctly Classified Instances       23097               83.1126 %
Incorrectly Classified Instances      4693               16.8874 %
Kappa statistic                          0.6623
Mean absolute error                      0.2544
Root mean squared error                  0.3567
Relative absolute error                 50.8842 %
Root relative squared error             71.3332 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
 10167  3728 |     a = CONS
   965 12930 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       23093               83.0982 %
Incorrectly Classified Instances      4697               16.9018 %
Kappa statistic                          0.662
Mean absolute error                      0.2558
Root mean squared error                  0.3581
Relative absolute error                 51.1614 %
Root relative squared error             71.6294 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
 10160  3735 |     a = CONS
   962 12933 |     b = INVEST
```

Log do 3º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree

------------------
UF = RS
|   IDADE = D: INVEST (3270.0/804.0)
|   IDADE = A: INVEST (1493.0/283.0)
|   IDADE = C
|   |   TEMPO_CONTA = F
|   |   |   TIPO = F: CONS (522.0/192.0)
|   |   |   TIPO = J: INVEST (157.0/60.0)
|   |   TEMPO_CONTA = A: INVEST (653.0/104.0)
|   |   TEMPO_CONTA = C: INVEST (645.0/181.0)
|   |   TEMPO_CONTA = B: INVEST (704.0/193.0)
|   |   TEMPO_CONTA = E: INVEST (788.0/251.0)
|   |   TEMPO_CONTA = D: INVEST (585.0/142.0)
|   IDADE = E: INVEST (4751.0/601.0)
|   IDADE = B: INVEST (3490.0/1075.0)
UF = PR
|   TEMPO_CONTA = F: CONS (2334.0/101.0)
|   TEMPO_CONTA = A
|   |   IDADE = D: CONS (165.0/58.0)
|   |   IDADE = A: CONS (133.0/47.0)
|   |   IDADE = C: CONS (190.0/68.0)
|   |   IDADE = E: INVEST (161.0/57.0)
```

```
|   |   IDADE = B: CONS (286.0/83.0)
|   TEMPO_CONTA = C: CONS (771.0/90.0)
|   TEMPO_CONTA = B: CONS (899.0/152.0)
|   TEMPO_CONTA = E: CONS (391.0/63.0)
|   TEMPO_CONTA = D: CONS (477.0/95.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :             24
Size of the tree :      30

Time taken to build model: 0.2 seconds
Time taken to test model on training data: 0.25 seconds

=== Error on training data ===
Correctly Classified Instances       23090               83.0874 %
Incorrectly Classified Instances      4700               16.9126 %
Kappa statistic                          0.6617
Mean absolute error                      0.2546
Root mean squared error                  0.3568
Relative absolute error                 50.9254 %
Root relative squared error             71.362  %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
 10144  3751 |     a = CONS
   949 12946 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       23070               83.0155 %
Incorrectly Classified Instances      4720               16.9845 %
Kappa statistic                          0.6603
Mean absolute error                      0.2556
Root mean squared error                  0.3579
Relative absolute error                 51.1294 %
Root relative squared error             71.5826 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
 10142  3753 |     a = CONS
   967 12928 |     b = INVEST
```

Log do 4º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS
|   IDADE = D: CRED (3424.0/804.0)
|   IDADE = A: CRED (263.0/66.0)
|   IDADE = C
|   |   TEMPO_CONTA = F
|   |   |   TIPO_CONTA = C
|   |   |   |   ESTADO_CIVIL = C: CRED (208.0/90.0)
|   |   |   |   ESTADO_CIVIL = S: CONS (38.0/16.0)
|   |   |   |   ESTADO_CIVIL = O: CRED (0.0)
|   |   |   |   ESTADO_CIVIL = U: CONS (27.0/12.0)
|   |   |   |   ESTADO_CIVIL = D: CRED (1.0)
|   |   |   |   ESTADO_CIVIL = V: CRED (0.0)
|   |   |   |   ESTADO_CIVIL = N: CRED (0.0)
|   |   |   TIPO_CONTA = I
|   |   |   |   ESTADO_CIVIL = C: CONS (208.0/68.0)
|   |   |   |   ESTADO_CIVIL = S: CRED (84.0/34.0)
|   |   |   |   ESTADO_CIVIL = O: CRED (3.0)
|   |   |   |   ESTADO_CIVIL = U: CONS (36.0/14.0)
|   |   |   |   ESTADO_CIVIL = D: CONS (13.0/6.0)
|   |   |   |   ESTADO_CIVIL = V: CONS (0.0)
|   |   |   |   ESTADO_CIVIL = N: CONS (0.0)
|   |   |   TIPO_CONTA = M: CONS (0.0)
|   |   |   TIPO_CONTA = S: CONS (0.0)
```

```
|   |   TEMPO_CONTA = A: CRED (576.0/73.0)
|   |   TEMPO_CONTA = C: CRED (546.0/156.0)
|   |   TEMPO_CONTA = B: CRED (589.0/118.0)
|   |   TEMPO_CONTA = E: CRED (723.0/198.0)
|   |   TEMPO_CONTA = D: CRED (474.0/93.0)
|   IDADE = E: CRED (4155.0/601.0)
|   IDADE = B: CRED (2977.0/869.0)
UF = PR: CONS (5366.0/964.0)
UF = MT: CONS (3560.0)
UF = SP: CONS (571.0)
UF = SC: CONS (160.0)

Number of Leaves  :             29
Size of the tree :      35

Time taken to build model: 0.22 seconds
Time taken to test model on training data: 0.19 seconds

=== Error on training data ===
Correctly Classified Instances       19820               82.5765 %
Incorrectly Classified Instances      4182               17.4235 %
Kappa statistic                          0.6515
Mean absolute error                      0.2681
Root mean squared error                  0.3661
Relative absolute error                 53.6241 %
Root relative squared error             73.2285 %
Total Number of Instances            24002

=== Confusion Matrix ===
     a     b   <-- classified as
  8899  3102 |     a = CONS
  1080 10921 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       19763               82.339  %
Incorrectly Classified Instances      4239               17.661  %
Kappa statistic                          0.6468
Mean absolute error                      0.2724
Root mean squared error                  0.3694
Relative absolute error                 54.4744 %
Root relative squared error             73.8715 %
Total Number of Instances            24002

=== Confusion Matrix ===
     a     b   <-- classified as
  8802  3199 |     a = CONS
  1040 10961 |     b = CRED
```

Log do 5º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33984.0/9248.0)
UF = PR
|   TIPO = F
|   |   TEMPO_CONTA = F
|   |   |   IDADE = D: CRED (536.0/141.0)
|   |   |   IDADE = A: CRED (1.0)
|   |   |   IDADE = C: CRED (326.0/157.0)
|   |   |   IDADE = E: CART (2614.0/806.0)
|   |   |   IDADE = B: CART (26.0/10.0)
|   |   TEMPO_CONTA = A: CRED (2547.0/1036.0)
|   |   TEMPO_CONTA = C
|   |   |   IDADE = D: CRED (465.0/215.0)
|   |   |   IDADE = A: CART (0.0)
|   |   |   IDADE = C: CART (654.0/269.0)
|   |   |   IDADE = E: CRED (428.0/130.0)
|   |   |   IDADE = B: CART (583.0/237.0)
|   |   TEMPO_CONTA = B
|   |   |   IDADE = D: CRED (604.0/268.0)
|   |   |   IDADE = A: CART (62.0/30.0)
|   |   |   IDADE = C: CART (907.0/411.0)
|   |   |   IDADE = E: CRED (628.0/201.0)
|   |   |   IDADE = B: CART (937.0/402.0)
|   |   TEMPO_CONTA = E: CRED (1165.0/389.0)
```

```
|   |   TEMPO_CONTA = D
|   |   |   IDADE = D: CRED (488.0/198.0)
|   |   |   IDADE = A: CRED (0.0)
|   |   |   IDADE = C: CART (627.0/303.0)
|   |   |   IDADE = E: CRED (442.0/133.0)
|   |   |   IDADE = B: CART (376.0/161.0)
|   TIPO = J: CRED (781.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves  :             27
Size of the tree :      34

Time taken to build model: 0.38 seconds
Time taken to test model on training data: 0.47 seconds

=== Error on training data ===
Correctly Classified Instances       51071               77.5966 %
Incorrectly Classified Instances     14745               22.4034 %
Kappa statistic                          0.5519
Mean absolute error                      0.3061
Root mean squared error                  0.3912
Relative absolute error                 61.2248 %
Root relative squared error             78.2463 %
Total Number of Instances            65816

=== Confusion Matrix ===
     a     b   <-- classified as
 20792 12116 |     a = CART
  2629 30279 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       51067               77.5906 %
Incorrectly Classified Instances     14749               22.4094 %
Kappa statistic                          0.5518
Mean absolute error                      0.3062
Root mean squared error                  0.3914
Relative absolute error                 61.2491 %
Root relative squared error             78.2785 %
Total Number of Instances            65816

=== Confusion Matrix ===
     a     b   <-- classified as
 20788 12120 |     a = CART
  2629 30279 |     b = CRED
```

Log do 6º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33984.0/9248.0)
UF = PR
|   TIPO = F
|   |   TEMPO_CONTA = F
|   |   |   IDADE = D: CRED (536.0/141.0)
|   |   |   IDADE = A: CRED (1.0)
|   |   |   IDADE = C: CRED (326.0/157.0)
|   |   |   IDADE = E: CART (2614.0/806.0)
|   |   |   IDADE = B
|   |   |   |   TIPO_CONTA = C: CART (14.0/3.0)
|   |   |   |   TIPO_CONTA = I: CRED (12.0/5.0)
|   |   |   |   TIPO_CONTA = M: CART (0.0)
|   |   |   |   TIPO_CONTA = S: CART (0.0)
|   |   TEMPO_CONTA = A: CRED (2547.0/1036.0)
|   |   TEMPO_CONTA = C
|   |   |   IDADE = D: CRED (465.0/215.0)
|   |   |   IDADE = A: CART (0.0)
|   |   |   IDADE = C: CART (654.0/269.0)
|   |   |   IDADE = E: CRED (428.0/130.0)
|   |   |   IDADE = B: CART (583.0/237.0)
|   |   TEMPO_CONTA = B
```

```
|   |   |   IDADE = D: CRED (604.0/268.0)
|   |   |   IDADE = A: CART (62.0/30.0)
|   |   |   IDADE = C: CART (907.0/411.0)
|   |   |   IDADE = E: CRED (628.0/201.0)
|   |   |   IDADE = B: CART (937.0/402.0)
|   |   TEMPO_CONTA = E: CRED (1165.0/389.0)
|   |   TEMPO_CONTA = D
|   |   |   IDADE = D: CRED (488.0/198.0)
|   |   |   IDADE = A: CRED (0.0)
|   |   |   IDADE = C: CART (627.0/303.0)
|   |   |   IDADE = E: CRED (442.0/133.0)
|   |   |   IDADE = B: CART (376.0/161.0)
|   TIPO = J: CRED (781.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves  :             30
Size of the tree :      38

Time taken to build model: 0.44 seconds
Time taken to test model on training data: 0.52 seconds

=== Error on training data ===
Correctly Classified Instances       51073               77.5997 %
Incorrectly Classified Instances     14743               22.4003 %
Kappa statistic                          0.552
Mean absolute error                      0.3061
Root mean squared error                  0.3912
Relative absolute error                 61.2194 %
Root relative squared error             78.2429 %
Total Number of Instances            65816

=== Confusion Matrix ===
    a     b   <-- classified as
 20787 12121 |     a = CART
  2622 30286 |     b = CRED


=== Stratified cross-validation ===
Correctly Classified Instances       51053               77.5693 %
Incorrectly Classified Instances     14763               22.4307 %
Kappa statistic                          0.5514
Mean absolute error                      0.3063
Root mean squared error                  0.3914
Relative absolute error                 61.2509 %
Root relative squared error             78.2848 %
Total Number of Instances            65816

=== Confusion Matrix ===
    a     b   <-- classified as
 20757 12151 |     a = CART
  2612 30296 |     b = CRED
```

Log do 7º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: INVEST (35053.0/9248.0)
UF = PR
|   TIPO = F
|   |   IDADE = D: INVEST (2259.0/1044.0)
|   |   IDADE = A: INVEST (342.0/102.0)
|   |   IDADE = C: CART (3091.0/1257.0)
|   |   IDADE = E: INVEST (4929.0/2399.0)
|   |   IDADE = B: CART (2877.0/1231.0)
|   TIPO = J: INVEST (630.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves  :             10
Size of the tree :      13

Time taken to build model: 0.31 seconds
```

Time taken to test model on training data: 0.5 seconds

=== Error on training data ===
Correctly Classified Instances 50535 76.7822 %
Incorrectly Classified Instances 15281 23.2178 %
Kappa statistic 0.5356
Mean absolute error 0.3076
Root mean squared error 0.3922
Relative absolute error 61.5211 %
Root relative squared error 78.4354 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
20115 12793 | a = CART
2488 30420 | b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances 50535 76.7822 %
Incorrectly Classified Instances 15281 23.2178 %
Kappa statistic 0.5356
Mean absolute error 0.3076
Root mean squared error 0.3922
Relative absolute error 61.5273 %
Root relative squared error 78.4433 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
20115 12793 | a = CART
2488 30420 | b = INVEST

Log do 8º experimento
Options: -C 0.25 -M 2
J48 pruned tree

UF = RS: INVEST (35053.0/9248.0)
UF = PR
| TIPO = F
| | IDADE = D: INVEST (2259.0/1044.0)
| | IDADE = A: INVEST (342.0/102.0)
| | IDADE = C: CART (3091.0/1257.0)
| | IDADE = E: INVEST (4929.0/2399.0)
| | IDADE = B: CART (2877.0/1231.0)
| TIPO = J: INVEST (630.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves : 10
Size of the tree : 13

Time taken to build model: 0.34 seconds
Time taken to test model on training data: 0.47 seconds

=== Error on training data ===
Correctly Classified Instances 50535 76.7822 %
Incorrectly Classified Instances 15281 23.2178 %
Kappa statistic 0.5356
Mean absolute error 0.3076
Root mean squared error 0.3922
Relative absolute error 61.5211 %
Root relative squared error 78.4354 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
20115 12793 | a = CART
2488 30420 | b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances 50500 76.7291 %

Incorrectly Classified Instances 15316 23.2709 %
Kappa statistic 0.5346
Mean absolute error 0.3076
Root mean squared error 0.3922
Relative absolute error 61.5279 %
Root relative squared error 78.4453 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
20427 12481 | a = CART
2835 30073 | b = INVEST

Log do 9º experimento
Options: -C 0.25 -M 2
J48 pruned tree

UF = RS: CRED (33984.0/9248.0)
UF = PR
| TIPO = F
| | IDADE = D: CRED (2798.0/1044.0)
| | IDADE = A: CRED (222.0/102.0)
| | IDADE = C
| | | TIPO_CONTA = C: CRED (1405.0/690.0)
| | | TIPO_CONTA = I: CART (2183.0/1039.0)
| | | TIPO_CONTA = M: CART (0.0)
| | | TIPO_CONTA = S: CRED (1.0)
| | IDADE = E
| | | TIPO_CONTA = C: CRED (1960.0/801.0)
| | | TIPO_CONTA = I: CART (2812.0/1215.0)
| | | TIPO_CONTA = M: CART (1.0)
| | | TIPO_CONTA = S: CRED (2.0)
| | IDADE = B: CART (3032.0/1386.0)
| TIPO = J: CRED (781.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves : 16
Size of the tree : 21

Time taken to build model: 0.38 seconds
Time taken to test model on training data: 0.47 seconds

=== Error on training data ===
Correctly Classified Instances 50291 76.4115 %
Incorrectly Classified Instances 15525 23.5885 %
Kappa statistic 0.5282
Mean absolute error 0.3116
Root mean squared error 0.3947
Relative absolute error 62.3112 %
Root relative squared error 78.9374 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
21023 11885 | a = CART
3640 29268 | b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances 50269 76.3781 %
Incorrectly Classified Instances 15547 23.6219 %
Kappa statistic 0.5276
Mean absolute error 0.3116
Root mean squared error 0.3948
Relative absolute error 62.3259 %
Root relative squared error 78.9545 %
Total Number of Instances 65816

=== Confusion Matrix ===
a b <-- classified as
21078 11830 | a = CART
3717 29191 | b = CRED

Log do 10º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: INVEST (36789.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C
|   |   IDADE = D: INVEST (1686.0/746.0)
|   |   IDADE = A: INVEST (8.0/1.0)
|   |   IDADE = C: CART (1829.0/797.0)
|   |   IDADE = E: INVEST (3030.0/1376.0)
|   |   IDADE = B: CART (803.0/314.0)
|   ESTADO_CIVIL = S
|   |   IDADE = D: INVEST (561.0/171.0)
|   |   IDADE = A: INVEST (429.0/109.0)
|   |   IDADE = C: CART (1267.0/610.0)
|   |   IDADE = E: CART (1447.0/705.0)
|   |   IDADE = B: INVEST (2287.0/1119.0)
|   ESTADO_CIVIL = O
|   |   IDADE = D: INVEST (58.0/23.0)
|   |   IDADE = A: CART (0.0)
|   |   IDADE = C: CART (52.0/21.0)
|   |   IDADE = E: CART (111.0/44.0)
|   |   IDADE = B: CART (27.0/4.0)
|   ESTADO_CIVIL = U
|   |   IDADE = D: INVEST (177.0/79.0)
|   |   IDADE = A: INVEST (8.0/1.0)
|   |   IDADE = C: CART (297.0/139.0)
|   |   IDADE = E: CART (208.0/101.0)
|   |   IDADE = B: CART (247.0/115.0)
|   ESTADO_CIVIL = D
|   |   IDADE = D: CART (143.0/71.0)
|   |   IDADE = A: INVEST (0.0)
|   |   IDADE = C: CART (112.0/50.0)
|   |   IDADE = E: INVEST (251.0/108.0)
|   |   IDADE = B: INVEST (24.0/10.0)
|   ESTADO_CIVIL = V: INVEST (845.0/140.0)
|   ESTADO_CIVIL = N: INVEST (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             31
Size of the tree :      38

Time taken to build model: 0.31 seconds
Time taken to test model on training data: 0.48 seconds

=== Error on training data ===
Correctly Classified Instances       53370               75.9067 %
Incorrectly Classified Instances     16940               24.0933 %
Kappa statistic                          0.5181
Mean absolute error                      0.3162
Root mean squared error                  0.3976
Relative absolute error                 63.2349 %
Root relative squared error             79.5204 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 21186 13969 |     a = CART
  2971 32184 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       53349               75.8768 %
Incorrectly Classified Instances     16961               24.1232 %
Kappa statistic                          0.5175
Mean absolute error                      0.3163
Root mean squared error                  0.3978
Relative absolute error                 63.2685 %
Root relative squared error             79.5671 %
Total Number of Instances            70310
```

```
=== Confusion Matrix ===
     a     b   <-- classified as
 21141 14014 |     a = CART
  2947 32208 |     b = INVEST
```

Log do 11º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33984.0/9248.0)
UF = PR
|   TIPO = F
|   |   IDADE = D: CRED (2798.0/1044.0)
|   |   IDADE = A: CRED (222.0/102.0)
|   |   IDADE = C: CART (3589.0/1755.0)
|   |   IDADE = E: CART (4775.0/2376.0)
|   |   IDADE = B: CART (3032.0/1386.0)
|   TIPO = J: CRED (781.0)
UF = MT: CART (11889.0)
UF = SP: CART (1959.0)
UF = SC: CART (2787.0)

Number of Leaves  :             10
Size of the tree :      13

Time taken to build model: 0.3 seconds
Time taken to test model on training data: 0.47 seconds

=== Error on training data ===
Correctly Classified Instances       49905               75.825  %
Incorrectly Classified Instances     15911               24.175  %
Kappa statistic                          0.5165
Mean absolute error                      0.3125
Root mean squared error                  0.3953
Relative absolute error                 62.501  %
Root relative squared error             79.0576 %
Total Number of Instances            65816

=== Confusion Matrix ===
     a     b   <-- classified as
 22514 10394 |     a = CART
  5517 27391 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       49871               75.7734 %
Incorrectly Classified Instances     15945               24.2266 %
Kappa statistic                          0.5155
Mean absolute error                      0.3125
Root mean squared error                  0.3953
Relative absolute error                 62.5092 %
Root relative squared error             79.0684 %
Total Number of Instances            65816

=== Confusion Matrix ===
     a     b   <-- classified as
 22275 10633 |     a = CART
  5312 27596 |     b = CRED
```

Log do 12º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33961.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C: CRED (10163.0/3644.0)
|   ESTADO_CIVIL = S
|   |   IDADE = D: CRED (566.0/171.0)
|   |   IDADE = A: CRED (271.0/109.0)
|   |   IDADE = C: CRED (1452.0/657.0)
|   |   IDADE = E
|   |   |   TIPO_CONTA = C: CRED (111.0/49.0)
|   |   |   TIPO_CONTA = I: CART (970.0/278.0)
|   |   |   TIPO_CONTA = M: CART (1.0)
|   |   |   TIPO_CONTA = S: CART (0.0)
```

```
|   |  IDADE = B: CRED (2424.0/1119.0)
|  ESTADO_CIVIL = O
|   |  IDADE = D: CRED (85.0/23.0)
|   |  IDADE = A: CRED (0.0)
|   |  IDADE = C: CRED (79.0/31.0)
|   |  IDADE = E
|   |   |  TIPO_CONTA = C: CRED (7.0/2.0)
|   |   |  TIPO_CONTA = I: CART (115.0/50.0)
|   |   |  TIPO_CONTA = M: CART (0.0)
|   |   |  TIPO_CONTA = S: CART (0.0)
|   |  IDADE = B: CART (37.0/14.0)
|  ESTADO_CIVIL = U: CRED (1186.0/477.0)
|  ESTADO_CIVIL = D: CRED (610.0/252.0)
|  ESTADO_CIVIL = V: CRED (658.0/140.0)
|  ESTADO_CIVIL = N: CRED (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             25
Size of the tree :      31

Time taken to build model: 0.39 seconds
Time taken to test model on training data: 0.5 seconds

=== Error on training data ===
Correctly Classified Instances       53208               75.6763 %
Incorrectly Classified Instances     17102               24.3237 %
Kappa statistic                          0.5135
Mean absolute error                      0.3248
Root mean squared error                  0.403
Relative absolute error                 64.9559 %
Root relative squared error             80.5952 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18395 16760 |     a = CART
   342 34813 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       53196               75.6592 %
Incorrectly Classified Instances     17114               24.3408 %
Kappa statistic                          0.5132
Mean absolute error                      0.3249
Root mean squared error                  0.4032
Relative absolute error                 64.9828 %
Root relative squared error             80.631  %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18402 16753 |     a = CART
   361 34794 |     b = CRED
```

Log do 13º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33961.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C: CRED (10163.0/3644.0)
|   ESTADO_CIVIL = S
|   |   IDADE = D: CRED (566.0/171.0)
|   |   IDADE = A: CRED (271.0/109.0)
|   |   IDADE = C: CRED (1452.0/657.0)
|   |   IDADE = E
|   |   |   TIPO_CONTA = C: CRED (111.0/49.0)
|   |   |   TIPO_CONTA = I: CART (970.0/278.0)
|   |   |   TIPO_CONTA = M: CART (1.0)
|   |   |   TIPO_CONTA = S: CART (0.0)
|   |   IDADE = B: CRED (2424.0/1119.0)
```

```
|   ESTADO_CIVIL = O
|   |   IDADE = D: CRED (85.0/23.0)
|   |   IDADE = A: CRED (0.0)
|   |   IDADE = C: CRED (79.0/31.0)
|   |   IDADE = E
|   |   |   TIPO_CONTA = C
|   |   |   |   SEXO = M: CART (2.0)
|   |   |   |   SEXO = F: CRED (5.0)
|   |   |   TIPO_CONTA = I: CART (115.0/50.0)
|   |   |   TIPO_CONTA = M: CART (0.0)
|   |   |   TIPO_CONTA = S: CART (0.0)
|   |   IDADE = B
|   |   |   SEXO = M: CRED (10.0/3.0)
|   |   |   SEXO = F: CART (27.0/7.0)
|   ESTADO_CIVIL = U: CRED (1186.0/477.0)
|   ESTADO_CIVIL = D: CRED (610.0/252.0)
|   ESTADO_CIVIL = V: CRED (658.0/140.0)
|   ESTADO_CIVIL = N: CRED (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             27
Size of the tree :      35

Time taken to build model: 0.48 seconds
Time taken to test model on training data: 0.53 seconds

=== Error on training data ===
Correctly Classified Instances       53214               75.6848 %
Incorrectly Classified Instances     17096               24.3152 %
Kappa statistic                          0.5137
Mean absolute error                      0.3247
Root mean squared error                  0.4029
Relative absolute error                 64.9397 %
Root relative squared error             80.5852 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18394 16761 |     a = CART
   335 34820 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       53194               75.6564 %
Incorrectly Classified Instances     17116               24.3436 %
Kappa statistic                          0.5131
Mean absolute error                      0.3248
Root mean squared error                  0.4031
Relative absolute error                 64.9661 %
Root relative squared error             80.6256 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18408 16747 |     a = CART
   369 34786 |     b = CRED
```

Log do 14º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33961.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C: CRED (10163.0/3644.0)
|   ESTADO_CIVIL = S
|   |   IDADE = D: CRED (566.0/171.0)
|   |   IDADE = A: CRED (271.0/109.0)
|   |   IDADE = C: CRED (1452.0/657.0)
|   |   IDADE = E: CART (1082.0/340.0)
|   |   IDADE = B: CRED (2424.0/1119.0)
|   ESTADO_CIVIL = O
|   |   IDADE = D: CRED (85.0/23.0)
|   |   IDADE = A: CRED (0.0)
|   |   IDADE = C: CRED (79.0/31.0)
```

```
|   |   IDADE = E: CART (122.0/55.0)
|   |   IDADE = B
|   |   |   SEXO = M: CRED (10.0/3.0)
|   |   |   SEXO = F: CART (27.0/7.0)
|   ESTADO_CIVIL = U: CRED (1186.0/477.0)
|   ESTADO_CIVIL = D: CRED (610.0/252.0)
|   ESTADO_CIVIL = V: CRED (658.0/140.0)
|   ESTADO_CIVIL = N: CRED (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             20
Size of the tree :      25

Time taken to build model: 0.39 seconds
Time taken to test model on training data: 0.49 seconds

=== Error on training data ===
Correctly Classified Instances       53196               75.6592 %
Incorrectly Classified Instances     17114               24.3408 %
Kappa statistic                          0.5132
Mean absolute error                      0.325
Root mean squared error                  0.4031
Relative absolute error                 64.9932 %
Root relative squared error             80.6184 %
Total Number of Instances            70310

=== Confusion Matrix ===
    a     b   <-- classified as
 18443 16712 |     a = CART
   402 34753 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       53191               75.6521 %
Incorrectly Classified Instances     17119               24.3479 %
Kappa statistic                          0.513
Mean absolute error                      0.3251
Root mean squared error                  0.4032
Relative absolute error                 65.014  %
Root relative squared error             80.6459 %
Total Number of Instances            70310

=== Confusion Matrix ===
    a     b   <-- classified as
 18448 16707 |     a = CART
   412 34743 |     b = CRED
```

Log do 15º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: CRED (33961.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C: CRED (10163.0/3644.0)
|   ESTADO_CIVIL = S
|   |   IDADE = D: CRED (566.0/171.0)
|   |   IDADE = A: CRED (271.0/109.0)
|   |   IDADE = C: CRED (1452.0/657.0)
|   |   IDADE = E: CART (1082.0/340.0)
|   |   IDADE = B: CRED (2424.0/1119.0)
|   ESTADO_CIVIL = O
|   |   IDADE = D: CRED (85.0/23.0)
|   |   IDADE = A: CRED (0.0)
|   |   IDADE = C: CRED (79.0/31.0)
|   |   IDADE = E: CART (122.0/55.0)
|   |   IDADE = B: CART (37.0/14.0)
|   ESTADO_CIVIL = U: CRED (1186.0/477.0)
|   ESTADO_CIVIL = D: CRED (610.0/252.0)
|   ESTADO_CIVIL = V: CRED (658.0/140.0)
|   ESTADO_CIVIL = N: CRED (0.0)
UF = MT: CART (12628.0)
```

```
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             19
Size of the tree :      23

Time taken to build model: 0.31 seconds
Time taken to test model on training data: 0.48 seconds

=== Error on training data ===
Correctly Classified Instances       53192               75.6535 %
Incorrectly Classified Instances     17118               24.3465 %
Kappa statistic                          0.5131
Mean absolute error                      0.325
Root mean squared error                  0.4031
Relative absolute error                 65.0013 %
Root relative squared error             80.6234 %
Total Number of Instances            70310

=== Confusion Matrix ===
    a     b   <-- classified as
 18446 16709 |     a = CART
   409 34746 |     b = CRED

=== Stratified cross-validation ===
Correctly Classified Instances       53187               75.6464 %
Incorrectly Classified Instances     17123               24.3536 %
Kappa statistic                          0.5129
Mean absolute error                      0.3251
Root mean squared error                  0.4033
Relative absolute error                 65.0215 %
Root relative squared error             80.6504 %
Total Number of Instances            70310

=== Confusion Matrix ===
    a     b   <-- classified as
 18451 16704 |     a = CART
   419 34736 |     b = CRED
```

Log do 16º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: INVEST (36789.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C
|   |   TIPO_CONTA = C
|   |   |   SEXO = M: INVEST (3124.0/1510.0)
|   |   |   SEXO = F: CART (805.0/373.0)
|   |   TIPO_CONTA = I
|   |   |   SEXO = M: CART (2297.0/1111.0)
|   |   |   SEXO = F: INVEST (1121.0/516.0)
|   |   TIPO_CONTA = M: INVEST (0.0)
|   |   TIPO_CONTA = S: INVEST (9.0)
|   ESTADO_CIVIL = S: INVEST (5991.0/2798.0)
|   ESTADO_CIVIL = O
|   |   TIPO_CONTA = C: INVEST (11.0/2.0)
|   |   TIPO_CONTA = I: CART (236.0/94.0)
|   |   TIPO_CONTA = M: CART (0.0)
|   |   TIPO_CONTA = S: INVEST (1.0)
|   ESTADO_CIVIL = U
|   |   TIPO_CONTA = C: CART (380.0/176.0)
|   |   TIPO_CONTA = I
|   |   |   SEXO = M: INVEST (335.0/160.0)
|   |   |   SEXO = F: CART (220.0/107.0)
|   |   TIPO_CONTA = M: CART (0.0)
|   |   TIPO_CONTA = S: INVEST (2.0)
|   ESTADO_CIVIL = D
|   |   TIPO_CONTA = C: INVEST (53.0/17.0)
|   |   TIPO_CONTA = I
|   |   |   SEXO = M: CART (283.0/135.0)
|   |   |   SEXO = F: INVEST (192.0/87.0)
|   |   TIPO_CONTA = M: INVEST (0.0)
|   |   TIPO_CONTA = S: INVEST (2.0)
|   ESTADO_CIVIL = V: INVEST (845.0/140.0)
```

```
|   ESTADO_CIVIL = N: INVEST (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             27
Size of the tree :      37

Time taken to build model: 0.39 seconds
Time taken to test model on training data: 0.5 seconds

=== Error on training data ===
Correctly Classified Instances       52998               75.3776 %
Incorrectly Classified Instances     17312               24.6224 %
Kappa statistic                          0.5076
Mean absolute error                      0.3181
Root mean squared error                  0.3988
Relative absolute error                 63.6169 %
Root relative squared error             79.7602 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 19839 15316 |     a = CART
  1996 33159 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       52936               75.2894 %
Incorrectly Classified Instances     17374               24.7106 %
Kappa statistic                          0.5058
Mean absolute error                      0.3182
Root mean squared error                  0.399
Relative absolute error                 63.6396 %
Root relative squared error             79.7995 %
Total Number of Instances            70310

=== Confusion Matrix ===
     a     b   <-- classified as
 19808 15347 |     a = CART
  2027 33128 |     b = INVEST
```

Log do 17º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: INVEST (36789.0/10086.0)
UF = PR
|   ESTADO_CIVIL = C: INVEST (7356.0/3644.0)
|   ESTADO_CIVIL = S: INVEST (5991.0/2798.0)
|   ESTADO_CIVIL = O: CART (248.0/104.0)
|   ESTADO_CIVIL = U: CART (937.0/460.0)
|   ESTADO_CIVIL = D: INVEST (530.0/252.0)
|   ESTADO_CIVIL = V: INVEST (845.0/140.0)
|   ESTADO_CIVIL = N: INVEST (0.0)
UF = MT: CART (12628.0)
UF = SP: CART (2033.0)
UF = SC: CART (2953.0)

Number of Leaves  :             11
Size of the tree :      13

Time taken to build model: 0.3 seconds
Time taken to test model on training data: 0.48 seconds

=== Error on training data ===
Correctly Classified Instances       52826               75.133  %
Incorrectly Classified Instances     17484               24.867  %
Kappa statistic                          0.5027
Mean absolute error                      0.3184
Root mean squared error                  0.399
Relative absolute error                 63.6864 %
Root relative squared error             79.8037 %
```

```
Total Number of Instances           70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18235 16920 |     a = CART
   564 34591 |     b = INVEST

=== Stratified cross-validation ===
Correctly Classified Instances       52670               74.9111 %
Incorrectly Classified Instances     17640               25.0889 %
Kappa statistic                          0.4982
Mean absolute error                      0.3195
Root mean squared error                  0.3998
Relative absolute error                 63.9027 %
Root relative squared error             79.9612 %
Total Number of Instances           70310

=== Confusion Matrix ===
     a     b   <-- classified as
 18194 16961 |     a = CART
   679 34476 |     b = INVEST
```

Log do 18º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

TIPO = F
|   UF = RS: SEG (10720.0/3308.0)
|   UF = PR
|   |   TEMPO_CONTA = F
|   |   |   IDADE = D: SEG (961.0/327.0)
|   |   |   IDADE = A: CONS (2.0)
|   |   |   IDADE = C: SEG (477.0/193.0)
|   |   |   IDADE = E
|   |   |   |   TIPO_CONTA = C: SEG (1121.0/544.0)
|   |   |   |   TIPO_CONTA = I: CONS (1349.0/343.0)
|   |   |   |   TIPO_CONTA = M: CONS (3.0)
|   |   |   |   TIPO_CONTA = S: SEG (6.0/1.0)
|   |   |   IDADE = B: SEG (21.0/6.0)
|   |   TEMPO_CONTA = A: SEG (1603.0/512.0)
|   |   TEMPO_CONTA = C
|   |   |   IDADE = D: SEG (312.0/122.0)
|   |   |   IDADE = A: SEG (0.0)
|   |   |   IDADE = C: SEG (379.0/129.0)
|   |   |   IDADE = E: SEG (201.0/80.0)
|   |   |   IDADE = B: CONS (431.0/203.0)
|   |   TEMPO_CONTA = B: SEG (1637.0/597.0)
|   |   TEMPO_CONTA = E: SEG (1158.0/322.0)
|   |   TEMPO_CONTA = D: SEG (1224.0/330.0)
|   UF = MT: CONS (3560.0)
|   UF = SP: CONS (571.0)
|   UF = SC: CONS (160.0)
TIPO = J: CONS (1894.0)

Number of Leaves  :             22
Size of the tree :      28

Time taken to build model: 0.3 seconds
Time taken to test model on training data: 0.27 seconds

=== Error on training data ===
Correctly Classified Instances       20773               74.7499 %
Incorrectly Classified Instances      7017               25.2501 %
Kappa statistic                          0.495
Mean absolute error                      0.3364
Root mean squared error                  0.4102
Relative absolute error                 67.2895 %
Root relative squared error             82.0302 %
Total Number of Instances           27790

=== Confusion Matrix ===
     a     b   <-- classified as
  7424  6471 |     a = CONS
   546 13349 |     b = SEG
```

```
=== Stratified cross-validation ===
Correctly Classified Instances       20750               74.6671 %
Incorrectly Classified Instances      7040               25.3329 %
Kappa statistic                          0.4933
Mean absolute error                      0.3368
Root mean squared error                  0.4106
Relative absolute error                 67.3569 %
Root relative squared error             82.1131 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
  7387  6508 |     a = CONS
   532 13363 |     b = SEG
```

Log do 19º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS
|   TIPO = F: SEG (10720.0/3308.0)
|   TIPO = J: CONS (716.0)
UF = PR
|   TIPO = F
|   |   TEMPO_CONTA = F
|   |   |   IDADE = D: SEG (961.0/327.0)
|   |   |   IDADE = A: CONS (2.0)
|   |   |   IDADE = C: SEG (477.0/193.0)
|   |   |   IDADE = E: CONS (2479.0/925.0)
|   |   |   IDADE = B: SEG (21.0/6.0)
|   |   TEMPO_CONTA = A: SEG (1603.0/512.0)
|   |   TEMPO_CONTA = C
|   |   |   IDADE = D: SEG (312.0/122.0)
|   |   |   IDADE = A: SEG (0.0)
|   |   |   IDADE = C: SEG (379.0/129.0)
|   |   |   IDADE = E: SEG (201.0/80.0)
|   |   |   IDADE = B: CONS (431.0/203.0)
|   |   TEMPO_CONTA = B: SEG (1637.0/597.0)
|   |   TEMPO_CONTA = E: SEG (1158.0/322.0)
|   |   TEMPO_CONTA = D: SEG (1224.0/330.0)
|   TIPO = J: CONS (544.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :             20
Size of the tree :      26

Time taken to build model: 0.2 seconds
Time taken to test model on training data: 0.2 seconds

=== Error on training data ===
Correctly Classified Instances       20736               74.6168 %
Incorrectly Classified Instances      7054               25.3832 %
Kappa statistic                          0.4923
Mean absolute error                      0.3396
Root mean squared error                  0.412
Relative absolute error                 67.9116 %
Root relative squared error             82.4085 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
  7969  5926 |     a = CONS
  1128 12767 |     b = SEG

=== Stratified cross-validation ===
Correctly Classified Instances       20713               74.534  %
Incorrectly Classified Instances      7077               25.466  %
Kappa statistic                          0.4907
Mean absolute error                      0.3399
Root mean squared error                  0.4124
```

```
Relative absolute error               67.974  %
Root relative squared error           82.4837 %
Total Number of Instances             27790

=== Confusion Matrix ===
    a     b   <-- classified as
 7932  5963 |     a = CONS
 1114 12781 |     b = SEG
```

Log do 20º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS
|   TIPO = F: SEG (10720.0/3308.0)
|   TIPO = J: CONS (716.0)
UF = PR
|   TIPO = F
|   |   IDADE = D: SEG (2579.0/849.0)
|   |   IDADE = A
|   |   |   TIPO_CONTA = C: SEG (14.0/5.0)
|   |   |   TIPO_CONTA = I: SEG (144.0/39.0)
|   |   |   TIPO_CONTA = M: CONS (6.0)
|   |   |   TIPO_CONTA = S: SEG (0.0)
|   |   IDADE = C: SEG (2514.0/801.0)
|   |   IDADE = E
|   |   |   TIPO_CONTA = C: SEG (1517.0/668.0)
|   |   |   TIPO_CONTA = I: CONS (2097.0/904.0)
|   |   |   TIPO_CONTA = M: CONS (3.0)
|   |   |   TIPO_CONTA = S: SEG (6.0/1.0)
|   |   IDADE = B: SEG (2005.0/837.0)
|   TIPO = J: CONS (544.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :             17
Size of the tree :      23

Time taken to build model: 0.2 seconds
Time taken to test model on training data: 0.2 seconds

=== Error on training data ===
Correctly Classified Instances       20378               73.3285 %
Incorrectly Classified Instances      7412               26.6715 %
Kappa statistic                          0.4666
Mean absolute error                      0.3462
Root mean squared error                  0.4161
Relative absolute error                 69.244  %
Root relative squared error             83.213  %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 7387  6508 |     a = CONS
  904 12991 |     b = SEG

=== Stratified cross-validation ===
Correctly Classified Instances       20378               73.3285 %
Incorrectly Classified Instances      7412               26.6715 %
Kappa statistic                          0.4666
Mean absolute error                      0.3464
Root mean squared error                  0.4162
Relative absolute error                 69.2712 %
Root relative squared error             83.2487 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 7387  6508 |     a = CONS
  904 12991 |     b = SEG
```

Log do 21º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
```

```
------------------

UF = RS
|   TIPO = F: SEG (10720.0/3308.0)
|   TIPO = J: CONS (716.0)
UF = PR
|   TIPO = F
|   |   IDADE = D: SEG (2579.0/849.0)
|   |   IDADE = A: SEG (164.0/50.0)
|   |   IDADE = C: SEG (2514.0/801.0)
|   |   IDADE = E: CONS (3623.0/1758.0)
|   |   IDADE = B: SEG (2005.0/837.0)
|   TIPO = J: CONS (544.0)
UF = MT: CONS (3923.0)
UF = SP: CONS (644.0)
UF = SC: CONS (358.0)

Number of Leaves  :             11
Size of the tree :      15

Time taken to build model: 0.17 seconds
Time taken to test model on training data: 0.2 seconds

=== Error on training data ===
Correctly Classified Instances       20187               72.6412 %
Incorrectly Classified Instances      7603               27.3588 %
Kappa statistic                          0.4528
Mean absolute error                      0.3476
Root mean squared error                  0.4169
Relative absolute error                 69.5188 %
Root relative squared error             83.3779 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
  8050  5845 |     a = CONS
  1758 12137 |     b = SEG

=== Stratified cross-validation ===
Correctly Classified Instances       20187               72.6412 %
Incorrectly Classified Instances      7603               27.3588 %
Kappa statistic                          0.4528
Mean absolute error                      0.3477
Root mean squared error                  0.417
Relative absolute error                 69.5371 %
Root relative squared error             83.4024 %
Total Number of Instances            27790

=== Confusion Matrix ===
     a     b   <-- classified as
  8050  5845 |     a = CONS
  1758 12137 |     b = SEG
```

Log do 22º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: SEG (9755.0/3308.0)
UF = PR
|   TEMPO_CONTA = F
|   |   IDADE = D: SEG (869.0/327.0)
|   |   IDADE = A: CONS (2.0)
|   |   IDADE = C: SEG (444.0/193.0)
|   |   IDADE = E: CONS (2334.0/780.0)
|   |   IDADE = B: SEG (17.0/6.0)
|   TEMPO_CONTA = A: SEG (1449.0/512.0)
|   TEMPO_CONTA = C
|   |   IDADE = D: SEG (293.0/122.0)
|   |   IDADE = A: SEG (0.0)
|   |   IDADE = C: SEG (352.0/129.0)
|   |   IDADE = E: SEG (181.0/80.0)
|   |   IDADE = B: CONS (407.0/179.0)
```

```
|   TEMPO_CONTA = B: SEG (1511.0/597.0)
|   TEMPO_CONTA = E: SEG (1029.0/322.0)
|   TEMPO_CONTA = D: SEG (1068.0/330.0)
UF = MT: CONS (3560.0)
UF = SP: CONS (571.0)
UF = SC: CONS (160.0)

Number of Leaves  :             18
Size of the tree :      22

Time taken to build model: 0.19 seconds
Time taken to test model on training data: 0.16 seconds

=== Error on training data ===
Correctly Classified Instances       17117               71.3149 %
Incorrectly Classified Instances      6885               28.6851 %
Kappa statistic                          0.4263
Mean absolute error                      0.3718
Root mean squared error                  0.4312
Relative absolute error                 74.3576 %
Root relative squared error             86.2308 %
Total Number of Instances            24002

=== Confusion Matrix ===
     a     b   <-- classified as
  6075  5926 |     a = CONS
   959 11042 |     b = SEG

=== Stratified cross-validation ===
Correctly Classified Instances       17093               71.2149 %
Incorrectly Classified Instances      6909               28.7851 %
Kappa statistic                          0.4243
Mean absolute error                      0.3706
Root mean squared error                  0.4309
Relative absolute error                 74.1111 %
Root relative squared error             86.1713 %
Total Number of Instances            24002

=== Confusion Matrix ===
     a     b   <-- classified as
  6112  5889 |     a = CONS
  1020 10981 |     b = SEG
```

Log do 23º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

TIPO = F
|   UF = MT: CONS (4897.0/1337.0)
|   UF = SP
|   |   TIPO_CONTA = I: NAO_CONS (900.0/379.0)
|   |   TIPO_CONTA = C
|   |   |   IDADE = E: CONS (248.0/117.0)
|   |   |   IDADE = C: CONS (60.0/30.0)
|   |   |   IDADE = B: CONS (6.0/3.0)
|   |   |   IDADE = D: NAO_CONS (73.0/25.0)
|   |   |   IDADE = A: NAO_CONS (0.0)
|   |   TIPO_CONTA = M: CONS (4.0/1.0)
|   |   TIPO_CONTA = S: NAO_CONS (2.0)
|   UF = SC
|   |   IDADE = E: NAO_CONS (402.0/44.0)
|   |   IDADE = C: CONS (38.0)
|   |   IDADE = B: CONS (33.0)
|   |   IDADE = D: CONS (45.0)
|   |   IDADE = A: NAO_CONS (0.0)
|   UF = RS: NAO_CONS (10619.0/3308.0)
|   UF = PR
|   |   IDADE = E: CONS (2485.0/620.0)
|   |   IDADE = C: NAO_CONS (2204.0/801.0)
|   |   IDADE = B: NAO_CONS (2120.0/837.0)
|   |   IDADE = D
|   |   |   TIPO_CONTA = I: NAO_CONS (801.0/360.0)
|   |   |   TIPO_CONTA = C: CONS (824.0/335.0)
|   |   |   TIPO_CONTA = M: CONS (0.0)
|   |   |   TIPO_CONTA = S: CONS (0.0)
```

```
|   |   IDADE = A: NAO_CONS (135.0/50.0)
TIPO = J: CONS (1894.0)

Number of Leaves  :             24
Size of the tree :      31

Time taken to build model: 0.35 seconds
Time taken to test model on training data: 0.35 seconds

=== Error on training data ===
Correctly Classified Instances       19543               70.3239 %
Incorrectly Classified Instances      8247               29.6761 %
Kappa statistic                          0.4065
Mean absolute error                      0.3968
Root mean squared error                  0.4454
Relative absolute error                 79.3635 %
Root relative squared error             89.0862 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 8091  5804 |     a = CONS
 2443 11452 |     b = NAO_CONS

=== Stratified cross-validation ===
Correctly Classified Instances       19512               70.2123 %
Incorrectly Classified Instances      8278               29.7877 %
Kappa statistic                          0.4042
Mean absolute error                      0.3972
Root mean squared error                  0.4458
Relative absolute error                 79.4435 %
Root relative squared error             89.1646 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 7996  5899 |     a = CONS
 2379 11516 |     b = NAO_CONS
```

Log do 24º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

UF = RS: SEG (9755.0/3308.0)
UF = PR
|   TEMPO_CONTA = F
|   |   ESTADO_CIVIL = C
|   |   |   SEXO = M: SEG (2676.0/1313.0)
|   |   |   SEXO = F: CONS (228.0/53.0)
|   |   ESTADO_CIVIL = S: CONS (475.0/63.0)
|   |   ESTADO_CIVIL = O: CONS (74.0/17.0)
|   |   ESTADO_CIVIL = U: CONS (67.0/29.0)
|   |   ESTADO_CIVIL = D: CONS (88.0/28.0)
|   |   ESTADO_CIVIL = V: SEG (58.0/27.0)
|   |   ESTADO_CIVIL = N: CONS (0.0)
|   TEMPO_CONTA = A: SEG (1449.0/512.0)
|   TEMPO_CONTA = C
|   |   ESTADO_CIVIL = C: SEG (736.0/326.0)
|   |   ESTADO_CIVIL = S
|   |   |   SEXO = M: SEG (280.0/137.0)
|   |   |   SEXO = F: CONS (82.0/38.0)
|   |   ESTADO_CIVIL = O
|   |   |   SEXO = M: CONS (20.0/3.0)
|   |   |   SEXO = F: SEG (7.0)
|   |   ESTADO_CIVIL = U
|   |   |   SEXO = M: SEG (41.0/9.0)
|   |   |   SEXO = F: CONS (25.0/8.0)
|   |   ESTADO_CIVIL = D: SEG (22.0/7.0)
|   |   ESTADO_CIVIL = V: SEG (20.0/2.0)
|   |   ESTADO_CIVIL = N: SEG (0.0)
|   TEMPO_CONTA = B: SEG (1511.0/597.0)
|   TEMPO_CONTA = E: SEG (1029.0/322.0)
```

```
|   TEMPO_CONTA = D: SEG (1068.0/330.0)
UF = MT: CONS (3560.0)
UF = SP: CONS (571.0)
UF = SC: CONS (160.0)

Number of Leaves  :              26
Size of the tree :      34

Time taken to build model: 0.17 seconds
Time taken to test model on training data: 0.17 seconds

=== Error on training data ===
Correctly Classified Instances       16873               70.2983 %
Incorrectly Classified Instances      7129               29.7017 %
Kappa statistic                          0.406
Mean absolute error                      0.3707
Root mean squared error                  0.4305
Relative absolute error                 74.1341 %
Root relative squared error             86.1011 %
Total Number of Instances            24002

=== Confusion Matrix ===
    a     b   <-- classified as
 5111  6890 |     a = CONS
  239 11762 |     b = SEG

=== Stratified cross-validation ===
Correctly Classified Instances       16846               70.1858 %
Incorrectly Classified Instances      7156               29.8142 %
Kappa statistic                          0.4037
Mean absolute error                      0.3711
Root mean squared error                  0.431
Relative absolute error                 74.2209 %
Root relative squared error             86.2071 %
Total Number of Instances            24002

=== Confusion Matrix ===
    a     b   <-- classified as
 5121  6880 |     a = CONS
  276 11725 |     b = SEG
```

Log do 25º experimento

```
Options: -C 0.25 -M 2
J48 pruned tree
------------------

TIPO = F
|   UF = MT: CONS (4897.0/1337.0)
|   UF = SP: NAO_CONS (1293.0/571.0)
|   UF = SC
|   |   IDADE = E: NAO_CONS (402.0/44.0)
|   |   IDADE = C: CONS (38.0)
|   |   IDADE = B: CONS (33.0)
|   |   IDADE = D: CONS (45.0)
|   |   IDADE = A: NAO_CONS (0.0)
|   UF = RS: NAO_CONS (10619.0/3308.0)
|   UF = PR
|   |   IDADE = E: CONS (2485.0/620.0)
|   |   IDADE = C: NAO_CONS (2204.0/801.0)
|   |   IDADE = B: NAO_CONS (2120.0/837.0)
|   |   IDADE = D: CONS (1625.0/776.0)
|   |   IDADE = A: NAO_CONS (135.0/50.0)
TIPO = J: CONS (1894.0)

Number of Leaves  :              14
Size of the tree :      18

Time taken to build model: 0.19 seconds
Time taken to test model on training data: 0.25 seconds

=== Error on training data ===
Correctly Classified Instances       19446               69.9748 %
Incorrectly Classified Instances      8344               30.0252 %
Kappa statistic                          0.3995
Mean absolute error                      0.3977
Root mean squared error                  0.4459
```

```
Relative absolute error               79.5416 %
Root relative squared error           89.1861 %
Total Number of Instances             27790

=== Confusion Matrix ===
    a     b   <-- classified as
 8284  5611 |     a = CONS
 2733 11162 |     b = NAO_CONS

=== Stratified cross-validation ===
Correctly Classified Instances       19446               69.9748 %
Incorrectly Classified Instances      8344               30.0252 %
Kappa statistic                          0.3995
Mean absolute error                      0.3978
Root mean squared error                  0.4461
Relative absolute error              79.5683 %
Root relative squared error          89.2171 %
Total Number of Instances            27790

=== Confusion Matrix ===
    a     b   <-- classified as
 8284  5611 |     a = CONS
 2733 11162 |     b = NAO_CONS
```

APÊNDICE B

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%		
2º	8	9	83,10%		
3º	8	13	83,02%		
4º	7	11	82,34%		
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		
9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		
18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	4

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____
_____

Colaborador: Eduardo Berno

Formação: Estatístico

Área de atuação: Analista de BI

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções.

Concordo que a acurácia é a melhor forma de ranquear.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%		
2º	8	9	83,10%		
3º	8	13	83,02%		
4º	7	11	82,34%		
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		
9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		
18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	4

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

O trabalho é extremamente útil para identificação de um possível ramo de resposta inicial, com isso indicando um possível norte para o trabalho. Além de já indicar algumas respostas mais claras.

Colaborador: Wagner Rodeski

Formação: Estatístico

Área de atuação: Engenharia de Processos

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%		
2º	8	9	83,10%		
3º	8	13	83,02%		
4º	7	11	82,34%		
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		
9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		
18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	4

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

Acredito que a mineração de dados é de extrema relevância, pois transforma dados brutos em inteligência mercadológica. A automatização do processo de mineração de dados potencializa a viabilidade de estudos via data mining em tempo real, ou muito próximo disso, o que sob a ótica do usuário final dessa ferramenta é um diferencial competitivo extremamente valioso. Outro ponto relevante é a robustez de um data mining automatizado, que permite tanto consultas para os mais variados temas quanto uma rápida e eficiente imputação de novos dados, assim como um repositório único de informações, garantindo com isso a integridade dessas informações, uma vez que diferentes fontes de dados geram um alto risco de incongruências nos dados extraídos. Por fim vale citar que a aplicabilidade deste trabalho vai desde a eficaz prospecção de novos clientes ou vendas de novos produtos, até a análise de cenários geográficos com alto potencial para a expansão de novos negócios, ou até mesmo um relacionamento mais preciso e valoroso para com os atuais clientes.

Colaborador: Gustavo Rossi

Formação: Estatístico

Área de atuação: Analista de Mercado

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%	1	
2º	8	9	83,10%	2	
3º	8	13	83,02%	3	
4º	7	11	82,34%	4	Devido a ao corte de 80% na acurácia
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		
9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		

18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	4

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

O trabalho vem resolver de uma forma bem estruturada e simplificada o processo de modelagem através do uso de ferramentas de mineração de dados. De uma forma bem interessante foi proposto uma 'escolha' do melhor modelo através da acurácia. Trata-se de uma proposta bem interessante e que vale a pena validá-la o quanto antes. Inicialmente deve-se propor um corte e testar os grupos formados. Recomendo a 'leitura' dos grupos, a fim de verificar se faz sentido o agrupamento de variáveis que se 'fundiram'. Também, a partir desta proposta de modelagem será interessante aplicar a metodologia em outras áreas, como por exemplo estudos em geomarketing. A escolha da melhor localização para a abertura de pontos comerciais é o grande desafio das redes de varejo.

Colaborador: Jonathan Gottfridsson

Formação: Administrador de Empresas

Área de atuação: Analista de Planejamento Comercial

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções. As árvores geradas não foram avaliadas, uma a uma, por mim, porém acredito sim, que a acurácia seja um bom critério para construção do ranking.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%		
2º	8	9	83,10%		
3º	8	13	83,02%		
4º	7	11	82,34%		
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		

9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		
18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	4

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

A atual estrutura de negócios do Sicredi está vivendo um momento ímpar com relação ao seu potencial futuro. A vontade de crescer é grande, e os recursos para investimentos também existem. Porém, algo que ainda o sistema carece é de uma informação acurada, que funcione como "bússola"; *driver* de negócios.

Enxergo no trabalho do colega Peterson Colares um início da construção desta "bússola". Um "ponta-pé inicial" que muito poderá nos ajudar a navegar de forma mais certeira neste "oceano do mercado financeiro". Os resultados por ele apresentados são extremamente significativos, e terão grande relevância para organização, se implantados a nível sistêmico.

Colaborador: Rafael Macedo

Formação: Administrador de Empresas

Área de atuação: Analista de BI

Questionário de avaliação de resultados

(a) Analisando as árvores geradas, como você ranquearia os experimentos realizados? Utilize a tabela abaixo para definir o ranque de acordo com suas percepções.

Ranque	Experimento	Consulta	Acurácia	Especialista de negócio	
				Ranque	Comentários
1º	8	15	83,71%		

2º	8	9	83,10%		
3º	8	13	83,02%		
4º	7	11	82,34%		
5º	4	13	77,59%		
6º	4	9	77,57%		
7º	5	1	76,78%		
8º	5	5	76,73%		
9º	4	5	76,38%		
10º	5	3	75,88%		
11º	4	1	75,77%		
12º	4	7	75,66%		
13º	4	6	75,66%		
14º	4	2	75,65%		
15º	4	3	75,65%		
16º	5	8	75,29%		
17º	5	4	74,91%		
18º	6	9	74,67%		
19º	6	13	74,53%		
20º	6	5	73,33%		
21º	6	1	72,64%		
21º	6	15	71,21%		
23º	13	5	70,21%		
24º	6	16	70,19%		
25º	13	1	69,97%		

(b) Observando que o propósito do processo apresentado nesta pesquisa é identificar oportunidades para aplicação de projetos de mineração de dados, como você classificaria o grau de relevância deste no contexto da organização onde você trabalha?

De 0 a 4, sendo 0 o menor grau de relevância e 4 o maior grau.

0	1	2	3	**4**

(c) Qual a sua percepção geral quanto aos propósitos e a aplicabilidade do trabalho realizado?

Na questão (a), relativo ao ranqueamento, fica o entendimento de que a acurácia é um excelente critério a ser utilizado. Mesmo não avaliando a árvore de decisão.

Atualmente, o processo de Data Mining além de preparar e integrar dados estruturados, pode também:

- Construir e validar modelos, utilizando-se das mais avançadas técnicas de estatística;
- Disponibilizar eficientemente o conhecimento e aplicar os modelos preditivos, para os tomadores de decisão de sua empresa e os sistemas que os apóiam.

Este processo é visto com custo as organizações, que através de uma lente míope, esperam que tal modelo complexo gere resultados no curtíssimo prazo. Estilo de empresas mecanicistas. E, que segundo Gareth Morgan (1996.), são entidades vistas através da metáfora da máquina, onde são propostas como um fim em si mesma.

Apoiando-se sobre esta visão curta, as organizações não compreendem o quão ótimo é o retorno sobre o investimento em mineração de dados, mas, no longo prazo. Apesar de, globalizado o mercado dificilmente dará retorno de imediato.

Sendo assim, o modelo proposto neste trabalho evidencia sua total aplicabilidade no mercado atual. Desta forma, as organizações poderão antecipar as necessidades dos seus mercados consumidores.

APÊNDICE C

Classe PrepareFile.java

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileFilter;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileOutputStream;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.InputStream;
import java.io.Reader;
import java.io.StringWriter;
import java.io.Writer;
import java.net.URL;
import java.sql.*;
import java.util.Properties;

import oracle.jdbc.driver.*;

class PrepareFile {

	public static void main(String args[]) {

		String dir_origem = "";

		// o arquivo encontra-se no mesmo diretório //da aplicação
		File file = new File("config.properties");
		Properties props = new Properties();
		FileInputStream fis = null;
		try {
			fis = new FileInputStream(file);
			// lê os dados que estão no arquivo
			props.load(fis);
			fis.close();
		} catch (IOException ex) {
			System.out.println(ex.getMessage());
			ex.printStackTrace();
		}

		Statement st = null;
		st = conexao(props.getProperty("url"));

		dir_origem = props.getProperty("diretorio.origem");

		File arquivos[];
		File dir = new File(dir_origem);

		FileFilter filter = new FileFilter() {
			public boolean accept(File file) {
				return file.getName().endsWith(".sql");
			}
		};

		FileFilter filter_csv = new FileFilter() {
			public boolean accept(File file) {
				return file.getName().endsWith(".csv");
			}
		};

		arquivos = dir.listFiles(filter);
		File f_orig = null;
		FileReader reader = null;
		FileWriter f_dest = null;
		BufferedReader leitor = null;
		String linha = null;
```

```
StringBuffer sql = null;

for (int i = 0; i < arquivos.length; i++) {
	// fazer a leitura de cada arquivo e executar a consulta
	f_orig = new File(arquivos[i].toString());
	try {
		f_dest = new FileWriter(arquivos[i].toString().replace(".sql",
		".csv"));
	} catch (IOException e1) {
		// TODO Auto-generated catch block
		e1.printStackTrace();
	}

	System.out.println("Gerando arquivo: " + f_orig);
	sql = new StringBuffer();

	try {
		reader = new FileReader(f_orig);
		leitor = new BufferedReader(reader);
		linha = null;

		try {
			while ((linha = leitor.readLine()) != null) {
				// System.out.println("Linha: " + linha);
				sql.append(linha).append(" ");
			}
		} catch (IOException e) {
			e.printStackTrace();
		}

		try {
			leitor.close();
			reader.close();
		} catch (IOException e) {
			e.printStackTrace();
		}

	} catch (FileNotFoundException e) {
		e.printStackTrace();
	}

	// executa consulta (arquivo sql carregado)
	ResultSet rs = null;
	try {

		rs = st.executeQuery(sql.toString());
		ResultSetMetaData rsmd = rs.getMetaData();

		int numCols = rsmd.getColumnCount();
		String[] line = new String[numCols];

		StringWriter sw = new StringWriter();

		CSVWriter writer = new CSVWriter(sw);
		try {
			// writer.writeAll(rs, true);
			if (arquivos[i].toString().contains("consulta")) {
				writer.writeColumnNames(rs);
			}

			while (rs.next()) {
				// System.out.println(rs.getArray(1));
				for (int j = 0; j < numCols; j++) {
					line[j] = rs.getString(j + 1);

				}
				writer.writeNext(line);

			}

			f_dest.write(sw.toString());

		} catch (IOException e1) {
			e1.printStackTrace();
		}
		try {
```

```
                                writer.close();
                        } catch (IOException e1) {
                                e1.printStackTrace();
                        }
                        System.out.println("Arquivo "
                                        + arquivos[i].toString().replace(".sql", ".csv")
                                        + " gerado com sucesso");

                        rs.close();
                        try {
                                f_dest.close();
                        } catch (IOException e) {
                                e.printStackTrace();
                        }

                } catch (SQLException e) {
                        e.printStackTrace();
                }

        }

        // gerando o bat para executar a conversao e experimentos
        arquivos = dir.listFiles(filter_csv);
        FileWriter f_bat = null;
        try {
                f_bat = new FileWriter(dir_origem.concat("\\gera_arff.bat"), true);

        } catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
        }

        String arff = null;
        arff = props
        .getProperty("java.path")
        .concat(props.getProperty("Xmx"))
        .concat(" -cp ")
        .concat(props.getProperty("weka.home"))
        .concat(
        " weka.core.converters.CSVLoader ARQUIVO.csv > ARQUIVO.arff");
        String exec = null;

        StringBuffer sb = new StringBuffer();
        BufferedWriter bf = new BufferedWriter(f_bat);
        for (int j = 0; j < arquivos.length; j++) {
                String temp = arff.replace("ARQUIVO", arquivos[j].toString());
                temp = temp.replace(".csv.arff", ".arff");
                temp = temp.replace(".csv.csv", ".csv");

                try {
                        bf.write(temp);
                        bf.newLine();
                } catch (IOException e) {
                        // TODO Auto-generated catch block
                        e.printStackTrace();
                }
        }
        try {
                bf.close();
        } catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
        }

}

public static Statement conexao(String url) {

        Statement stmt = null;
        try {
                Class.forName("oracle.jdbc.driver.OracleDriver");

                Connection con = DriverManager.getConnection(url, "crm", "crm");
```

```
                        con.setAutoCommit(true);

                        DatabaseMetaData dma = con.getMetaData();
                        System.out.println("\nConnected to " + dma.getURL());
                        System.out.println("Driver " + dma.getDriverName());
                        System.out.println("Version " + dma.getDriverVersion());
                        System.out.println("");
                        stmt = con.createStatement();
                } catch (SQLException ex) {
                        System.out.println("\n*** SQLException caught ***\n");
                        while (ex != null) {
                                System.out.println("SQLState: " + ex.getSQLState());
                                System.out.println("Message:  " + ex.getMessage());
                                System.out.println("Vendor:   " + ex.getErrorCode());
                                ex = ex.getNextException();
                                System.out.println("");
                        }

                } catch (java.lang.Exception ex) {
                        ex.printStackTrace();
                }

                return stmt;
        }

}
```

Classe BalanceFilter.java

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileFilter;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.util.Properties;

public class BalanceFilter {

        public static void main(String[] args) {

                BalanceFilter bf = new BalanceFilter();
                bf.BalanceFilter();
        }

        public void BalanceFilter() {
                // TODO Auto-generated method stub
                FileReader reader = null;
                FileWriter writer = null;
                BufferedReader leitor = null;
                BufferedReader leitor2 = null;
                BufferedWriter bf = null;
                String linha = null;
                File f_orig = null;
                File f_dest = null;
                boolean classe_nao_classe = false;

                File file = new File("config.properties");
                Properties props = new Properties();
                FileInputStream fis = null;
                try {
                        fis = new FileInputStream(file);
                        // lê os dados que estão no arquivo
                        props.load(fis);
                        fis.close();
                } catch (IOException ex) {
                        System.out.println(ex.getMessage());
                        ex.printStackTrace();
                }
```

```
FileFilter filter = new FileFilter() {
        public boolean accept(File file) {
                return file.getName().endsWith(".arff");
        }
};

File arquivos[];
String x = props.getProperty("diretorio.origem");
File dir = new File(x);
arquivos = dir.listFiles(filter);
FileWriter f_bat = null;
try {
        f_bat = new FileWriter(dir.toString().concat(
                        "\\filter\\experimentos.bat"), true);

} catch (IOException e) {
        // TODO Auto-generated catch block
        e.printStackTrace();
}
BufferedWriter bfw = new BufferedWriter(f_bat);

for (int i = 0; i < arquivos.length; i++) {
        f_orig = new File(arquivos[i].toString());
        f_dest = new File(arquivos[i].toString().concat("_new"));

        // System.out.println(args[0]);
        try {
                reader = new FileReader(f_orig);
                try {
                        writer = new FileWriter(f_dest);

                } catch (IOException e1) {
                        // TODO Auto-generated catch block
                        e1.printStackTrace();
                }
                leitor = new BufferedReader(reader);
                leitor2 = new BufferedReader(reader);
                bf = new BufferedWriter(writer);
                linha = null;
                String classes[] = null;
                int contagem[] = null;
                int limite = 0;

                try {

                        while ((linha = leitor.readLine()) != null) {
                                // identifica a linha com as classes
if (linha.contains("@attribute ".concat(props.getProperty("atributo.classe")))) {
        System.out.println(linha);
        String classes_tmp = (String) linha.subSequence(
                        linha.indexOf("{") + 1, linha.lastIndexOf("}"));
        System.out.println(classes);
        System.out.println(props.getProperty("classes.alvo"));
    linha = linha.replace(classes_tmp, props.getProperty("classes.alvo"));

        // classes = classes_tmp.split(",");
        classes = props.getProperty("classes.alvo").split(",");
        if (classes.length == 1) {
           System.out.println("definido balanceamento como classes e
                        nao_classe");
           classe_nao_classe = true;
           String[] classes_new = null;
           classes_new = (classes[0].toString().concat(
                   ",NAO_").concat(classes[0].toString())).split
           classes = null;
           classes = classes_new;
           linha = linha.replace(classes[0],
          classes[0].concat(",").concat("NAO_").concat(classes[0]));
        }
        contagem = new int[classes.length];
        for (int i1 = 0; i1 < contagem.length; i1++) {
             contagem[i1] = 0;
        }
```

```
                }

                // prepara novo arff
                if (linha.length() > 0) {
                        if (linha.startsWith("@")) {
                                bf.append(linha);
                                bf.newLine();
                                // writer.append(linha);
                        } else {
                                for (int i1 = 0; i1 < classes.length; i1++) {
                                        if (linha.endsWith(classes[i1])) {
                                                contagem[i1] = contagem[i1] + 1;
                                        }
                                }

                        }
                }

                }

                leitor.close();
                reader.close();
                for (int i1 = 0; i1 < contagem.length; i1++) {
                        if (limite == 0 || limite > contagem[i1]) {
                                limite = contagem[i1];
                        }
                        System.out.println(classes[i1] + " " + contagem[i1]);
                }

                if (classe_nao_classe) {
                        limite = contagem[0];
                }

                // faz a releitura para buscar as linhas dentro do limite de
                // da menor classe
                reader = new FileReader(f_orig);
                leitor2 = new BufferedReader(reader);
                // reinicia o vetor de contagens
                for (int i1 = 0; i1 < contagem.length; i1++) {
                        contagem[i1] = 0;
                }
                while ((linha = leitor2.readLine()) != null) {
                        if (linha.length() > 0) {
                                if (!linha.startsWith("@")) {
                                        if (linha.endsWith(classes[0]) && contagem[0] < limite) {
                                                contagem[0] = contagem[0] + 1;
                                                bf.append(linha);
                                                bf.newLine();
                                        } else if (classe_nao_classe &&
                                                linha.contains(classes[0]) &&
                                        contagem[1] < limite) {
                                                contagem[1] = contagem[1] + 1;
                                                String[] teste = linha.split(",");
                                                linha = linha.replace(
                                                        teste[teste.length - 1], "NAO_"
                                                        .concat(classes[0]));
                                                bf.append(linha);
                                                bf.newLine();
                                        }
                                }
                        }

                }

        } catch (IOException e) {
                e.printStackTrace();
        }

try {
        // leitor.close();
        leitor2.close();
        reader.close();
        // writer.close();
        bf.close();
} catch (IOException e) {
        e.printStackTrace();
```

```
}

} catch (FileNotFoundException e) {
		e.printStackTrace();
}

boolean ok2 = f_dest.renameTo(new File(dir.toString().concat(
			"\\filter\\").concat(f_dest.getName().replace("arff_new", "arff"))));

String exec = null;
exec = props.getProperty("java.path").concat(
props.getProperty("Xmx")).concat(" -cp ").concat(props.getProperty("weka.home"))
					.concat(" weka.classifiers.trees.J48 -C 0.25 -M 2 -t ")
					.concat(f_dest.getName().replace("arff_new", "arff"))
					.concat( -d  ")
					.concat(f_dest.getName().replace("arff_new", "model")
					.concat(" >  ")
					.concat(f_dest.getName().replace("arff_new", "log")));

					try {
						bfw.write(exec);
						bfw.newLine();
					} catch (IOException e) {
						// TODO Auto-generated catch block
						e.printStackTrace();
					}

			}

			try {

					bfw.close();
			} catch (IOException e) {
					// TODO Auto-generated catch block
					e.printStackTrace();
			}

		}

		private String get_classe_fake(String a, String b) {
			String result = a;
			String temp[] = b.split(",");
			for (int i = 0; i < temp.length; i++) {
				if (!a.equals(temp[i].toString())) {
					result = result.concat(",").concat(temp[i]);
					break;
				}
			}

			return result;
		}

}
```

Classe GetLog.java

```
import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.File;
import java.io.FileFilter;
import java.io.FileInputStream;
import java.io.FileNotFoundException;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.sql.SQLException;
import java.sql.Statement;
import java.util.ArrayList;
import java.util.Properties;

public class GetLog {

		public static void main(String[] args) {
```

```
		FileReader reader = null;
		BufferedReader leitor = null;
		File f_orig = null;
		String linha = null;
		boolean achei = false;

		// variaveis para armazenamento
		String Correctly_Classified_Instances = null;
		String Accuracy = null;
		String Incorrectly_Classified_Instances = null;
		String Kappa_statistic = null;
		String Mean_absolute_error = null;
		String Root_mean_squared_error = null;
		String Relative_absolute_error = null;
		String Root_relative_squared_error = null;
		String Total_Number_of_Instances = null;
		String Number_of_Leaves = null;
		String Size_of_the_tree = null;

		File file = new File("config.properties");
		Properties props = new Properties();
		FileInputStream fis = null;
		try {
			fis = new FileInputStream(file);
			// lê os dados que estão no arquivo
			props.load(fis);
			fis.close();
		} catch (IOException ex) {
			System.out.println(ex.getMessage());
			ex.printStackTrace();
		}

		Statement st = null;
		st = PrepareFile.conexao(props.getProperty("url"));

		FileFilter filter = new FileFilter() {
			public boolean accept(File file) {
				return file.getName().endsWith(".log");
			}
		};

		File arquivos[];

		String experimento = props.getProperty("experimento.nome");
		String x = props.getProperty("diretorio.origem").concat("\\").concat(experimento).concat("_filter");


		File dir = new File(x);
		arquivos = dir.listFiles(filter);
		boolean inserir = false;
		for (int i = 0; i < arquivos.length; i++) {
			try {
				f_orig = new File(arquivos[i].toString());
				reader = new FileReader(f_orig);

				leitor = new BufferedReader(reader);
				achei = false;
				inserir = false;

while ((linha = leitor.readLine()) != null) {
		if (linha.contains("Number of Leaves")) {
			String[] temp = linha.split(" ");
			ArrayList<String> temp2 = new ArrayList();
			for (int j = 0; j < temp.length; j++) {
				if (temp[j].length() > 0) {
					temp2.add(temp[j]);
				}
			}
			Number_of_Leaves = temp2.get(4);
		}
			if (linha.contains("Size of the tree")) {
			String[] temp = linha.split(" ");
			ArrayList<String> temp2 = new ArrayList();
			for (int j = 0; j < temp.length; j++) {
				if (temp[j].length() > 0) {
```

```
                                        temp2.add(temp[j]);
                                }
                        }
                        Size_of_the_tree = temp2.get(5);
                }
                        if (linha.contains("Stratified cross-validation")) {
                        System.out.println(linha);
                        achei = true;
                }
                        if (achei) {
                        if (linha.length() > 0) {
                                System.out.println(linha);

if (linha.contains("Correctly Classified Instances")) {
                inserir = true;
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
                for (int j = 0; j < temp.length; j++) {
                        if (temp[j].length() > 0) {
                                temp2.add(temp[j]);
                        }
                }
                Correctly_Classified_Instances = temp2.get(3);
                Accuracy = temp2.get(4);
}


if (linha.contains("Incorrectly Classified Instances")) {
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
                for (int j = 0; j < temp.length; j++) {
                        if (temp[j].length() > 0) {
                                temp2.add(temp[j]);
                        }
                }
                Incorrectly_Classified_Instances = temp2.get(3);
}
if (linha.contains("Kappa statistic")) {
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
                for (int j = 0; j < temp.length; j++) {
                        if (temp[j].length() > 0) {
                                temp2.add(temp[j]);
                        }
                }
                Kappa_statistic = temp2.get(2);
}

if (linha.contains("Mean absolute error")) {
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
                for (int j = 0; j < temp.length; j++) {
                        if (temp[j].length() > 0) {
                                temp2.add(temp[j]);
                        }
                }
                Mean_absolute_error = temp2.get(3);
}

if (linha.contains("Root mean squared error")) {
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
                for (int j = 0; j < temp.length; j++) {
                        if (temp[j].length() > 0) {
                                temp2.add(temp[j]);
                        }
                }
                Root_mean_squared_error = temp2.get(4);
}

if (linha.contains("Relative absolute error")) {
                String[] temp = linha.split(" ");
                ArrayList<String> temp2 = new ArrayList();
```

```
                        for (int j = 0; j < temp.length; j++) {
                                if (temp[j].length() > 0) {
                                        temp2.add(temp[j]);
                                }
                        }
                        Relative_absolute_error = temp2.get(3);
                }

                if (linha.contains("Root relative squared error")) {
                        String[] temp = linha.split(" ");
                        ArrayList<String> temp2 = new ArrayList();
                        for (int j = 0; j < temp.length; j++) {
                                if (temp[j].length() > 0) {
                                        temp2.add(temp[j]);
                                }
                        }
                        Root_relative_squared_error = temp2.get(4);
                }

                if (linha.contains("Total Number of Instances")) {
                        String[] temp = linha.split(" ");
                        ArrayList<String> temp2 = new ArrayList();
                        for (int j = 0; j < temp.length; j++) {
                                if (temp[j].length() > 0) {
                                        temp2.add(temp[j]);
                                }
                        }
                        Total_Number_of_Instances = temp2.get(4);
                }
        }

    }
 }

// verifica se tem dados para inserir
if (inserir) {
                // insere dados na tabela de logs
                try {
                        String sql = "insert into crm_experimento_log (EXPERIMENTO, ARQUIVO, ALGORITMO,
CORRECTLY_CLASSIFIED_INSTANCES,"+
"ACCURACY,INCORRECTLY_CLASS_INSTANCES,KAPPA_STATISTIC,MEAN_ABSOLUTE_ERROR,"+
"ROOT_MEAN_SQUARED_ERROR,RELATIVE_ABSOLUTE_ERROR,ROOT_RELATIVE_SQUARED_ERROR,"
                                + "TOTAL_NUMBER_OF_INSTANCES,NUMBER_OF_LEAVES,SIZE_OF_THE_TREE) "
+ "values ('"+ experimento+ "','"                 + arquivos[i].toString()+ "','J48','"+ "'"
+ Correctly_Classified_Instances+ "','"+ Accuracy+ "','"+ Incorrectly_Classified_Instances
+ "'"+ "','"+ Kappa_statistic+ "','"                 + Mean_absolute_error+ "','"+ Root_mean_squared_error
+ "','"+ Relative_absolute_error+ "'"+ "','" + Root_relative_squared_error+ "','"
+ Total_Number_of_Instances+ "','"+ Number_of_Leaves+ "','"+ Size_of_the_tree+ "')";

                        st.executeUpdate(sql);
                } catch (SQLException e) {
                // TODO Auto-generated catch block
                        e.printStackTrace();
                }

                // limpa as variaveis
                Correctly_Classified_Instances = null;
                Accuracy = null;
                Incorrectly_Classified_Instances = null;
                Kappa_statistic = null;
                Mean_absolute_error = null;
                Root_mean_squared_error = null;
                Relative_absolute_error = null;
                Root_relative_squared_error = null;
                Total_Number_of_Instances = null;
                Number_of_Leaves = null;
                Size_of_the_tree = null;

                }
} catch (FileNotFoundException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
} catch (IOException e) {
                // TODO Auto-generated catch block
                e.printStackTrace();
}
```

```
}
try {
		st.close();
} catch (SQLException e) {
		// TODO Auto-generated catch block
		e.printStackTrace();
}
}
}
```

Arquivo config.properties

```
#Configurações
weka.home=C:\\softwares\\Weka-3-6\\weka.jar
url =jdbc:oracle:thin:@peterson-pc:1521:dblocal
Xmx = -D64 -Xmx2000m
java.path =C:\\java\\64\\jdk1.6.0_21\\bin\\java
weka.algoritmo = weka.classifiers.trees.J48 -C 0.25 -M 2 -t

#geral
diretorio.origem =C:\\_experimentos\\2
experimento.nome = EXPERIMENTO_16

#Parametrizações do Balancer
#definicao do atributo classe do arquivo original
atributo.classe=AREA

#definicao do conjunto de classes desejadas no arquivo de saída
#importante: a) quando nenhuma classe for informada, o algoritmo deverá balancear todas
#               as classes encontradas no arquivo original
#            b) quando for informada somente uma classe, deve-se gerar "classe" e
"não_classe"
#               como o conjunto de classes no arquivo de saída
#classes.alvo=CRED,INVEST
#classes.alvo=CART,CONS,SEG,CRED,INVEST
classes.alvo=INVEST
```

Arquivo balancer.properties

```
#Parametrizações do Balancer
#definicao do atributo classe do arquivo original
atributo.classe=AREA
diretorio.origem =C:\\TEMP\\experimentos5

#definicao do conjunto de classes desejadas no arquivo de saída
#importante: a) quando nenhuma classe for informada, o algoritmo deverá balancear todas
as classes encontradas no arquivo original
#            b) quando for informada somente uma classe, deve-se gerar "classe" e
"não_classe" como o conjunto de classes no arquivo de saída
classes.alvo=CART,CONS,SEG
#classes.alvo=CART,CONS,SEG,CRED,INVEST
```