MINERAÇÃO DE DADOS PARA DETECÇÃO DE FRAUDES EM TRANSAÇÕES ELETRÔNICAS

JOSÉ FELIPE JÚNIOR

MINERAÇÃO DE DADOS PARA DETECÇÃO DE FRAUDES EM TRANSAÇÕES ELETRÔNICAS

Dissertação apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Mestre em Ciência da Computação.

Orientador: Wagner Meira Júnior
Coorientador: Adriano César Machado Pereira

Belo Horizonte
Março de 2012

Felipe Júnior, José

F314m Mineração de Dados para Detecção de Fraudes em Transações Eletrônicas / José Felipe Júnior. — Belo Horizonte, 2012
xxiv, 110 f. : il. ; 29cm

Dissertação (mestrado) — Universidade Federal de Minas Gerais
Orientador: Wagner Meira Júnior
Coorientador: Adriano César Machado Pereira

1. Computação - Teses. 2. Mineração de Dados - Teses. 3. Comércio Eletrônico - Teses. I. Orientador. II. Coorientador. III. Título.

CDU 519.6*73(043)

UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

Mineração de dados para detecção de fraudes em transações eletrônicas

JOSÉ FELIPE JÚNIOR

Dissertação defendida e aprovada pela banca examinadora constituída pelos Senhores:

PROF. WAGNER MEIRA JÚNIOR - Orientador
Departamento de Ciência da Computação - UFMG

PROF. ADRIANO CÉSAR MACHADO PEREIRA
Departamento de Computação - CEFET

PROF. ADRIANO ALONSO VELOSO
Departamento de Ciência da Computação - UFMG

PROF. LEONARDO BARBOSA E OLIVEIRA
Departamento de Ciência da Computação - UFMG

SR. MARDEN SILVEIRA NEUBERT
Diretor de Pesquisa e Desenvolvimento - Universo Online S.A

Belo Horizonte, 30 de março de 2012.

Dedico esse trabalho à minha mãe Norma e minha tia Maria que sempre me apoiaram e que sempre pude contar durante todas as fases da minha vida.

Agradecimentos

Agradeço primeiramente à minha mãe Norma e a minha tia Maria que sempre me apoiaram e auxiliaram durante toda a minha vida. Agradeço aos meus orientadores, Adriano Pereira e Wagner Meira pelos ensinamentos, pela paciência e pela confiança no meu trabalho. Agradeço ao professor Adriano Veloso pelas significativas contribuições ao longo desta pesquisa. Agradeço ao UOL PagSeguro pela disponibilização dos dados e por todas as consultorias prestadas para a realização deste trabalho. Agradeço à Siemens pelo apoio, o que foi essencial para a realização deste mestrado. Agradeço aos amigos: Fabrício Costa, Gilmara Teixeira, Sinaide Bezerra, Glauber Dias, Diego Marinho, Cristiano Santos, Kênia Carolina, Rafael Amaral, Carlos Eduardo, Evandro Caldeira, Rafael Lima e Gabriel Bandão que contribuíram muito para que eu pudesse vencer mais esta etapa.

"Às vezes é preciso que nossa
fé seja recompensada."
(Autor desconhecido)

Resumo

Com a popularização da Internet, cresce cada vez mais o número de pessoas que utilizam esse meio para realizar transações financeiras. Isso se deve às facilidades promovidas pela Web para realização de compras e pagamentos a qualquer momento e em qualquer lugar. No entanto, essa popularização tem atraído a atenção de criminosos, resultando em um número significativo e crescente de casos de fraude em transações eletrônicas feitas na Internet. As perdas financeiras chegam à ordem de bilhões de dólares por ano. Nesse contexto, a detecção de fraude consiste em diferenciar as transações fraudulentas daquelas que são legítimas. Entretanto, este é um desafio técnico devido à diversidade de estratégias utilizadas pelos fraudadores, o número relativamente pequeno de fraudes em relação ao número total de transações e a constante evolução das práticas de mercado e tecnologia associada. Este trabalho propõe uma metodologia para detecção de fraude em pagamentos online baseada no processo de Descoberta do Conhecimento em Banco de Dados. Ele também fornece uma pesquisa abrangente sobre a detecção de fraudes com foco em questões que vão desde a base de dados, passando pela avaliação de técnicas mais promissoras, até questões relacionadas ao retorno financeiro obtido com as técnicas. A metodologia proposta foi aplicada em um conjunto de dados real de uma das maiores empresas no Brasil de serviços de pagamentos eletrônicos, o *UOL PagSeguro*. Os resultados obtidos demonstram a eficácia dessa metodologia na mineração de dados relevantes para a detecção de fraudes. Além disso, ela fornece técnicas e orientações que permitem verificar, com antecedência, se os dados extraídos serão relevantes para a identificação de fraude. Foi definido um conceito de eficiência econômica na captura de fraudes e diversas técnicas foram avaliadas aplicando esse conceito. Os resultados mostram um bom desempenho na detecção de fraudes, apresentando ganhos significativos ao cenário atual da empresa.

Palavras-chave: Mineração de Dados, Detecção de Fraudes, Comércio Eletrônico, Aplicação Web, KDD.

Abstract

The growing popularity of the Internet also enabled an increasing number of financial transactions through the Web, due to the convenience of making purchases and payments at any time and anywhere. On the other hand, such popularity also attracted criminals, resulting in a significant and increasing number of fraud cases in Internet-based electronic transactions, which lead to losses in the order of billions of dollars per year. In this context, the fraud detection consists in distinguish fraudulent transactions from those that are legitimate. Further, it is challenging technically because of the diversity of strategies used by the fraudsters, the relatively small number of frauds compared to the overall number of transactions and the constant evolution of the market practices and associated technology. This work proposes a methodology for fraud detection in online payments based on the knowledge discovery process. It also provides a comprehensive survey on fraud detection, focusing on issues that range from the data storage to the financial return to the techniques, through the effectiveness of the techniques. The proposed methodology was applied to a real data set from one of the largest of electronic payment services in Brazil, *UOL PagSeguro* and the results demonstrate its effectiveness in mining relevant data for fraud detection. Moreover, it provides guidelines and techniques for identifying, in advance, whether extracted data is relevant or not for the fraud detection process. We also proposed a concept of economic efficiency in fraud detection, and evaluated the various detection techniques under this concept. The results show a good performance in detecting frauds, showing significant gains in the current scenario of *UOL PagSeguro*.

Keywords: Data Mining, Fraud Detection, E-Commerce, Web Application, KDD.

Lista de Figuras

2.1 Sistema de autorização de uma transação. Fonte:[Gadi, 2008]. 9
2.2 Processo de gestão de risco. Fonte: [Mindware Research Group, 2011]. . . 17
2.3 Aceitação dos resultados da triagem automática. Fonte: [Mindware Research Group, 2011]. 18
2.4 Ferramentas adotadas/planejadas para análise de pedidos. Fonte: [Mindware Research Group, 2011]. 19
2.5 Ferramentas adotadas/planejadas para análise manual de pedidos. Fonte: [Mindware Research Group, 2011]. 23
2.6 Ferramentas mais efetivas no combate a fraude. Fonte: [Mindware Research Group, 2011]. 24
2.7 Taxas de aceitação e rejeição de pedidos. Fonte: [Mindware Research Group, 2011]. 25

4.1 Visão geral das etapas que compõem o processo *KDD*. Fonte: [Fayyad et al., 1996b]. 38
4.2 Esforço relativo gasto em etapas específicas do processo *KDD*. Fonte: [Cios et al., 2007]. 39
4.3 Processo de seleção dos dados. 41
4.4 Processo de auditoria na construção do *dataset*. 43
4.5 Uma árvore de decisão para o problema de classificação de vertebrados. Fonte: [Tan et al., 2009]. 53
4.6 Modelo de Stacking. Fonte:[Maria Izabela R. Caffé, 2011]. 61
4.7 Eficiência econômica ao longo do *ranking*. A região em cinza indica que ganhos são obtidos em relação aos ganhos reais. 66
4.8 Metodologia para avaliação dos classificadores. 66
4.9 Geração do *ranking* a partir dos resultados de um classificador 67

5.1 Página inicial do site do *UOL PagSeguro*. 70

5.2 Processo de seleção dos dados. 73
5.3 Dispersão dos dados em relação à unidade federativa. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa a unidade federativa. . . . 78
5.4 Dispersão dos dados em relação ao tempo de registro do comprador. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o tempo de registro do comprador. 79
5.5 Dispersão dos dados em relação aos dias da semana. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa os dias da semana. 80
5.6 Dispersão dos dados em relação às horas do dia. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa as horas do dia. 81
5.7 Dispersão dos dados em relação ao tempo de registro do vendedor. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o tempo de registro do vendedor. 82
5.8 Dispersão dos dados em relação a categoria de tipo de produto vendido pelo vendedor. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa a categoria de tipo de produto vendido pelo vendedor. 83
5.9 Dispersão dos dados em relação ao número de parcelas. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o número de parcelas na compra do produto. 85
5.10 Distribuição dos registros em relação ao dia da semana. 85
5.11 Distribuição dos registros em relação às horas do dia. 86
5.12 Distribuição dos registros em relação a idade do comprador. 86
5.13 Distribuição dos registros em relação a bandeira de cartão de crédito. . . . 87
5.14 Distribuição dos registros em relação aos *chargebacks*. 87
5.15 *Precisão X Revocação* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores. . 91
5.16 *Eficiência Econômica* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores. . 91

5.17 *Ganho em relação ao máximo* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores. 92
5.18 *Precisão X Revocação* dos resultados obtidos em dezembro com o *Stacking*. 94
5.19 *Eficiência Econômica* dos resultados obtidos em dezembro com o *Stacking*. 94
5.20 *Ganho em relação ao máximo* dos resultados obtidos em dezembro com o *Stacking*. 95
5.21 *Precisão X Revocação* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*. 95
5.22 *Eficiência Econômica* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*. 99
5.23 *Ganho em relação ao máximo* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*. 99

Lista de Tabelas

4.1 Matriz de confusão para um problema de classificação binária na qual as classes não são igualmente importantes 63

5.1 Lista de atributos selecionados 84
5.2 Resultado em percentual das técnicas selecionadas sem alteração na distribuição e sem combinação de classificadores. 90
5.3 Resultado em percentual das técnicas de meta aprendizagem. 93
5.4 Resultado em percentual do *oversampling* para a técnica *J48*. 96
5.5 Resultado em percentual do *oversampling* para a técnica *Naive Bayes*. . . 97
5.6 Resultado em percentual do *oversampling* para a técnica *LAC*. 98

Sumário

Agradecimentos **ix**

Resumo **xiii**

Abstract **xv**

Lista de Figuras **xvii**

Lista de Tabelas **xxi**

1 Introdução **1**
- 1.1 Objetivo . . . 3
- 1.2 Principais Contribuições . . . 4
- 1.3 Organização da Dissertação . . . 5

2 Fraudes em Transações Eletrônicas **7**
- 2.1 Funcionamento do Cartão de Crédito . . . 7
- 2.2 Fraudes de Cartão de Crédito . . . 9
  - 2.2.1 Como as Fraudes são Cometidas no Mundo . . . 10
  - 2.2.2 Impacto das Fraudes de Cartão de Crédito . . . 14
- 2.3 Prevenção e Gestão de Risco . . . 16
  - 2.3.1 Etapa 1: Triagem Automática . . . 17
  - 2.3.2 Etapa 2: Revisão Manual . . . 22
  - 2.3.3 Etapa 3: Situação do Pedido (Aceitar/Rejeitar) . . . 24
  - 2.3.4 Etapa 4: Gerenciamento de Disputas . . . 25
- 2.4 Considerações Finais . . . 26

3 Trabalhos Relacionados **27**
- 3.1 Métodos para Detecção de Fraude . . . 27
- 3.2 Técnicas de Mineração de Dados . . . 28

3.3 Cenários de Aplicação 31
3.4 Considerações Finais 32

4 Metodologia para Detecção de Fraude 35
4.1 Processo de Descoberta de Conhecimento em Banco de Dados 35
4.2 Passo 1: Seleção dos Dados 39
4.3 Passo 2: Pré-Processamento 42
4.4 Passo 3: Transformação 45
4.5 Passo 4: Mineração de Dados 49
4.5.1 Árvore de Decisão 53
4.5.2 Classificador Baseado em Regras 55
4.5.3 Naive Bayes 58
4.5.4 Meta Aprendizagem 59
4.5.5 Oversampling 60
4.6 Passo 5: Interpretação e Avaliação dos Resultados 61
4.7 Considerações Finais 66

5 Estudo de Caso - UOL PagSeguro 69
5.1 Visão Geral 69
5.2 Seleção dos Dados 71
5.3 Pré-Processamento 75
5.4 Transformação 80
5.5 Mineração de Dados 85
5.6 Interpretação e Avaliação dos Resultados 89
5.7 Considerações Finais 100

6 Conclusão e Trabalhos Futuros 101

Referências Bibliográficas 105

Capítulo 1

Introdução

Durante as últimas décadas tem-se observado um aumento significativo no volume de transações eletrônicas, principalmente devido à popularização do *e-commerce*. Essa popularidade, associada ao grande volume financeiro envolvido e ao tráfego de informações sigilosas, como por exemplo, CPF e número do cartão, tem atraído a atenção de criminosos com o objetivo de obter vantagens financeiras. Segundo Bhatla et al. [2003], a taxa em que a fraude ocorre na Internet é de 12 a 15 vezes maior do que a fraude no "mundo físico". Isso porque para as vendas em sites de comércio eletrônico não existem as vantagens das verificações físicas tais como verificação de assinatura, identificação com foto, confirmação visual das informações do cartão, dentre outras. Sendo assim, as vendas pela *Web* podem representar uma grande ameaça para os comerciantes.

De acordo com Mindware Research Group [2011], nas vendas *online* da América do Norte estima-se que a perda de receita total em 2011 foi de aproximadamente US$ 3,4 bilhões, correspondendo a um aumento de US$ 700 milhões em relação aos resultados de 2010. Segundo a Federação Brasileira de Bancos (FEBRABAN)[1], as perdas com fraudes bancárias realizadas por meio eletrônico somaram R$ 685 milhões no primeiro semestre de 2011, o que representa um aumento de 36% em relação ao mesmo período de 2010.

Como é possível perceber, as fraudes no comércio eletrônico têm aumentado drasticamente e representam perdas significativas para os negócios. Dessa forma, a prevenção e a detecção de fraude têm se mostrado essenciais. De acordo com Brause et al. [1999], dado o elevado número de transações realizadas a cada dia, uma redução de 2,5% em atos fraudulentos irá proporcionar uma economia de um milhão de dólares por ano.

[1] http://www.febraban.org.br

O combate à fraude envolve um elevado número de problemas a serem enfrentados e um deles é o grande volume de dados associados. As vendas feitas pela Internet, tanto no Brasil quanto no mundo envolvem milhões de transações por dia. No Brasil, segundo Mindware Research Group [2011], o número de compradores *online* foi de 23 milhões em 2010 e isso representou um crescimento de 35% em relação ao ano anterior. O volume de vendas nesse período foi de R$ 14,3 bilhões, sendo que 64% dessas vendas foram feitas com cartão de crédito. A empresa *Barclaycard* apresenta cerca de 350 milhões de transações por ano só no Reino Unido. O *Royal Bank of Scotland*, que tem o maior mercado de cartão de crédito na Europa, possui mais de um bilhão de transações por ano. Esse grande volume de informação torna inviável a análise manual de cada uma das transações com o objetivo de decidir, de forma rápida, se ela é ou não fraudulenta. Além do mais, este claramente não é um problema de classificação fácil de resolver, já que além do grande volume de dados envolvidos, as transações de fraude não ocorrem com frequência. Sendo assim, há uma necessidade de teorias computacionais e ferramentas para ajudar os seres humanos nessa tarefa não trivial de classificação.

A mineração de dados tem apresentado grandes contribuições nessa área. De acordo com Tan et al. [2009], a mineração de dados consiste em um conjunto de técnicas organizadas para analisar grandes bancos de dados com o intuito de descobrir padrões úteis e recentes que poderiam, de outra forma, permanecer ignorados. Elas também permitem prever o resultado de uma observação futura, como por exemplo, a previsão se um determinado cliente possui uma pré-disposição a atos fraudulentos.

Para que seja possível a utilização das técnicas de mineração de dados, é necessário que os dados estejam em um formato adequado. No entanto, as grandes bases de dados são compostas por centenas de tabelas em formatos que, geralmente não são convenientes para a aplicação das técnicas. Sendo assim, são necessárias metodologias que auxiliem na seleção, limpeza e preparação dos dados para que seja possível a utilização da mineração de dados.

O processo de Descoberta de Conhecimento em Banco de Dados (KDD) preocupa-se com o desenvolvimento de métodos e técnicas para extrair conhecimento a partir das informações existentes nos dados. O problema básico tratado por esse processo é um mapeamento dos dados (que são tipicamente volumosos demais para uma fácil compreensão) em outras formas que possam ser mais compactas (por exemplo, um relatório curto) e mais úteis (por exemplo, um modelo preditivo para estimar o valor de futuros casos). No cerne do processo está a aplicação de determinadas técnicas de mineração de dados para descoberta de padrões e extração de conhecimento.

Esta pesquisa propõe uma metodologia para detecção de fraudes com o objetivo

de criar modelos de classificação que auxiliem na identificação de registros fraudulentos. A metodologia é composta de 5 etapas *(Seleção dos Dados, Pré-Processamento, Transformação, Mineração de Dados e Interpretação e Avaliação dos Resultados)* e tem como base o Processo de Descoberta do Conhecimento em Banco de Dados com inclusões, modificações e orientações, descritas a seguir, que visam auxiliar os usuários a avaliar cenários de fraude na *Web* ou outros que apresentem características semelhantes. No passo de *Seleção de Dados*, é proposto um método para auxiliar na extração de informações de um banco de dados, tornando mais fácil essa tarefa. É apresentado um método de auditoria para garantir a integridade e a coerência dos dados extraídos. No *Pré-Processamento*, algumas manipulações nos dados e testes são empregados para auxiliar na identificação de padrões de fraude e não fraude. Na *Transformação*, a metodologia ajuda a encontrar algumas evidências de dados que possam identificar os melhores atributos para mineração de dados. Na etapa de *Mineração de Dados*, técnicas mais promissoras para detecção e entendimento da fraude são apresentadas. Finalmente, o último passo proporciona um método para avaliar a fraude e o seu impacto nas transações eletrônicas. Para verificar a eficácia da metodologia foi utilizado um conjunto de dados real de uma das maiores empresas no Brasil de serviços de pagamentos eletrônicos, o *UOL PagSeguro*.

Uma consideração importante a ser feita é que a detecção de fraude está associada a duas vertentes. Essas vertentes consistem na identificação de padrões intra ou inter-transações. A identificação de padrões intra-transações tem como objetivo classificar uma transação como fraudulenta ou não, baseada em seus atributos. O padrão inter-transações consiste em analisar uma sequência de transações e a partir disso identificar a existência de fraude. Esta pesquisa trabalha com a identificação de fraude na busca de padrões intra-transações, ou seja, dado como entrada uma transação com um conjunto de atributos, um modelo deverá classificar essa transação como fraudulenta ou não. Grande parte desse foco se deve às características dos dados obtidos, além do mais, as duas vertentes existentes podem envolver estudos diferentes, o que ampliaria ainda mais o universo de pesquisa.

1.1 Objetivo

O objetivo desta pesquisa é desenvolver uma metodologia que possa auxiliar de forma eficaz o processo de detecção de fraude. Essa metodologia deverá abordar questões que vão desde a criação de um *dataset*, passando pela identificação das técnicas mais promissoras, até a avaliação dos resultados alcançados, considerando também o

retorno financeiro obtido com as técnicas.

Pode-se enumerar os seguintes objetivos específicos:

1. Criar um *dataset* com atributos significativos para detecção de fraude;
2. Identificar, avaliar e aperfeiçoar ou propor algoritmos de mineração de dados para detecção de fraudes;
3. Aplicar a metodologia e algoritmos a cenários reais. Nesse caso a base de dados selecionada será utilizada para avaliar a efetividade dos algoritmos.

1.2 Principais Contribuições

Durante a realização desta pesquisa, as seguintes contribuições podem ser listadas:

1. Uma pesquisa abrangente no cenário de fraudes em pagamentos *online* onde são apresentadas informações significativas para a detecção de fraude.
2. Criação de uma metodologia para a detecção de fraude.
3. Criação de um método para extração de dados em bancos de dados relacionais e um procedimento de auditoria para minimizar a ocorrência de erros durante a extração.
4. Criação de um método para avaliação dos resultados obtidos, onde os mesmos podem ser maximizados com base na utilização de um *ranking* das probabilidades de fraude.
5. Criação de uma medida de avaliação da eficiência econômica obtida pela técnica de mineração de dados.
6. Artigo intitulado *Fraud Detection in Electronic Transactions* publicado em *Iadis International Conference WWW/Internet*, realizada em 2011 no Rio de Janeiro. Nesse artigo são apresentados os resultados preliminares deste trabalho.
7. Artigo intitulado *Slice and Aggregate: New Evolutionary Approaches to High-Dimensional Data* publicado como resumo estendido (poster) em *GECCO'12 - Genetic and Evolutionary Computation Conference*, realizada em 2012 na Filadélfia, EUA. Nesse artigo é apresentado um método utilizando algoritmos genéticos para lidar com dados de alta dimensionalidade.

8. Artigo intitulado *Methodology for Fraud Detection in Electronic Transaction* publicado como resumo estendido no *18th Brazilian Symposium on Multimedia and the Web - WebMedia'12* realizado em 2012 em São Paulo - SP. Nesse artigo são apresentados alguns resultados deste trabalho.

9. Artigo intitulado Mineração de Dados para Detecção de Fraudes em Transações Eletrônicas publicado no XII Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSEG) - Concurso de Teses e Dissertações (CTD-Seg) realizado em 2012 em Curitiba - PR. Nesse Simpósito este trabalho foi finalista do Concurso de Teses e Dissertações.

10. Artigo intitulado *A KDD-Based Methodology to Rank Trust in e-Commerce Systems* publicado em *The 2013 IEEE/WIC/ACM International Conference on Web Intelligence (WI13)* realizada em Atlanta, EUA. Nesse artigo são apresentados os resultados deste trabalho.

1.3 Organização da Dissertação

O restante desta dissertação é organizado da seguinte forma: no Capítulo 2 é realizada uma ampla discussão sobre as fraudes de cartão de crédito em compras *online*. São abordados assuntos como o funcionamento do cartão, as fraudes associadas e seus impactos e apresentado o processo de gestão de risco utilizado no comércio eletrônico. No Capítulo 3 são apresentados os trabalhos relacionados, onde são apresentadas metodologias e técnicas de mineração de dados utilizadas no combate à fraude. No Capítulo 4 é apresentada a metodologia para detecção de fraude baseada no processo de Descoberta de Conhecimento em Banco de Dados. Essa metodologia é composta por cinco passos com o objetivo de auxiliar o processo de detecção de fraude desde a obtenção dos dados para a formação de um *dataset*, passando pela escolha de técnicas mais promissoras, até a avaliação dos resultados obtidos. No Capítulo 5 é feito um estudo de caso onde a metodologia proposta é instanciada para a realização do processo de detecção de fraude em um conjunto de dados real disponibilizado pelo *UOL PagSeguro*. Finalmente, no Capítulo 6 são apresentados as conclusões e trabalhos futuros.

Capítulo 2

Fraudes em Transações Eletrônicas

As fraudes em pagamentos *online* com cartão de crédito causam, a cada ano, prejuízos financeiros da ordem de bilhões de dólares, o que ressalta a necessidade da criação de meios eficazes para combatê-las. Para que isso seja possível, é necessário o entendimento de como essas fraudes ocorrem e as formas de prevenção utilizadas atualmente pelos comerciantes *online*. Este capítulo busca contextualizar o leitor com relação ao funcionamento dos cartões de crédito, assim como, as fraudes associadas a esse meio de pagamento. O foco desta pesquisa são as fraudes associadas a vendas *online* com cartão de crédito, dessa forma é feita uma análise detalhada dos meios de prevenção e é descrito o processo de gestão de fraude adotado pelos comerciantes da *Web*.

2.1 Funcionamento do Cartão de Crédito

O cartão de crédito é utilizado como meio de pagamento para realização de compras ou contração de serviços. O titular irá receber mensalmente no seu endereço a fatura para pagamento. Ele poderá optar por pagar o total cobrado, somente o valor mínimo ou algum valor intermediário, adiando o pagamento do restante da fatura para o mês seguinte onde serão cobrados juros. O banco emissor define um limite de crédito para as compras. Cada compra efetuada reduz o limite disponível até que, quando insuficiente, a tentativa de novas compras será negada. Cada pagamento da fatura libera o limite para ser usado novamente.

As operações com cartões de crédito possuem cinco entidades envolvidas no seu funcionamento, são elas: Portador *(Card Holder)*, Estabelecimento *(Merchant)*, Adquirente *(Acquirer)*, Bandeira *(Brand)* e Emissor *(Issuer)*. Cada uma dessas entidades são detalhadas a seguir:

1. Portador *(Card Holder)*: Pessoa que tem como objetivo adquirir bens ou contratar serviços realizando o pagamento por meio do cartão de crédito. Essa pessoa pode ser o titular da conta ou apenas portador de um cartão adicional.

2. Estabelecimento *(Merchant)*: Empresa cujo interesse é vender ou prestar serviços e receber o pagamento dos seus clientes por meio do cartão de crédito. Aqui é representado pelos sites de pagamentos *online*.

3. Adquirente *(Acquirer)*: Empresa que tem como responsabilidade a comunicação da transação entre o estabelecimento e a bandeira. Essas empresas alugam e mantém os equipamentos usados pelos estabelecimentos como, por exemplo, o *Point of Sales(POS)*. As maiores adquirentes no Brasil são Redecard, Cielo (antiga Visanet Brasil), Hipercard e Getnet.

4. Bandeira *(Brand)*: Empresa que tem como responsabilidade a comunicação da transação entre o adquirente e o emissor do cartão de crédito. As maiores bandeiras no Brasil são Visa, MasterCard e Hipercard.

5. Emissor *(Issuer)*: É a Instituição financeira, tipicamente um banco, que emite o cartão de crédito, define limite de compras, decide se as transações são aprovadas, emite fatura para pagamento, cobra os titulares em caso de inadimplência e oferece produtos atrelados ao cartão como seguro, cartões adicionais e plano de recompensas. Ela também é chamada de empresa administradora do cartão.

Todas essas entidades irão interagir por meio do processo de autorização de uma transação que ocorre, por exemplo, quando um portador executa uma compra em um estabelecimento comercial, conforme mostra Figura 2.1. Para a realização da transação, o estabelecimento passa o cartão em um equipamento eletrônico que pode ser um POS (comum em pequenas lojas, restaurantes e postos de gasolina) ou um equipamento integrado com o sistema do estabelecimento (usado em supermercados e lojas de departamentos) ou então o portador fornece as informações do cartão no site do vendedor na *Web*. Nesse momento, um funcionário do estabelecimento ou o próprio portador do cartão, no caso de sites na *Web*, digita a opção de crédito ou débito, o número de parcelas e o tipo de parcelamento. Esse aparelho se comunica com o adquirente, que envia a transação para a bandeira que por sua vez, direciona para o emissor. O emissor decide se a transação será aprovada ou não e envia a decisão de volta para a bandeira, que envia para o adquirente e então, para o estabelecimento [Gadi, 2008].

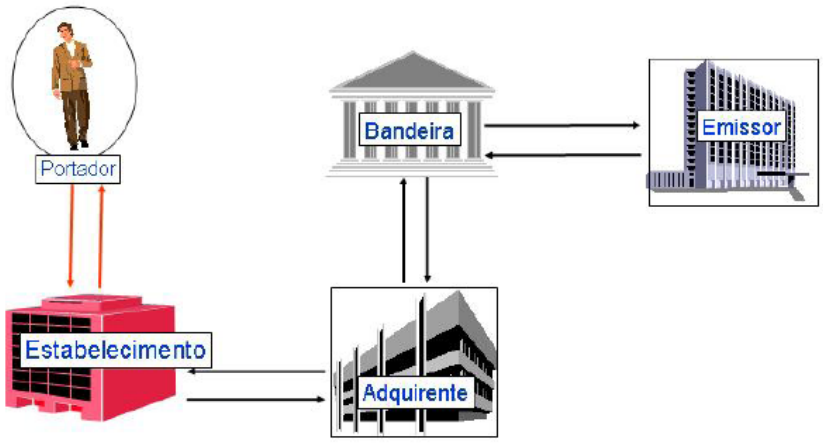

Figura 2.1. Sistema de autorização de uma transação. Fonte:[Gadi, 2008].

2.2 Fraudes de Cartão de Crédito

Fraude com cartão de crédito é uma das maiores ameaças para estabelecimentos comerciais atualmente. No entanto, para combater a fraude de forma eficaz, é importante primeiro entender os mecanismos de sua execução. Fraudadores de cartão de crédito empregam um grande número de *modus operandi* para cometer fraudes.

Em termos simples, fraude de cartão de crédito é definido como quando um indivíduo usa o cartão de crédito de outro indivíduo, enquanto o proprietário e o emissor do cartão não estão cientes do fato do mesmo estar sendo usado. Além disso, o indivíduo que utiliza o cartão não tem nenhuma conexão com seu titular ou emitente e não tem intenção nem de entrar em contato com o proprietário ou fazer reembolsos das compras feitas [Bhatla et al., 2003].

Um dos grandes riscos na venda com cartão é o *chargeback*. Trata-se de um dos maiores temores dos comerciantes de lojas virtuais e muitas vezes podem causar o seu fechamento. *Chargeback* pode ser definido como o cancelamento de uma venda feita com cartão de débito ou crédito, que pode acontecer por dois motivos: Não reconhecimento da compra por parte do titular do cartão; A transação não obedece às regulamentações previstas nos contratos, termos, aditivos e manuais editados pelas administradoras. Resumindo, o lojista vende e depois descobre que o valor da venda não será creditado porque a compra foi considerada inválida. Se o valor já tiver sido creditado ele será imediatamente estornado ou lançado a débito no caso de inexistência

de fundos no momento do lançamento do estorno[1].

Ao contrário da crença popular, os comerciantes sofrem muito mais com o risco de fraude de cartão de crédito do que os portadores de cartão. Enquanto os consumidores podem enfrentar problemas para tentar obter a reversão de uma cobrança fraudulenta, os comerciantes perdem o custo do produto vendido, pagam taxas de *chargeback* e ainda sofrem o risco de ter sua conta de comerciante fechada [Bhatla et al., 2003].

Cada vez mais, o cenário de cartão não presente como por exemplo, fazer compras na internet representa uma ameaça maior para os comerciantes, uma vez que não possuem as vantagens das verificações físicas tais como, verificação de assinatura, identificação com fotografia, etc. Na verdade, é quase impossível realizar qualquer uma das verificações do "mundo físico" necessárias para detectar quem está do outro lado da transação. Isso torna a internet extremamente atraente para quem deseja cometer fraude. De acordo com Bhatla et al. [2003], a taxa em que a fraude ocorre na internet é de 12 a 15 vezes maior do que a fraude no "mundo físico". No entanto, os recentes desenvolvimentos tecnológicos estão mostrando algumas promessas para verificar fraude no cenário de cartão não presente.

2.2.1 Como as Fraudes são Cometidas no Mundo

Existem muitas maneiras pelas quais os fraudadores cometem uma fraude de cartão de crédito. Como a tecnologia muda, a tecnologia utilizada pelos fraudadores também muda e portanto, a maneira como eles irão realizar as atividades fraudulentas. Fraudes podem ser classificadas em três categorias, são elas: fraudes relacionadas a cartão, fraudes relacionadas a comerciantes e fraudes de internet. As diferentes categorias para se cometer fraudes de cartão de crédito são descritas a seguir.

2.2.1.1 Fraudes Relacionadas a Cartão de Crédito

1. Fraude de Proposta *(FA - Fraud Application)*: Um fraudador tenta criar uma pessoa fictícia ou usa dados roubados de alguma pessoa para abrir uma conta de cartão de crédito. Na maioria das vezes, o endereço apresentado diverge do endereço do cliente honesto, que muitas vezes nunca entrou em contato com a instituição. Modelos de *Fraud Application* e cruzamento de informações com *bureaus* [2] internos e externos (como a Serasa e o ACSP) são altamente eficazes

[1] Fonte: http://www.cursodeecommerce.com.br/blog/chargeback/

[2] No mercado financeiro um bureau é uma instituição responsável pela operacionalização de algum compartilhamento de informação. Esse bureau pode ser interno, onde o compartilhamento acontece entre departamentos de uma mesma empresa, ou externo, onde o compartilhamento se dá entre as diferentes empresas associadas ao bureau.

para detecção [Gadi, 2008].

2. Perda ou Roubo: O cliente perde seu cartão ou tem seu cartão roubado e alguma pessoa não autorizada tenta realizar transações com ele. Como normalmente o fraudador desconhece o limite de crédito disponível do cliente, o que se observa é uma sequência de transações de valores pequenos e algumas transações negadas de valores maiores que o disponível, até que se descubra alguma transação que caiba no limite de crédito [Wikipedia, 2010].

3. Aquisição de Conta: Este tipo de fraude ocorre quando um fraudador obtém ilegalmente informações pessoais válidas dos clientes. O fraudador assume o controle de uma conta legítima fornecendo o número da conta do cliente ou o número do cartão. O fraudador, em seguida, entra em contato com o emissor do cartão como sendo o titular para pedir que as correspondências sejam redirecionadas para um novo endereço. O fraudador relata a perda do cartão e pede que um substituto seja enviado para o novo endereço [Bhatla et al., 2003].

4. Cartões Falsos: A criação de cartões falsificados juntamente com cartões perdidos/roubados representam a maior ameaça em fraudes de cartão de crédito. Fraudadores estão constantemente encontrando meios novos e mais inovadores para criar cartões falsificados. Algumas das técnicas utilizadas para a criação de cartões falsos estão listadas abaixo:

    a) Apagar a faixa magnética: Um fraudador pode adulterar um cartão existente que tenha sido adquirido ilegalmente apagando a tira magnética com um eletroímã poderoso. O fraudador então, altera os detalhes do cartão de modo que eles coincidam com os detalhes de um cartão válido. O fraudador ao usar o cartão deixa que o caixa tente passá-lo no terminal várias vezes sem sucesso. O caixa então vai proceder a inserção manual dos dados do cartão no terminal. Esta forma de fraude tem alto risco porque o caixa irá olhar para o cartão de perto para ler os números. Cartões adulterados, assim como, muitos dos métodos tradicionais de fraude de cartão de crédito estão se tornando um método ultrapassado de acumulação ilícita de qualquer fundo ou bem [Bhatla et al., 2003].

    b) Clonagem *(Skimming)*: Maioria dos casos de fraude de falsificação envolve *skimming*, um processo em que dados reais sobre a tarja magnética de um cartão eletrônico é copiado para outro. *Skimming* é a forma mais popular de fraude de cartão de crédito. Empregados/caixas de estabelecimentos comerciais foram encontrados com aparelhos de bolso para *skimming*, com os

quais obtém os dados do cartão do cliente. O fraudador faz isso, enquanto o cliente está esperando para a transação ser validada. Em outros casos, os detalhes obtidos por *skimming* são usados para realizar fraudes em transações com cartão não presente. Muitas vezes, o titular do cartão não tem conhecimento da fraude até que chegue a fatura do cartão [Bhatla et al., 2003].

2.2.1.2 Fraudes Realizadas por Comerciantes

Fraudes relacionadas aos comerciantes são iniciadas tanto por proprietários do estabelecimento quanto pelos empregados. Os tipos de fraudes realizadas por comerciantes estão descritas abaixo:

1. Conluio entre comerciantes: Este tipo de fraude ocorre quando os proprietários e/ou seus funcionários conspiram para cometer fraudes usando contas e/ou informações pessoais de seus clientes. Proprietários e/ou funcionários passam as informações dos titulares dos cartões para os fraudadores [Bhatla et al., 2003].

2. Triangulação: O fraudador neste tipo de fraude opera a partir de um site da *Web*. Os bens, geralmente, são oferecidos a preços bem abaixo do mercado. O site fraudulento parece ser um leilão legítimo ou um site de vendas tradicionais. O cliente ao realizar um pedido *online* fornece informações como nome, endereço e detalhes do cartão de crédito para o site. Uma vez que fraudadores recebem estas informações, compras são realizadas em um site legítimo com as informações do cartão de crédito roubado. O fraudador vai então realizar compras usando os números do cartão de crédito do cliente [Bhatla et al., 2003].

2.2.1.3 Fraudes na Internet

A Internet tem proporcionado um terreno ideal para se cometer fraudes de cartão de crédito de uma maneira fácil. Fraudadores agora começam a operar em um nível verdadeiramente global. Com a expansão da tecnologia, a internet tornou-se um novo mercado mundial, capturando os consumidores da maioria dos países ao redor do mundo. As técnicas mais comumente utilizadas em fraudes de internet são descritas abaixo:

1. Clonagem de site: os fraudadores podem clonar um site inteiro ou apenas as páginas de onde o usuário realiza a compra. Clientes não tem nenhuma razão para desconfiar do site, porque as páginas que eles estão acessando são idênticas

às do site original. O site clonado irá receber essas informações e enviar ao cliente um recibo da transação via e-mail, assim como a empresa faria. O consumidor não suspeita de nada, enquanto os fraudadores tem todos os detalhes que necessitam para cometer fraudes de cartão de crédito [Bhatla et al., 2003].

2. Sites comerciais falsos: Estes sites muitas vezes oferecem ao cliente um serviço extremamente barato. Eles solicitam informações completas do cartão de crédito em troca de acesso ao seu conteúdo. A maioria destes sites pretendem ser livres, mas requerem um número válido de cartão de crédito para verificar a idade dos indivíduos. Os sites são criados para acumular o maior número de cartões de crédito possível. Eles não cobram dos indivíduos pelos serviços que prestam. São geralmente parte de uma grande rede criminosa que usam os detalhes que recolhem para aumentar as receitas ou vendem detalhes do cartão de crédito válido para fraudadores de pequeno porte [Bhatla et al., 2003].

3. Geradores de cartão de crédito: geradores de número de cartão de crédito são programas de computador que geram informações válidas de números e datas de validade. Esses geradores funcionam criando listas de números de cartão de crédito de conta única. O software funciona usando o algoritmo de Luhn[3] que emissores de cartões usam para gerar combinações de números válidos. Os geradores permitem aos usuários gerar, ilegalmente, quantos números desejarem no formato de qualquer um dos cartões de crédito seja American Express, Visa ou MasterCard [Bhatla et al., 2003].

4. *Phishing Scam*: O *Phishing Scam* serve para capturar dados de identificação de uma pessoa, como nome, CPF e endereço para uso posterior por um fraudador. Suas características consistem basicamente no envio de e-mails fraudulentos, no qual o autor convence o usuário a baixar e executar um programa malicioso. Muitas vezes, o e-mail finge ser uma mensagem autêntica, proveniente de uma grande empresa, trazendo uma boa formatação, logotipos e outras características da empresa. Em outras ocasiões, é apenas um assunto curioso que leva a vítima a efetuar o download e executar o arquivo [Gadi, 2008].

5. *Botnets* ou Redes de Robôs: As *Botnets* são criadas a partir de programas (os *bots* ou robôs), que possuem as características de cavalos-de-tróia e worms. Esses

[3]O Algoritmo de Luhn foi criado por Hans Peter Luhn (1896-1964), cientista da computação que trabalhou na IBM. O objetivo deste algoritmo é criar um dígito de verificação para uma sequência de números. O maior uso do algoritmo de Luhn é com cartões de crédito, onde as operadoras geram os n - 1 dígitos iniciais (da esquerda para a direita) e o enésimo dígito é calculado de acordo com os anteriores.

programas em geral invadem os computadores explorando as brechas de segurança ou fazem com que os usuários distraídos os instale em suas máquinas. Os computadores invadidos são levados a se conectar em canais de *Internet Relay Chat (IRC)* que se trata de uma rede de bate-papo *online* que aceita comandos de determinados softwares. Uma pessoa que controle esses canais consegue controlar os computadores infectados com os *bots* e realizar as ações maléficas que desejar. Estas ações podem ser: ataques distribuídos de negação de serviço (DDoS), envio de *spam*, captura de dados privados no segmento de rede comprometido com o bot, captura do que é digitado no teclado do computador invadido, instalação de programas *(adwares)* para exibição de publicidade *online* e disseminação de novos softwares maliciosos [Gadi, 2008].

2.2.2 Impacto das Fraudes de Cartão de Crédito

Infelizmente, as ocorrências de fraudes de cartão de crédito tem apenas mostrado uma tendência crescente até agora. A atividade fraudulenta em um cartão afeta a todos, ou seja, o titular do cartão, o comerciante, o adquirente e o emitente. Esta seção analisa o impacto das fraudes de cartão de crédito em todos os envolvidos na transação.

2.2.2.1 Impacto das Fraudes para o Titular do Cartão

É interessante notar que os titulares são os menos impactados com as fraudes em transações de cartão de crédito. Na maioria das vezes, a responsabilidade do consumidor para as transações com cartão é limitada pela legislação vigente dos países. Isto é verdade, tanto para cenários com cartões presentes, como para cenários com cartões não presentes. Muitos bancos ainda tem os seus próprios padrões que limitam ainda mais a responsabilidade do consumidor. Eles também possuem uma política de proteção do titular do cartão que cobre a maior parte das perdas do cliente. O titular do cartão tem apenas que relatar compras suspeitas ao banco emissor, que por sua vez, investiga a questão com o adquirente e o comerciante sendo realizado um processo de *chargeback* para recuperação do valor.

2.2.2.2 Impacto das Fraudes para os Comerciantes

Comerciantes são as partes mais afetadas em uma fraude de cartão de crédito, particularmente, mais nas transações com cartão não presente, uma vez que eles têm de aceitar total responsabilidade por perdas devido à fraude. Sempre que um titular

legítimo contesta uma cobrança de cartão de crédito, o banco emissor do cartão enviará uma cobrança retroativa ao comerciante (por meio do adquirente) revertendo o crédito para a transação. No caso, o comerciante não tem qualquer evidência física (por exemplo, assinatura de entrega) disponível para contestar o titular do cartão, sendo quase impossível reverter o *chargeback*. Portanto, o comerciante terá que absorver completamente o custo da transação fraudulenta. Na verdade, esse custo é composto por vários componentes, o que pode incorrer em um montante significativo. A seguir são descritos cada um dos custos de uma transação fraudulenta:

1. Custo dos produtos vendidos: Uma vez que é improvável que a mercadoria seja recuperada em um caso de fraude, o comerciante terá que amortizar o valor dos bens envolvidos em uma transação fraudulenta. O impacto dessa perda será maior para os comerciantes com baixa margem de lucro [Bhatla et al., 2003].

2. Custo de envio: O custo de transporte é geralmente embutido no valor do pedido, o comerciante também terá de absorver o custo de transporte para os produtos vendidos em uma transação fraudulenta. Além disso, os fraudadores normalmente solicitam o envido do pedido com alta prioridade para permitir a conclusão rápida da fraude, resultando em custos de transportes elevados [Bhatla et al., 2003].

3. Taxas de associação de cartão: Visa e MasterCard, por exemplo, tem posto em prática programas bastante rígidos que penalizam os comerciantes que geram *chargebacks* excessivos. Normalmente, se um comerciante excede as taxas de *chargebacks* estabelecidas por um período de três meses (por exemplo, 1% de todas as transações ou 2,5% do volume total em dinheiro), o comerciante pode ser penalizado com uma taxa para cada *chargeback*. Em casos extremos, o contrato do comerciante para aceitar cartões pode ser encerrado [Bhatla et al., 2003].

4. Taxas bancárias: Além das penalidades cobradas por associações de cartões, o comerciante tem que pagar uma taxa de processamento adicional para o banco adquirente por cada *chargeback* [Bhatla et al., 2003].

5. Custo administrativo: Toda transação que gera um *chargeback* requer custos administrativos significativos para o comerciante. Em média, cada *chargeback* requer entre uma a duas horas de processamento. Isto porque para o processamento de um *chargeback* é necessário que o comerciante receba e pesquise a alegação de *chargeback*, entre em contato com o consumidor e responda ao banco adquirente ou emitente com a documentação adequada [Bhatla et al., 2003].

6. Perda de Reputação: manter a reputação é muito importante para os comerciantes e excessos de casos de fraude podem fazer com que os proprietários dos cartões deixem de realizar negócios com um comerciante [Bhatla et al., 2003].

2.2.2.3 Impacto das fraudes nos Bancos (Emissor/Adquirente)

Às vezes é possível que a Emissora/Adquirente assuma os custos de fraude. Mesmo nos casos em que a Emissora/Adquirente não está arcando com o custo direto da fraude, há alguns custos indiretos que serão arcados por eles. Como no caso de cobranças emitidas para o comerciante, existem custos administrativos e de mão de obra que o banco tem que arcar. Os emissores e adquirentes também tem que fazer enormes investimentos em prevenção de fraudes através da implantação de sistemas sofisticados de TI para a detecção de transações fraudulentas [Bhatla et al., 2003].

2.3 Prevenção e Gestão de Risco

Todas as fraudes citadas anteriormente, tem como objetivo roubar as informações do cartão de crédito para que se possa realizar compras por meio desse cartão. As compras são feitas, principalmente, em sites de comércio eletrônico onde é possível apenas utilizar a informação obtida, sem a necessidade de se apresentar fisicamente o cartão, tão pouco realizar as verificações do "mundo físico". O aumento da probabilidade de fraudes devido à facilidade proporcionada pela compra com cartão não presente, em conjunto com as responsabilidades por perdas econômicas de fraude, faz da gestão de riscos um dos desafios mais importantes para os comerciantes da Internet em todo o mundo.

Segundo Bhatla et al. [2003] análises indicam que a defasagem média entre a data da transação e a notificação de *chargeback* pode ser superior a 72 dias. Isto significa que se não houver prevenção da fraude, um ou mais fraudadores poderiam facilmente gerar danos significativos a uma empresa antes que as partes afetadas sequer percebam o problema. Isso comprova ainda mais a importância do processo de gestão de risco.

A Figura 2.2 mostra o processo de gestão de risco utilizado pelas empresas de comércio eletrônico. De forma geral, o processo é executado quando o portador do cartão realiza um pedido que é analisado por ferramentas de triagem automática. Essas ferramentas sinalizam se a transação deve ser encaminhada para a revisão manual, se deve ser aceita ou rejeitada. Os pedidos encaminhados para a revisão manual são analisados por uma equipe de especialistas que irão aceitá-los ou não. Após a etapa de triagem automática e revisão manual, os pedidos estarão totalmente sinalizados, sendo

feita a análise de suas taxas de aceitação e rejeição que irão impactar diretamente nos lucros. Finalmente, a etapa de gerenciamento de disputas irá revelar as perdas com fraude, uma vez que é nessa etapa que as vendas serão realmente finalizadas ou fraudes serão descobertas. As seções seguintes irão descrever cada uma das etapas do processo de gestão de risco onde serão discutidas as informações e ferramentas utilizadas.

Figura 2.2. Processo de gestão de risco. Fonte: [Mindware Research Group, 2011].

2.3.1 Etapa 1: Triagem Automática

Enquanto os fraudadores estão usando métodos sofisticados para obter acesso a informações de cartão de crédito e perpetrar fraude, novas tecnologias estão disponíveis para ajudar os comerciantes a detectar e impedir transações fraudulentas. Tecnologias de detecção de fraudes permitem aos comerciantes e bancos realizarem triagens sofisticadas e automatizadas em transações sinalizando-as como suspeitas ou não. Embora nenhuma das ferramentas e tecnologias aqui apresentadas podem por si só eliminar a fraude, cada técnica fornece um valor incremental em termos de capacidade de detecção.

Comerciantes gerenciam grandes volumes de pedidos *online* e normalmente utilizam uma avaliação automatizada inicial a fim de determinar se um pedido pode representar um risco de fraude. Alguns comerciantes utilizam essa avaliação para cancelar pedidos sem a intervenção humana. Segundo Mindware Research Group [2011] 49% de todos os comerciantes cancelam alguns pedidos, como resultado de seu processo de triagem automática e 57% dos grandes comerciantes indicam que alguns pedidos são cancelados nesta fase conforme indica Figura 2.3.

Ainda segundo Mindware Research Group [2011], comerciantes relataram o uso de três ou mais ferramentas de detecção de fraudes para a triagem automática com uma média de 4,6 ferramentas. Comerciantes maiores relataram o uso de 7,4 ferramentas de detecção em média. As ferramentas mais populares utilizadas para avaliar o risco de fraude online são apresentadas na Figura 2.4, que mostra aquelas atualmente adotadas e aquelas que se planeja utilizar. Nessa figura são apresentadas tanto as ferramentas para

triagem automática quanto para revisão manual. A seguir são apresentadas algumas ferramentas de triagem automática.

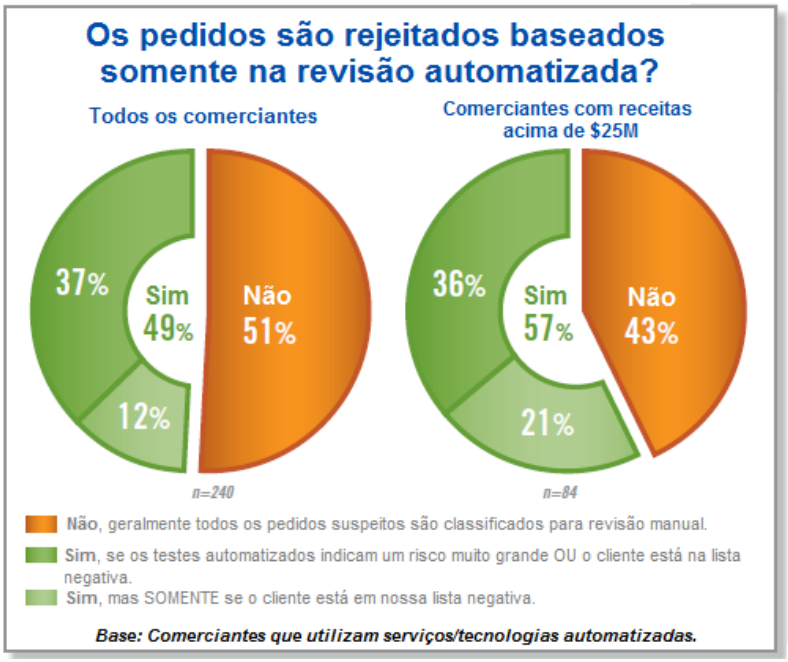

Figura 2.3. Aceitação dos resultados da triagem automática. Fonte: [Mindware Research Group, 2011].

2.3.1.1 Ferramentas de Validação

Estas ferramentas são muitas vezes fornecidas pelas marcas de cartão para ajudar a autenticar cartões e titulares dos cartões. As ferramentas mais frequentemente mencionadas pelos comerciantes são o Número de Verificação do Cartão - *Card Verification Number (CVN)* e do Serviço de Verificação de Endereço (*Address Verification Service* (AVS)):

1. Serviço de Verificação de Endereço verifica as informações do endereço de entrega/faturamento com as informações do titular do cartão. Um código que representa o nível de concordância entre estes endereços é devolvido para o comerciante. Normalmente, não é utilizado exclusivamente *AVS* para aceitar ou rejeitar um pedido.

2. O Número de Verificação do Cartão (CVN - também conhecido como CVV2 para Visa, CVC2 para MasterCard, CID para a American Express e Discover) é a segunda ferramenta de detecção mais comumente usada. O objetivo do CVN em

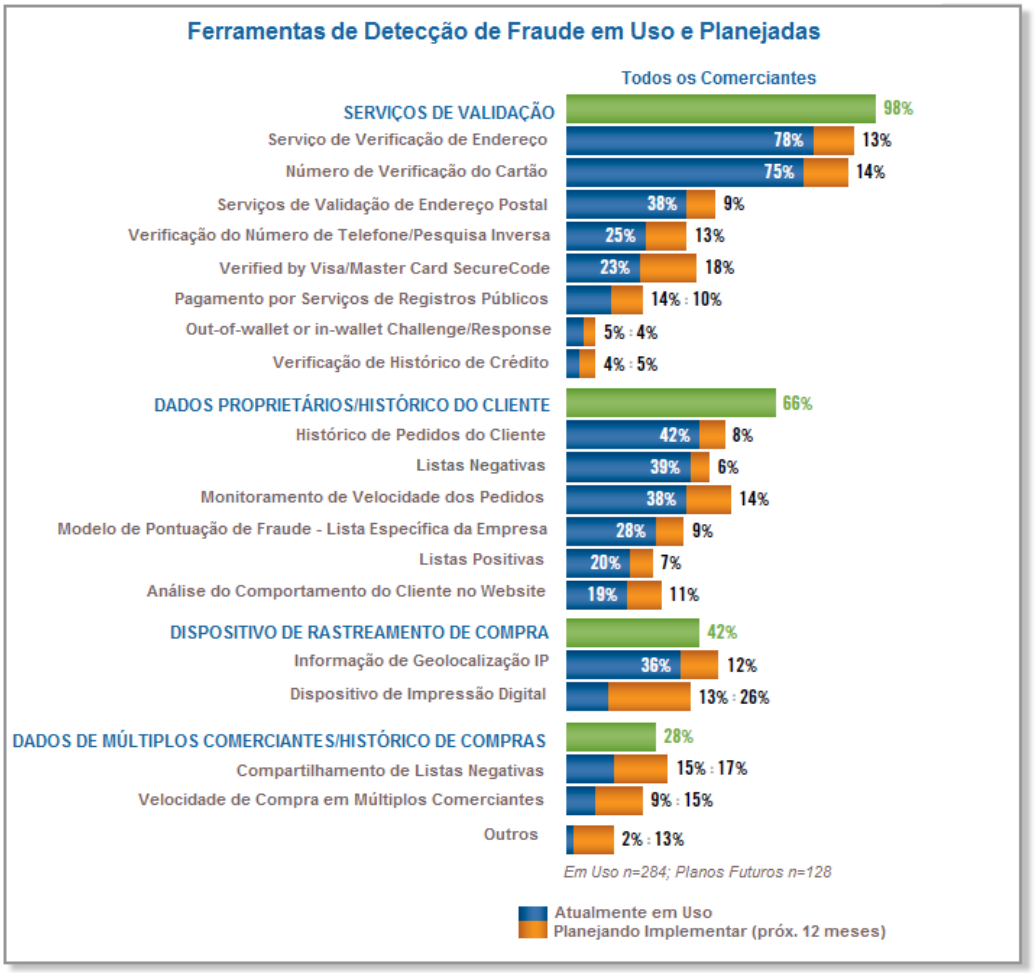

Figura 2.4. Ferramentas adotadas/planejadas para análise de pedidos. Fonte: [Mindware Research Group, 2011].

uma transação com cartão não presente é tentar verificar se a pessoa que realiza o pedido realmente tem o cartão em sua posse. Solicitar o número de verificação do cartão durante uma compra *online* pode adicionar uma medida de segurança para a transação. No entanto, CVNs podem ser obtidos por fraudadores tão facilmente como os números de cartão de crédito. Segundo Mindware Research Group [2011], a utilização de CVN por comerciantes *online* aumentou significativamente nos últimos cinco anos, passando de 44% em 2003 para 75% em 2010.

2.3.1.2 Sistema de Regras

Sistema de regras envolvem a criação de critérios "se ... então" para filtrar as transações. Sistemas baseados em regras dependem de um conjunto de regras projetadas para identificar tipos específicos de transações de alto risco. Regras são criadas

usando o conhecimento sobre o que caracteriza transações fraudulentas. Por exemplo, uma regra poderia ser - Se valor da transação é > "5.000 dólares" e o local de aceitação do cartão = "Casino" e País = "um país de alto risco".

Regras de fraude permitem automatizar os processos de triagem aproveitando o conhecimento adquirido ao longo do tempo sobre as características das transações fraudulentas e legítimas. Normalmente, a eficácia de um sistema baseado em regras vai aumentar ao longo do tempo, uma vez que mais regras são adicionadas ao sistema. Deve ficar claro, entretanto, que em última análise, a eficácia desses sistemas depende do conhecimento e experiência da pessoa que define as regras.

A desvantagem desta solução é que ela pode aumentar a probabilidade de colocar muitas transações válidas como exceções, no entanto, existem maneiras pelas quais esta limitação pode ser superada com algumas medidas como priorizar as regras e fixar limites de número de transações filtradas.

2.3.1.3 Ferramentas de Pontuação de Risco

Ferramentas de pontuação de risco são baseadas em modelos estatísticos projetados para reconhecer transações fraudulentas, com base em uma série de indicadores derivados a partir das características da transação. Normalmente, essas ferramentas geram uma pontuação numérica indicando a probabilidade de uma transação ser fraudulenta: quanto maior a pontuação, mais suspeito é o pedido. Sistemas de pontuação de risco fornecem uma das ferramentas mais eficazes de prevenção à fraude. A principal vantagem da pontuação de risco é a avaliação global de uma transação sendo capturada por um único número. Um sistema de pontuação de risco chega à pontuação final por dezenas de ponderações sobre vários indicadores de fraude derivados de atributos da transação corrente, bem como, atividades históricas do titular do cartão. Por exemplo, quantidades de transação com valores maiores do que três vezes o valor médio da transação do titular do cartão no último ano.

A segunda vantagem da pontuação de risco é que, enquanto uma regra de fraude pode ou não sinalizar uma transação como fraudulenta, a pontuação das transações indica o grau de suspeição sobre cada transação. Assim, as transações podem ser priorizadas com base na pontuação de risco e dada a capacidade limitada da revisão manual, apenas aquelas com a maior pontuação seriam revistas.

2.3.1.4 Lista Negativa/Positiva

Lista negativa é um banco de dados utilizado para identificar transações de alto risco baseadas em campos de dados específicos. Um exemplo de uma lista negativa

seria um arquivo contendo todos os números de cartão que produziram *chargeback* no passado, usado para evitar fraudes reincidentes. Da mesma forma, um comerciante pode construir listas negativas com base em nomes, endereços, *e-mails* e protocolos de Internet (IPs) que resultaram em fraude ou tentativa de fraude, bloqueando efetivamente quaisquer outras tentativas. Um comerciante/adquirente poderá criar e manter uma lista de países de alto risco e decidir rever ou restringir pedidos provenientes desses países [Bhatla et al., 2003].

Outro exemplo popular de lista negativa é o arquivo SAFE distribuído pela MasterCard para comerciantes e bancos membros. Esta lista contém números de cartão que poderiam ser potencialmente utilizados por fraudadores, por exemplo, os cartões que foram relatados como perdidos ou roubados recentemente.

Arquivos positivos são normalmente utilizados para reconhecer os clientes de confiança, talvez por seu número de cartão ou o endereço de *e-mail* e portanto, ignorar determinadas verificações. Arquivos positivos representam uma ferramenta importante para evitar atrasos desnecessários no processamento de pedidos válidos.

2.3.1.5 Geolocalização IP

Ferramentas de geolocalização IP tentam identificar a localização geográfica do dispositivo a partir do qual um pedido *online* foi realizado. Ele fornece uma peça adicional de informação para comparar com as outras informações do pedido e suas regras de aceitação, de forma a ajudar na avaliação do risco de fraude. Em alguns casos, apenas o endereço de um provedor de serviços de Internet é devolvido, por isso, a localização geográfica final do dispositivo permanece desconhecida. Os fraudadores também podem utilizar formas para esconder o seu endereço IP e localização verdadeira *(anonymizers/proxy servers)*.

2.3.1.6 Dispositivos de Impressões Digitais

Esses dispositivos examinam e gravam detalhes sobre a configuração do dispositivo a partir do qual o pedido está sendo feito. Isso pode ajudar a identificar ataques de fraude onde uma variedade de pedidos fraudulentos são realizados a partir de um dispositivo comum ou um conjunto de dispositivos.

2.3.1.7 Serviços de Autenticação do Comprador

Serviços de autenticação do comprador (por exemplo, *Verified by Visa* e *MasterCard SecureCode*) é uma tecnologia emergente que promete trazer um novo nível de segurança para empresas e consumidores na Internet. O programa é baseado em um

Número de Identificação Pessoal (PIN) associado com o cartão e um canal de autenticação seguro e direto entre o consumidor e o banco emissor. O PIN é emitido pelo banco quando o titular do cartão o inscreve no programa e é usado exclusivamente para autorizar as transações *online*.

Quando o titular registrado realiza sua verificação no site de um comerciante participante, será feita a solicitação da senha pelo seu banco emissor. Quando a senha é verificada, o comerciante pode completar a transação e enviar a informação sobre a verificação para a sua adquirente.

2.3.2 Etapa 2: Revisão Manual

Pedidos que foram sinalizados na fase de triagem automática, normalmente, entram numa fila de revisão manual. Durante esta fase, informações adicionais são frequentemente recolhidas para determinar se os pedidos devem ser aceitos ou rejeitados devido ao risco excessivo de fraude. Revisão manual representa uma área crítica de perda de receita e para muitos comerciantes, representa metade do seu orçamento de gestão de risco. Aumentar a produtividade da equipe, mantendo o quantitativo de pessoal, representa um desafio significativo para o crescimento do lucro. Esse desafio se deve ao fato de que, o número total de pedidos que devem ser revistos aumenta na proporção do aumento total das vendas *online*, mesmo quando um percentual estável de pedidos são enviados para revisão.

Embora muitas das ferramentas ou resultados de triagem automática possam ser usados durante a revisão manual, várias ferramentas adicionais e processos são empregados pelos revisores manuais. A seguir são listados alguns deles:

1. Rever o histórico de pedidos do cliente.
2. Entrar em contato com o cliente para confirmação de dados.
3. Entrar em contato com o emissor do cartão.
4. Validar o número de telefone para identificar se o número do titular é igual ao número informado.
5. Consultar lista negativa.
6. Utilizar *Google Maps* para investigar rua e vistas aéreas de endereços de entrega.
7. Pesquisar compradores suspeitos em sites de redes sociais.

As ferramentas mais populares atualmente utilizadas no processo de revisão manual são apresentadas na Figura 2.5, nessa figura pode-se observar também o percentual de comerciantes que planejam adicionar cada ferramenta em 2011. A Figura 2.6 assinala, de acordo com a opinião dos comerciantes, as ferramentas mais efetivas no combate a fraude tanto em revisão automática quanto em manual [Mindware Research Group, 2011].

Onde deveria ser um ambiente altamente automatizado de vendas, a maioria dos comerciantes estão verificando manualmente os pedidos o que acarreta em taxas de revisão manual muito altas. Dadas as limitações sobre a contratação de pessoal adicional para revisão manual, há o aumento do foco no investimento na melhoria da precisão de classificação das ferramentas automatizadas. Isso irá permitir a diminuição da necessidade de, manualmente, rever os pedidos. Deve-se também investir em ferramentas e sistemas para aumentar a produtividade e a eficácia da equipe de revisão.

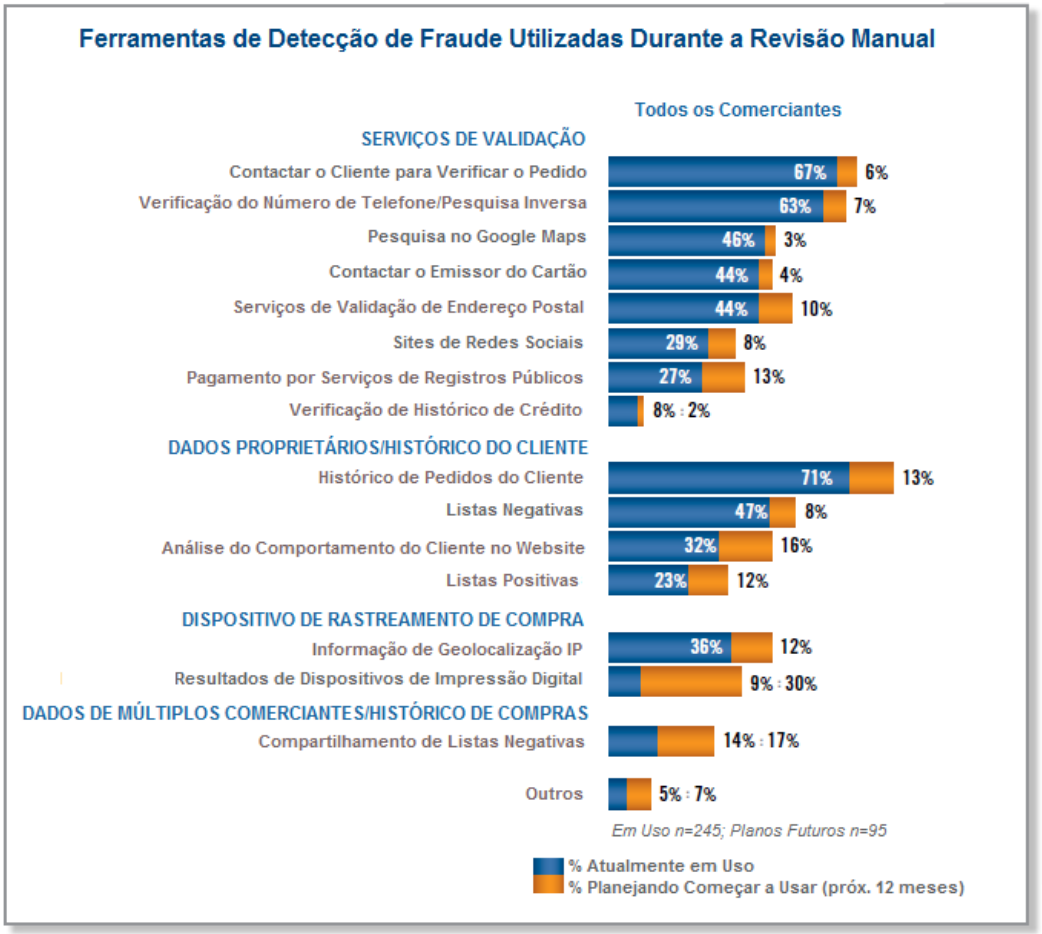

Figura 2.5. Ferramentas adotadas/planejadas para análise manual de pedidos. Fonte: [Mindware Research Group, 2011].

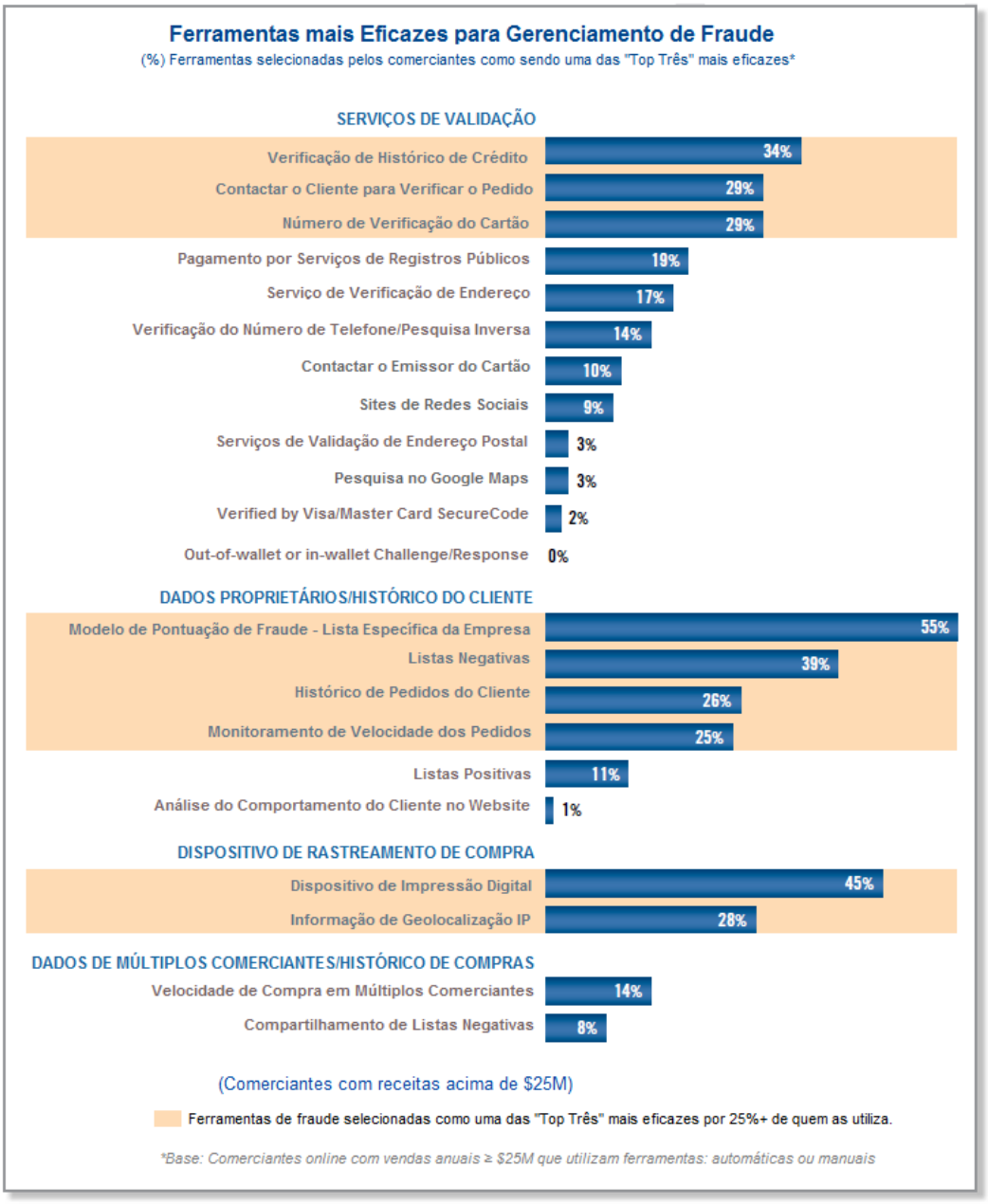

Figura 2.6. Ferramentas mais efetivas no combate a fraude. Fonte: [Mindware Research Group, 2011].

2.3.3 Etapa 3: Situação do Pedido (Aceitar/Rejeitar)

Triagem automática e revisão manual de pedidos acabará por resultar na sua aceitação ou rejeição. Um percentual relativamente alto de pedidos manualmente revistos são, em última análise, aceitos. Essa alta taxa de aceitação evidencia a necessidade dos comerciantes de melhorar a precisão da triagem automática e reduzir a necessidade de revisão manual.

Taxas de aceitação de pedidos na fase de revisão manual são geralmente elevadas para comerciantes *online* de todos os tamanhos. Segundo Mindware Research Group [2011], a maioria dos comerciantes *online* tiveram a maior taxa média de aceitação de pedidos em 2010, conforme mostra Figura 2.6. Taxas de rejeição de pedidos podem refletir riscos verdadeiros de fraude ou, sinal de perda de lucro em termos de rejeição de pedidos válidos ou, altos índices de revisão manual desnecessários.

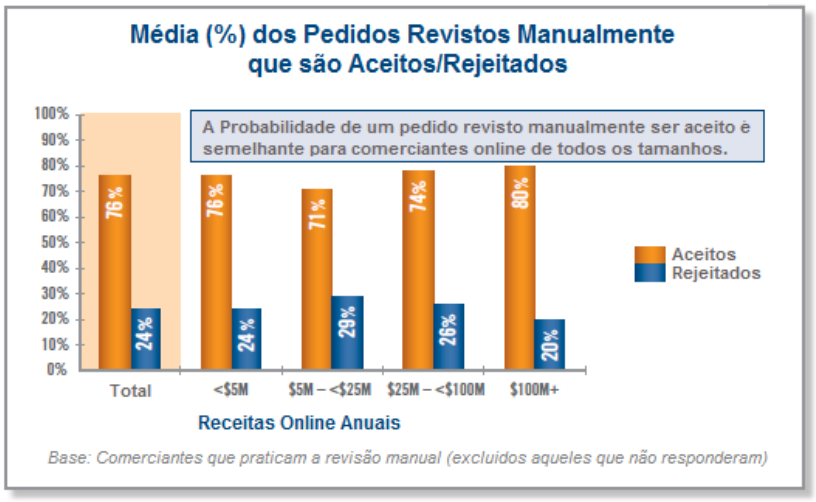

Figura 2.7. Taxas de aceitação e rejeição de pedidos. Fonte: [Mindware Research Group, 2011].

2.3.4 Etapa 4: Gerenciamento de Disputas

Inevitavelmente, alguns pedidos são aceitos e mais tarde são descobertos como sendo fraudulentos. Os comerciantes tentam, de alguma forma, reaver o valor perdido abrindo disputas de *charbeback*. É claro que, uma disputa não é um processo fácil nem gratuito. Comerciantes devem gerenciar e organizar todas as informações do pedido, da entrega e do pagamento para disputar com sucesso esses pedidos com as instituições financeiras. Muitos comerciantes tem adotado sistemas automatizados para lidar com este processo. Em 2010, segundo Mindware Research Group [2011], 63% dos grandes comerciantes relataram o uso ferramentas automatizadas. O tempo médio gasto para lidar com essas disputas foi de 1,8 horas(tempo total consumido para a investigação, documentação e submissão). Os maiores comerciantes relataram um tempo médio de 30 minutos por *chargeback*.

Claramente, o gerenciamento de disputas é uma despesa significativa para os comerciantes. No entanto, ter ferramentas automatizadas que facilitem a contestação das fraudes podem reduzir essas perdas, já que os comerciantes muitas vezes ganham uma

parcela significativa dos *chargebacks* quando esse processo é bem gerenciado. Segundo Mindware Research Group [2011], comerciantes relatam que eles ganham em média 41% das disputas de *chargeback*. Nos últimos cinco anos, a taxa média de ganho de disputas variou de 40% para 44%. A taxa de recuperação líquida é de 24% (o que significa que 24% de todos os *chargebacks* são recuperados). Sendo assim, disputando a maioria dos *chargebacks* e tendo um processo eficiente de gestão dessas disputas, pode-se aumentar a rentabilidade e reduzir a perda de fraude.

2.4 Considerações Finais

Este capítulo apresentou uma ampla discussão sobre as fraudes de cartão de crédito em compras *online*. Inicialmente, foi feita uma descrição do funcionamento do cartão de crédito identificando as entidades envolvidas. Em seguida, foi apresentado os vários tipos de fraudes existentes, que no geral, se resumem a obter as informações do cartão para realização de compras. Neste caso, estas compras serão contestadas pelos titulares do cartão o que acionará um processo de *chargeback*. É descrito também, os impactos associados às fraudes de cartão de crédito e foi observado que o vendedor é a entidade que sofre os maiores impactos. Para minimização da ocorrência de fraude nas compras *online* vários comerciantes investem em processos de gestão de riscos. Esses processos envolvem a utilização de ferramentas automatizadas e também, fazem uso da revisão manual, sendo que essa última envolve os maiores custos. A gestão de risco tem como objetivo tornar as compras na Internet ainda mais seguras e tem sido cada vez mais utilizada por médios e grandes comerciantes.

Capítulo 3

Trabalhos Relacionados

Devido à importância do problema de detecção de fraude, vários são os trabalhos associados a essa área [Chau et al., 2006; Kumar & Nagadevara, 2006; Metwally et al., 2005; Grazioli & Jarvenpaa, 2000; Abbott et al., 1998]. Esses trabalhos buscam criar metodologias que permitam o seu combate ou um melhor entendimento e caracterização da fraude. As técnicas de mineração de dados são empregadas e aperfeiçoadas para lidar com a crescente evolução e expansão do problema. Para que seja possível avaliar a efetividade das técnicas são utilizadas base de dados que buscam representar o cenário real onde as fraudes ocorrem. As próximas subseções buscam identificar e analisar trabalhos associados a cada um desses itens.

3.1 Métodos para Detecção de Fraude

Existem pesquisas que buscam identificar as classes de fraudes e criam metodologias que permitam a sua classificação [Alvarez & Petrovic, 2003; Lindqvist & Jonsson, 1997]. O objetivo dessas pesquisas é compreender melhor o fenômeno das fraudes e identificar aquelas que compartilham alguma similaridade. Thomas et al. [2004] propõem uma árvore de decisão bastante simples que é usada para identificar classes gerais de fraudes. Eles propõem, também, um primeiro passo para uma taxonomia de fraude. Vasiu & Vasiu [2004] propõem uma taxonomia para as fraudes de computador e, para isso, fazem uso de uma metodologia em cinco fases. Segundo os autores, a taxonomia apresentada foi elaborada a partir de uma perspectiva de prevenção e pode ser usada de várias formas. Para eles, essa metodologia pode ser útil como uma ferramenta de conscientização e educação e também pode ajudar os responsáveis pela luta contra a fraude de computador a projetar e implementar políticas para diminuir os riscos.

Pode-se perceber que a criação de metodologias que caracterizem as fraudes têm como objetivo principal ampliar o conhecimento existente do fenômeno e ajudar a projetar meios de prevenção além de servir de base para novas pesquisas. Elas são essenciais para a primeira fase do processo, já que são a partir delas que será possível criar um modelo do problema, bem como definir qual a melhor técnica para a sua resolução. Chau et al. [2006] propõem uma metodologia chamada *2-Level Fraud Spotting (2LFS)* para modelar as técnicas que os fraudadores costumam usar para realizar atividades fraudulentas e para detectar os infratores preventivamente. Essa metodologia é usada para caracterizar os usuários de leilões *online* como honestos, desonestos e cúmplices e para isso, o problema é abordado em duas etapas: (1) são analisadas as características em nível de usuário, ou seja, as informações intrínsecas aos usuários (por exemplo, idade do usuário, o número e os preços dos itens vendidos/comprados, o tempo de transação, dentre outros), e (2) são analisados os recursos de nível de rede para detectar padrões suspeitos na rede de transações entre os usuários. Eles combinam essas duas abordagens utilizando um algoritmo de propagação de crença *(Belief Propagation Algorithm)* em um modelo de grafo *Markov Random Field* para detectar padrões suspeitos.

3.2 Técnicas de Mineração de Dados

Várias são as pesquisas que desenvolvem metodologias para detecção de fraudes [Fawcett & Provost, 1997; Maranzato et al., 2010; Barse et al., 2003; Lundin et al., 2002] e o que pode ser percebido, é que devido às especificidades das fraudes, essas metodologias podem apresentar diferenças significativas em razão das particularidades de cada tipo de fraude. Entretanto, o que pode ser notado é que as técnicas de mineração de dados têm sido amplamente utilizadas na detecção de fraudes independente da metodologia adotada. Isso se deve ao fato dessas técnicas permitirem a extração de informações úteis em bases com grande volume de dados. Phua et al. [2005] realiza uma pesquisa exploratória dos inúmeros artigos associados à detecção de fraude utilizando mineração de dados e apresenta métodos e técnicas juntamente com os seus problemas. Segundo o autor, esses algoritmos são baseados em algumas abordagens como: estratégia supervisionada, estratégia não supervisionada e estratégia híbrida.

Na estratégia supervisionada, algoritmos de aprendizagem examinam todas as transações, rotuladas previamente, para matematicamente determinar o perfil de uma transação fraudulenta e estimar o seu risco. Redes Neurais, *Support Vector Machines (SVMs)*, Árvores de Decisão e Redes Bayesianas são algumas das técnicas utilizadas

por essa estratégia. Maes et al. [1993] utilizou o algoritmo *STAGE* para Redes Bayesianas e o algoritmo *"backpropagation"* para Redes Neurais para detecção de fraudes em transações de cartão de crédito. Os resultados mostram que Redes Bayesianas são mais precisas e mais rápidas para treinamento, mas são mais lentas quando aplicadas em novas instâncias.

Na estratégia não supervisionada, os métodos dispensam o conhecimento prévio das transações fraudulentas e não fraudulentas. Por outro lado, são detectadas alterações no comportamento ou identificadas transações não usuais. Exemplos de técnicas utilizadas são Agrupamentos e Detecção de Anomalias. Netmap [2004] descreve como o algoritmo de agrupamento é usado para formar grupos de dados bem conectados e como ele levou à captura de um fraudador real de seguros. Bolton & Hand [2002] propuseram uma detecção de fraude em cartão de crédito utilizando técnicas de detecção de anomalias em transações. Comportamentos anormais em gastos são identificados e a frequência que eles ocorrem é utilizada para definir quais casos podem ser fraudes.

Na abordagem híbrida (supervisionada e não supervisionada) existem trabalhos utilizando dados rotulados com algoritmos supervisionados e não supervisionados na detecção de fraudes em seguros e telecomunicações. Abordagens não-supervisionadas têm sido utilizadas para segmentar os dados de seguros em grupos para as abordagens supervisionadas. Williams & Huang [1997] aplicam um processo de três passos: *k-means* para detecção dos grupos, *C4.5* para a tomada de decisão e resumos estatísticos e ferramentas de visualização para avaliação da regra.

Existem alguns trabalhos que investigam o uso de um Sistema Imunológico Artificial (*AIS - Artificial Immune System*). A AIS emula o mecanismo de sistema imunológico que salvam o corpo humano de ataques biológicos naturais complexos. Wong et al. [2012a] discute o uso de AIS em um aspecto do gerenciamento de segurança, conseguindo bons resultados para detecção de fraudes de cartão de crédito. O Sistema Imunológico Artificial (AIS) também foi usado por Wong et al. [2012b] e também tem proporcionado bons resultados para detectar fraudes em transações de cartão de crédito.

Modelos preditivos para detecção de fraudes de cartão de crédito estão em uso ativo na prática. Entre esses, a maioria dos trabalhos têm examinado Redes Neurais Aleskerov et al. [1997]; Brause et al. [1999], o que não é surpreendente, dada a sua popularidade na década de 1990. Um resumo desses trabalhos é dado em Kou et al. [2004], que analisa técnicas analíticas para detecção de fraudes em geral, incluindo fraude de cartão de crédito.

Whitrow et al. [2009] avalia várias técnicas, incluindo *Support Vector Machines* (SVM) e Florestas Aleatórias para a previsão de fraude de cartão de crédito. Eles

se concentram sobre o impacto do nível de agregação dos dados da transação sobre o desempenho de previsão de fraude. A pesquisa analisa a agregação em diferentes períodos de tempo em dois conjuntos de dados reais e descobre que a agregação pode ser vantajosa, com a agregação da duração do período sendo um fator importante. Ela foi especialmente eficaz com Florestas Aleatórias que demonstraram um melhor desempenho em relação às outras técnicas, embora a Regressão Logística e *Support Vector Machines* também tiveram um bom desempenho.

SVM e Florestas Aleatórias são técnicas de mineração de dados sofisticadas, que têm sido observadas nos últimos anos e mostram um desempenho superior em diferentes aplicações Larivière & Van den Poel [2005]; Statnikov et al. [2008], SVM é uma técnica de aprendizagem estatística, com forte base teórica e aplicação bem sucedida em uma variedade de problemas Chang & Lin [2011]. Elas estão relacionadas com as Redes Neurais e através da utilização de funções do kernel. Pode ser considerada como um método alternativo para a obtenção de classificadores de Redes Neurais. Ao invés de minimizar o erro empírico em dados de treinamento, SVM procura minimizar o limite superior sobre o erro de generalização. Em comparação com técnicas como Redes Neurais que são propensas a mínimos locais, *overfitting* e ruído, SVM pode obter soluções globais com um bom erro de generalização. A seleção do parâmetro adequado é, no entanto, importante para se obter bons resultados com o SVM. Neste trabalho, que possui dados muito desbalanceados, SVM não fornece bons resultados.

Existe um trabalho muito completo Ngai et al. [2011] que realiza uma revisão da literatura sobre a aplicação de técnicas de mineração de dados para a detecção de fraudes financeiras. Embora a detecção de fraudes financeiras (*FFD - Financial Fraud Detection*) seja um tema emergente de grande importância, uma ampla revisão da literatura sobre o assunto ainda não foi realizada. O artigo representa, assim, a primeira revisão da literatura acadêmica sistemática, identificável e abrangente das técnicas de mineração de dados que foram aplicadas ao FFD. 49 artigos de revistas sobre o assunto, publicados entre 1997 e 2008, foram analisados e classificados em quatro categorias de fraude financeira (fraude bancária, fraude de seguros, fraudes de títulos e commodities e outras fraudes financeiras relacionadas) e seis classes de técnicas de mineração de dados (classificação, regressão, clusterização, previsão, detecção de *outlier* e visualização). Os resultados dessa análise mostram claramente que as técnicas de mineração de dados foram aplicadas mais amplamente para a detecção de fraude de seguros, apesar de fraude corporativa e fraude de cartão de crédito também terem atraído muita atenção nos últimos anos. As principais técnicas de mineração de dados utilizadas para FFD são Modelos Logísticos, Redes Neurais, Rede de Crença Bayesiana e Árvores de Decisão, os quais fornecem soluções principais para os problemas inerentes

à detecção e classificação de dados fraudulentos. O artigo também aborda as diferenças entre FFD e as necessidades da indústria para incentivar a pesquisa adicional sobre temas negligenciados e conclui com algumas sugestões para futuras pesquisas em FFD.

É importante ressaltar que a escolha da técnica de mineração de dados a ser utilizada depende da metodologia definida bem como da base de dados disponível. No caso de existirem dados já rotulados indicando transações fraudulentas, a aprendizagem supervisionada poderá gerar melhores resultados além de permitir a criação de um modelo preditivo para identificação de futuras fraudes. A maioria das pesquisas de detecção de fraudes se baseiam nessa estratégia. A estratégia não supervisionada tem sido utilizada para identificar desvios de comportamento como por exemplo, em ligações telefônicas ou transações de cartão de crédito e assim rotular possíveis transações fraudulentas. No entanto, a combinação dessas duas abordagens pode permitir uma maior acurácia no modelo, em que técnicas de agrupamento podem permitir uma melhor caracterização dos dados e uma melhor escolha daqueles que são utilizados no treinamento em técnicas supervisionadas.

3.3 Cenários de Aplicação

Para que seja possível avaliar a qualidade dos resultados obtidos com as técnicas de mineração de dados é necessária a utilização de uma base de dados que represente de forma precisa o cenário em que a fraude pode ocorrer. O problema mais comum em pesquisas nessa área é a escassez ou a falta de bases reais para a realização de experimentos. Para contornar esse problema, uma alternativa é a criação de dados sintéticos que correspondam de forma aproximada aos dados reais. Barse et al. [2003] justifica que os dados sintéticos possibilitam treinar e adaptar um sistema, além de servir de referência para vários outros sistemas diferentes. Suas propriedades podem ser adaptadas para atender a diversas condições não disponíveis em conjuntos de dados autênticos. O autor também propôs uma metodologia para geração de dados sintéticos baseada na metodologia proposta por Lundin et al. [2002] em que é usada uma pequena quantidade de dados autênticos para a geração de uma grande quantidade de dados sintéticos.

Para Fawcett [2003] uma alternativa é a utilização de dados de spam em *e-mails* no qual é possível estudar as questões de detecção de fraude. Além do mais, os dados de spam estão disponíveis ao público em grandes quantidades. Em contraste com os dados estruturados coletado para detecção de fraudes, os dados não estruturados de *e-mail* exigirão operações de processamento de texto.

Apesar de escassas, algumas bases de dados públicas podem ser encontradas, uma delas é a *Enron* [Shetty & Adibi, 2009] que consiste na base de *e-mails* da empresa *Enron* que atuava no setor de energia entre os anos de 1980 a 2000 e que entrou em colapso em 2001 devido a escândalos contábeis. Durante as investigações que se seguiram ao colapso da empresa, a Comissão Federal Reguladora da Energia tornou pública um grande número de mensagens de *e-mails*. Esses *e-mails* têm sido usados como uma fonte de referência para a investigação em áreas como análise de links, análise de redes sociais, detecção de fraudes e análise textual. O *site* da *UCI Machine Learning Repository* [Frank & Asuncion, 2010] disponibiliza uma coleção de bases de dados de várias áreas como por exemplo, base de dados médica, de flores, de reconhecimento de face dentre outros. Nesse repositório está disponível a base *KDD Cup 1999 Data* que contém uma grande variedade de invasões simuladas em um ambiente de rede militar e que foi utilizada na *Third International Knowledge Discovery and Data Mining Tools Competition* realizada em conjunto com o *KDD-99*. Essa base pode ser usada para testes em algoritmos de detecção de intrusões em computadores.

É possível também utilizar bases de dados de empresas privadas para realização de testes. Maranzato et al. [2010] utiliza a base de dados do *site* de comércio eletrônico TodaOferta[1] pertencente ao Provedor de Serviços de Internet chamado Universo Online (UOL)[2] para a realização de testes em algoritmos de mineração de dados para detecção de fraude de reputação. No entanto, é importante ressaltar que a obtenção de dados reais de empresas para fins de pesquisa é extremamente difícil devido a razões legais e competitivas.

3.4 Considerações Finais

Apesar de todos os trabalhos existentes para detecção de fraude, não foi encontrado na literatura algum trabalho tão abrangente quanto o apresentado nesta dissertação. Sendo assim, o diferencial desta pesquisa está na sua abrangência, uma vez que é realizado um estudo completo sobre as fraudes de cartão de crédito e é apresentada uma metodologia para combater essas fraudes. Essa metodologia mostra-se bastante completa, uma vez que se inicia com a criação de um *dataset*, mostrando técnicas para seleção dos melhores atributos e a preparação adequada dos dados. É feita também uma seleção das melhores técnicas para deteção de fraude onde são apresentados os critérios adotados para escolha das técnicas. E finalmente, são feitas avaliações dos

[1] http://www.todaoferta.com.br
[2] http://www.uol.com.br

resultados alcançados com as técnicas e para isso, foi criada uma nova medida denominada eficiência econômica que avalia os ganhos financeiros obtido com as técnicas de mineração de dados.

Capítulo 4

Metodologia para Detecção de Fraude

Empresas de vendas pela Internet lidam com milhares de transações ao longo do dia o que torna inviável a análise manual de cada uma delas com o objetivo de decidir, de forma *online*, se uma transação é ou não fraudulenta. Além do mais, este claramente não é um problema de classificação fácil de resolver, já que além do grande volume de dados envolvidos, as transações de fraude não ocorrem com frequência. Sendo assim, há uma necessidade de teorias computacionais e ferramentas para ajudar os seres humanos nessa tarefa não trivial de classificação. Este capítulo busca apresentar uma metodologia para detecção de fraudes com o objetivo de criar modelos de classificação que auxiliem na identificação de registros fraudulentos. Mais do que isso, essa metodologia abrange questões que vão desde a extração de dados para a formação de um *dataset*, passando pela avaliação de técnicas mais promissoras para identificação de fraude, até questões relacionadas ao retorno financeiro obtido com as técnicas. A metodologia apresentada tem como base o Processo de Descoberta de Conhecimento em Banco de Dados *(KDD - Knowledge Discovery in Databases)*. Entretanto, são feitas inclusões, modificações e orientações focando-se no problema de detecção de fraude. Este capítulo se inicia com uma visão geral do processo *KDD* de forma a contextualizar o leitor sobre os conceitos envolvidos.

4.1 Processo de Descoberta de Conhecimento em Banco de Dados

Para conversão de dados brutos em informações úteis a área de Descoberta de

Conhecimento em Banco de Dados *(KDD - Knowledge Discovery in Databases)* utiliza-se de um processo. Ele consiste em um conjunto de etapas de processamento que devem ser seguidas pelos profissionais quando executam um projeto de descoberta de conhecimento. Esse processo descreve procedimentos que são realizados em cada um de suas etapas. Ele é usado, principalmente, para planejar, executar e reduzir o custo de um determinado projeto [Cios et al., 2007].

Desde os anos 90, vários processos *KDD's* diferentes foram desenvolvidos. Os esforços iniciais foram liderados pela pesquisa acadêmica, mas foram rapidamente seguidos pela indústria que tem como exemplo o modelo *CRISP-DM (CRoss-Industry Standard Process for Data Mining)* detalhado por Shearer [2000] e o modelo definido por Cabena et al. [1998]. O primeiro processo *KDD* foi proposto por Fayyad [1996] e mais tarde novas propostas foram discutidas por: Adriaans & Zantinge [1996]; Simoudis [1996]; Mannila [1997]; S. & A. [1998]; Cios et al. [2007]. Independente da proposta escolhida, o processo consiste em várias etapas que são executadas em uma sequência. Cada passo subsequente é iniciado após a conclusão do passo anterior e requer o resultado gerado pelo passo anterior como sua entrada.

Cios et al. [2007] realiza uma comparação entre os processos *KDD's* existentes e afirma que as principais diferenças entre eles encontram-se no número de passos e no escopo de alguns passos específicos. Este trabalho tem como base o processo descrito por Fayyad [1996] que é altamente popularizado na literatura sendo fortemente citado. Além do mais, existem vários estudos e documentações [Fayyad, 1996; Fayyad et al., 1996c,a,b] e também, esse processo tem sido utilizado em projetos reais de descoberta de conhecimento.

Segundo Fayyad [1996] *KDD* é um processo não trivial, interativo e iterativo, para a identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados. Com o objetivo de melhorar o entendimento da definição apresentada são descritos a seguir os termos utilizados:

1. O termo *processo* implica que *KDD* compreende várias etapas que envolvem preparação dos dados, procura de padrões, avaliação do conhecimentos e refinamentos, tudo repetido em várias iterações.

2. A expressão *não trivial* alerta para a complexidade normalmente presente na execução de processos de *KDD*. Alguma pesquisa ou inferência está envolvida, isto é, não é um cálculo simples das quantidades pré-definidas como o cálculo do valor médio de um conjunto de números.

3. O termo *interativo* revela a necessidade de um elemento que seja responsável pelo controle do processo.

4. O termo *iterativo*, por outro lado, sugere a possibilidade de repetições integrais ou parciais do processo de *KDD* na busca de resultados satisfatórios por meio de refinamentos sucessivos [Boente et al., 2007].

5. O termo *padrão* é a expressão do conhecimento em alguma linguagem que descreva o subconjunto dos dados ou o modelo aplicável ao subconjunto. Os padrões descobertos devem ser válidos e com algum grau de certeza. Eles também devem ser novos (pelo menos para o sistema e de preferência para o usuário) e potencialmente úteis, isto é, devem fornecer uma certa vantagem para o usuário ou a tarefa. Finalmente, os padrões devem ser compreensíveis, se não imediatamente, ou pelo menos depois de algum pós-processamento.

Como se pode perceber, *KDD* é um processo cooperativo onde os desenvolvedores irão projetar as bases de dados, descrever os problemas e definir os objetivos, enquanto os computadores irão processar os dados a procura de padrões que coincidam com os objetivos estabelecidos. A Figura 4.1 mostra o processo *KDD* segundo Fayyad et al.. A seguir é feita uma descrição de cada uma das etapas exibidas na figura.

1. Seleção dos dados: Nesta etapa é feita uma seleção de um conjunto de dados em que a descoberta de conhecimento será executada.

2. Pré-processamento: Nesta etapa é feita a limpeza e pré-processamento dos dados. As informações selecionadas na etapa anterior podem apresentar problemas como dados redundantes, ruidosos, incompletos e imprecisos. Com o intuito de resolver esses problemas, são definidas estratégias para tratamento desses dados.

3. Transformação: Nesta etapa os dados podem ser transformados e/ou reduzidos. Os dados são efetivamente trabalhados onde são utilizadas técnicas de agregação, amostragem, redução de dimensionalidade, discretização, binarização, dentre outras.

4. Mineração de dados: Nesta etapa é feita a busca pelos padrões nos dados. Nela é definida a tarefa de mineração a ser executada(classificação, regressão, agrupamento, dentre outras), definidos os algoritmos a serem utilizados e é realizada a mineração propriamente dita.

5. Interpretação dos resultados: Nesta etapa os resultados gerados pela mineração de dados são visualizados, interpretados e avaliados se possuem alguma validade para o problema.

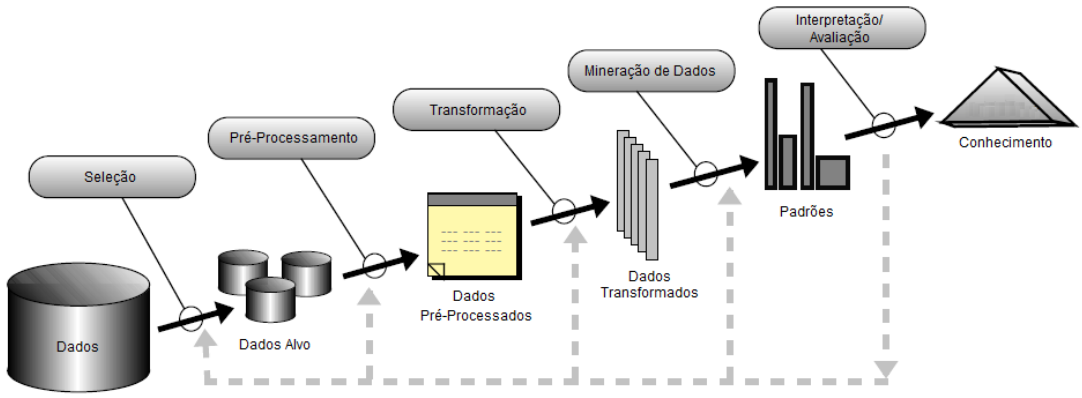

Figura 4.1. Visão geral das etapas que compõem o processo *KDD*. Fonte: [Fayyad et al., 1996b].

Por definição, *KDD* é um campo interdisciplinar que reúne pesquisadores e profissionais de uma grande variedade de áreas. Dentre os campos relacionados pode-se citar: estatísticas, aprendizado de máquina, inteligência artificial, reconhecimento de padrões, bancos de dados, recuperação de informação, visualização, computação paralela e distribuída [Fayyad, 1996]. O processo de *KDD* pode ser visto então como uma atividade multidisciplinar que abrange técnicas que vão além do âmbito de qualquer disciplina específica. Essa infinidade de áreas desempenham papéis chave na descoberta de conhecimento.

Um aspecto muito importante do *KDD* é o tempo relativo gasto para concluir cada um dos passos. A avaliação deste esforço permite uma programação precisa. Várias estimativas têm sido propostas por pesquisadores e profissionais da área [Cabena et al., 1998; Shearer, 2000; Cios & Kurgan, 2005]. A Figura 4.2 mostra uma comparação destas estimativas. Importante notar que, os números apresentados são apenas estimativas que são utilizadas para quantificar o esforço relativo e a sua soma pode não ser igual a 100%. Os valores estimados dependem de muitos fatores, tais como o conhecimento existente sobre o domínio do projeto, o nível de habilidade dos recursos humanos, a complexidade do problema, dentre outros. Pode-se perceber que a etapa de preparação dos dados é de longe a parte mais demorada do processo [Cios et al., 2007]. Existem várias razões pelas quais esta etapa requer uma maior quantidade de tempo, dentre as quais pode-se enumerar: os dados recolhidos por empresas corporativas pos-

suem cerca de 1% a 5% de erros, muitas vezes os dados são redundantes (especialmente entre bancos de dados), inconsistente e também as empresas podem não coletar todos os dados necessários. Estes sérios problemas de qualidade de dados contribuem para o elevado tempo gasto no passo de pré-processamento/transformação dos dados [Pal & Jain, 2005].

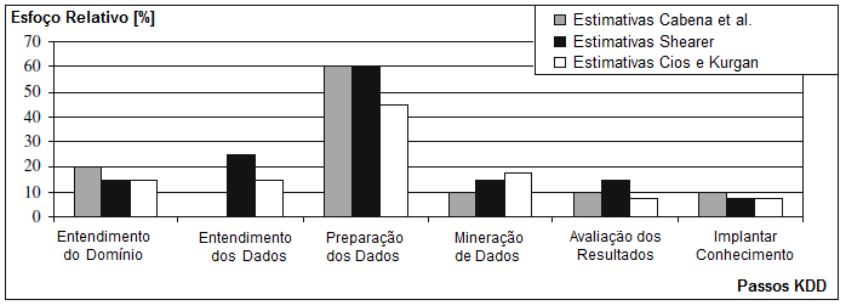

Figura 4.2. Esforço relativo gasto em etapas específicas do processo *KDD*. Fonte: [Cios et al., 2007].

Esta seção apresentou uma visão geral sobre o processo de descoberta do conhecimento já que ele é base para a metodologia proposta. As sequências de passos definidas nesse processo serão utilizadas pela metodologia, uma vez que permitem nortear a tarefa de detecção de fraude. Entretanto, por se tratar de um processo genérico que busca englobar várias áreas do conhecimento, alterações e orientações associadas ao contexto de fraude são necessárias de forma a torná-lo mais aderente ao domínio do problema. As próximas seções apresentam uma descrição detalhada de cada uma das etapas da metodologia onde são apresentadas orientações associadas ao cenário de fraude. O objetivo é guiar a aplicação do processo especificamente para esta área.

4.2 Passo 1: Seleção dos Dados

Uma vez que nem toda a informação existente nas grandes bases de dados serão úteis para a detecção de fraude, esta etapa envolve a identificação de quais informações, contidas nessas bases, deverão ser efetivamente consideradas. Entretanto, antes de se iniciar a seleção é necessário o entendimento do domínio do problema e dos dados, pois somente a extração de campos relevantes poderá garantir o sucesso no processo de detecção de fraude. Como forma de assegurar que os campos selecionados sejam realmente representativos, nesta etapa deve-se trabalha em estreita colaboração com especialistas da área. Dessa forma, objetiva-se desenvolver uma compreensão do domínio da aplicação e obtenção de conhecimentos relevantes do contexto e dos dados que

se deseja trabalhar. A partir daí, é possível definir o escopo do problema e identificar de forma específica o objetivo do processo do ponto de vista do cliente.

Um meta-conhecimento sobre a fonte de dados também é importante. Isto inclui a informação semântica fornecida pelo esquema, os domínios, tipos, faixas de valores dos atributos e as relações entre eles. Este meta-conhecimento é geralmente obtido a partir de especialistas do domínio ou pode ser obtido diretamente a partir dos dados ou podem existir documentos que descrevam esses dados, como por exemplo, um modelo entidade-relacionamento de um banco relacional.

Após o entendimento do problema e dos dados inicia-se o processo de seleção onde será decidido quais informações serão necessárias, incluindo o seu formato e tamanho. Será então criado um subconjunto que conterá apenas informações relevantes para a solução do problema. É necessário um entendimento dos dados para que possam ser verificados quanto a características redundantes e irrelevantes. Isso pode ser feito utilizando o conhecimento adquirido e/ou com a ajuda de um especialista do domínio da aplicação. A seleção foca tanto a escolha de atributos quanto de registros.

Nesta etapa, as fontes de armazenamento originais são acessadas para coleta dos dados. Essas fontes podem estar em uma diversidade de formatos: arquivo texto, planilhas, banco de dados, *data warehouse*, dentre outros e podem ficar em um repositório central de dados ou serem distribuídos em múltiplos locais. Para extrair esses dados é necessário ferramentas ou metodologias escolhidas de acordo com as fontes que se está lidando. Por exemplo, em banco de dados deverá utilizar-se de linguagem *SQL* e em arquivos textos, linguagens como *AWK* [1]. Algumas vezes, as fontes não podem ser acessadas diretamente devido a questões de confidencialidade. Sendo assim, é necessário que os proprietários forneçam os dados ou que sejam criados scripts para extração desses dados.

Na maioria dos casos, os dados são armazenados em bancos relacionais contendo um grande número de tabelas. A extração dos dados nesses bancos pode ser uma tarefa complexa e se realizada de forma incorreta pode representar o fracasso de todas as etapas subsequentes. Sendo assim, dada a complexidade e os riscos associados a esse tarefa, é proposta uma metodologia para extração da informação de banco de dados relacionais conforme ilustrado na Figura 4.3. Os passos a serem seguidos são:

1. *Passo 1:* Análise do banco de dados e seleção de tabelas que podem conter informações relevantes para detecção de fraude. Essa etapa deve ser feita em conjunto com os especialistas da área.

[1] http://www.gnu.org/software/gawk/manual/gawk.html

2. *Passo 2:* Agrupamento e junções das tabelas selecionadas considerando as informações existentes nas mesmas. Deve-se definir mecanismos para garantir a integridade dos dados após a realização das junções.

3. *Passo 3:* União dos grupos formados para criação de um *dataset* único.

4. *Passo 4:* Extração do *dataset* gerado para o formato de arquivo texto para que seja possível a utilização das técnicas de mineração de dados.

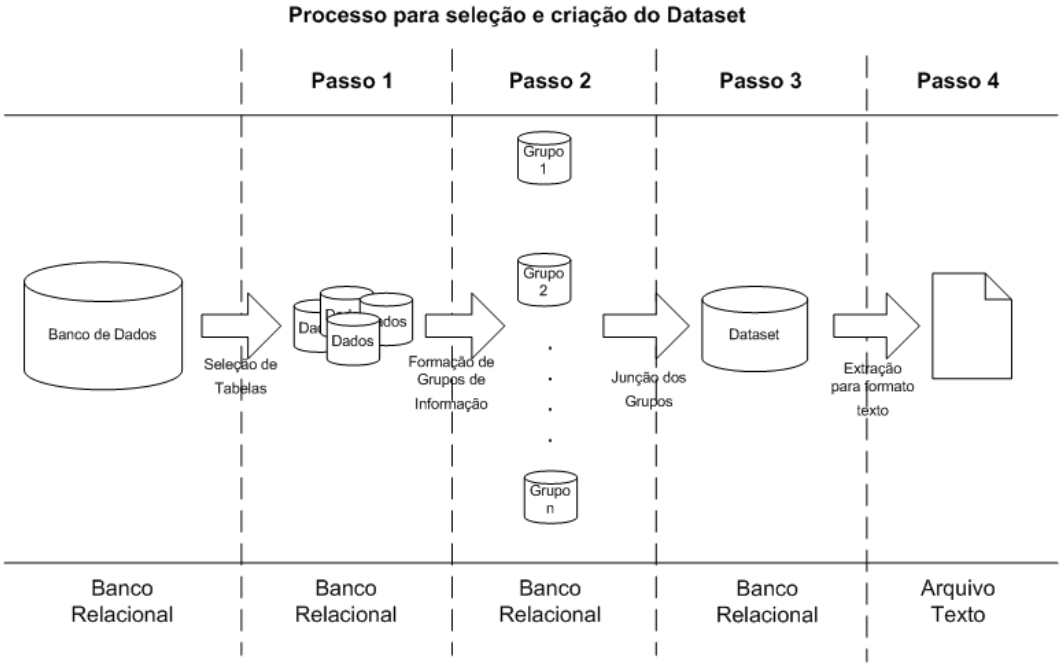

Figura 4.3. Processo de seleção dos dados.

O objetivo com essa metodologia é permitir a criação de um *dataset*, em que cada linha contenha todas as informações de uma transação realizada. Uma vantagem particular é que a criação dessa fonte de dados única fornece uma maior facilidade na transformação para um arquivo simples, o que é conveniente para muitos algoritmos de mineração de dados.

No *passo 2* uma das maiores preocupações é manter a integridade dos dados extraídos, de forma que as junções entre tabelas não incorporem erros no *dataset* final. Sendo assim, com o objetivo de direcionar melhor a análise e facilitar a identificação de erros, sugere-se a realização de junções considerando grupos de informações. Em cada grupo deve ser definido uma *tabela base* e de forma sequencial cada tabela do grupo deve ser unida a essa *tabela base*. Uma forma de garantir que essas uniões não introduzam erros no *dataset*, é definir um processo de auditoria dos dados de forma a

verificar a integridade dos mesmos a medida que as junções são realizadas. O processo aqui definido, consiste em inicialmente, contabilizar o número de registros existentes na *tabela base*, sendo que a cada junção de uma tabela deve ser avaliado se ocorreu alguma alteração no número desses registros. Deve-se fazer também uma verificação amostral de alguns registros de forma a analisar se as informações de relacionamento entre os dados se mantêm condizentes. Qualquer erro identificado deve ser corrigido, para que então uma nova junção possa ser realizada. O processo de auditoria na construção do *dataset* deve ser realizado em cada um dos grupos definidos. Esse processo é mostrado na Figura 4.4. No *passo 3* os grupos criados deverão ser unidos em um único *dataset*. Para isso, deve ser utilizado o mesmo processo de auditoria mostrado na Figura 4.4. Neste caso, um grupo deve ser eleito como representante da *tabela base* e cada grupo deve ser unido a esse representante.

Esta etapa pode definir o sucesso do processo, uma vez que ele depende da escolha correta do subconjunto de dados, pois é nesse subconjunto que será aplicado os algoritmos de mineração de dados. Note que essa, mesmo que criteriosa, é uma seleção inicial dos dados e as etapas posteriores (pré-processamento e transformação) irão refinar ainda mais as informações definidas nessa seleção inicial.

4.3 Passo 2: Pré-Processamento

Não é realista esperar que os dados estejam perfeitos. Podem haver problemas devido a erro humano, limitações nos dispositivos de medição ou falhas no processo de coleta. Podem estar faltando valores ou existir objetos de dados ilegítimos ou duplicados. Nesta etapa se inicia o processo de pré-processamento dos dados para limpeza dos mesmos onde serão aplicadas estratégias para seu tratamento.

Estratégias de visualização, que consiste na exibição dos dados na forma de gráficos ou tabelas, poderão auxiliar nesta etapa. Por meio dos artefatos gerados por esta estratégia é possível a identificação de erros nos dados. Exemplos de técnicas de visualização são: histogramas, gráficos *box plot*, gráficos de pizza, gráficos de porcentagens e funções de distribuição cumulativas, gráficos de dispersão, dentre outros. A seguir, segundo Tan et al. são descritos alguns problemas que podem ser encontrados nos dados e possíveis soluções.

1. *Valores Faltando:* Não é incomum que um objeto não tenha um ou mais valores de atributos. Em alguns casos, as informações não foram coletadas ou os atributos não são aplicáveis a todos os objetos. Independentemente, valores em falta devem ser levados em conta durante a análise de dados. Algumas das estratégias para

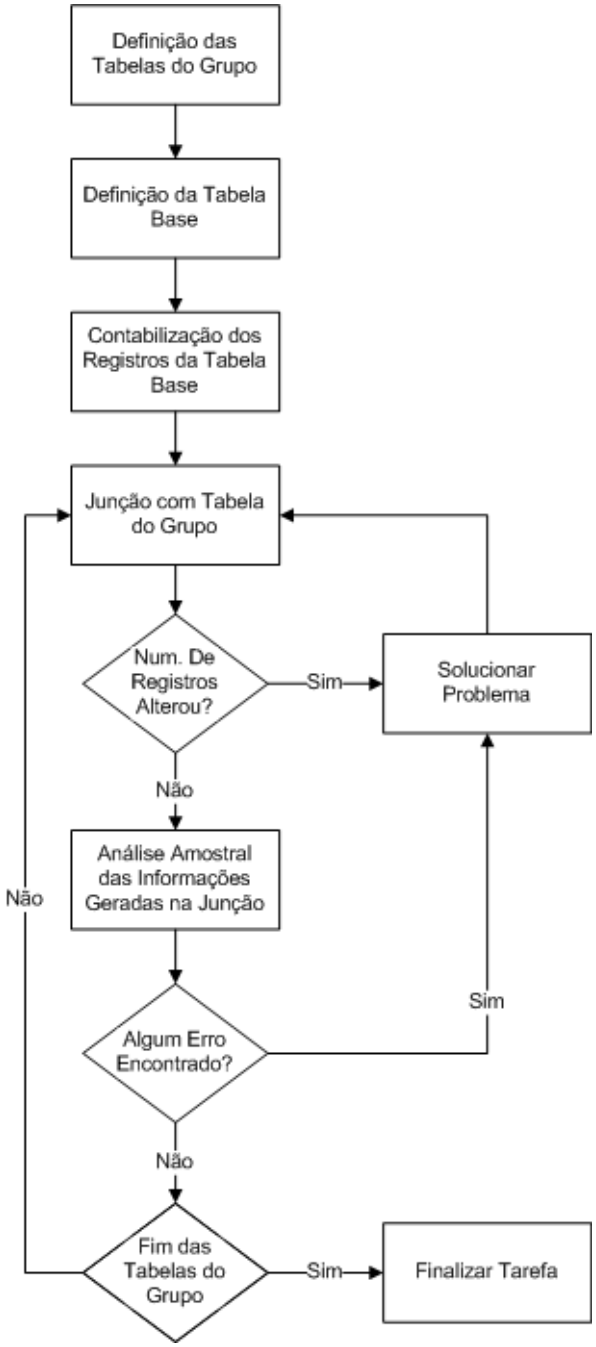

Figura 4.4. Processo de auditoria na construção do *dataset*.

lidar com dados faltantes são: *eliminar objetos ou atributos de dados*, *estimar valores faltantes* e *ignorar valores faltantes durante a análise*.

2. *Valores Inconsistentes:* Dados inconsistentes podem estar associados a alguma falha durante a medição, erro de digitação, dentre outros. Independente da causa, valores inconsistentes devem ser detectados e se possível corrigidos. Além da

visualização, medidas como média, mediana, máximo e mínimo podem auxiliar na identificação desses valores. A correção de uma inconsistência requer informações adicionais ou redundantes.

3. *Dados Duplicados:* Um conjunto de dados pode incluir objetos que sejam duplicata ou quase duplicata uns dos outros. É importante detectar e eliminar tais duplicatas, sendo que deve-se tomar cuidado para evitar eliminar acidentalmente objetos de dados que sejam semelhantes, mas que não são duplicatas, por exemplo, pessoas distintas com nomes idênticos.

Como é possível perceber, os dados muitas vezes estão longe da perfeição. Embora grande parte das técnicas de mineração de dados possam tolerar algum nível de imperfeição, é importante um foco na compreensão e melhora da qualidade dos mesmos. Isso permitirá um aumento da qualidade dos resultados das etapas posteriores.

No contexto da fraude, esta etapa pode apresentar indícios de sucesso do processo de detecção de transações fraudulentas. Isso pode ser percebido verificando se os atributos selecionados são realmente relevantes para o processo de detecção. Gráficos de dispersão podem ser utilizados para verificar como os registros de fraude e não fraude se distribuem em relação a cada um dos atributos. Esse contraponto, entre as duas classes de informações, poderá auxiliar na identificação de regiões do gráfico onde uma determinada classe ocorre com maior frequência, gerando indícios da existência de padrões que possam diferenciá-las. Histogramas podem auxiliar na identificação de como os registros de uma classe se agrupam em relação aos valores possíveis de um determinado atributo. Uma melhor forma de se verificar isso é considerar a distribuição relativa desses valores utilizando a Equação 4.1:

$$Disbribuicao = \frac{Qtdrelacionados}{QtdTotalClasse} \tag{4.1}$$

Onde *Qtdrelacionados* representa a quantidade de registros de uma classe específica associados a um determinado valor do atributo e *QtdTotalClasse* representa a quantidade de registros da classe específica. A construção de histogramas considerando os valores gerados por essa fórmula permitirá que padrões que não são visíveis, devido a diferenças no número de registros entre as classes, se tornem visíveis. Isso permitirá verificar a concentração de registros da classe em determinados valores dos atributos.

Claro que os padrões encontrados considerando as análises acima não serão suficientes para detecção da fraude. Mas, são um forte indício de que os atributos selecionados são relevantes para a tarefa e que por meio das técnicas de mineração de dados será possível extrair novos padrões a partir da combinação desses atributos. Sendo

assim, os resultados das análises feitas acima podem ser consideradas como um critério de sucesso do processo.

4.4 Passo 3: Transformação

Algumas vezes outras informações serão necessárias e poderão ser geradas a partir dos dados selecionados. Sendo assim, são utilizados métodos de transformação para gerar dados novos que sejam relevantes. Restrições computacionais também podem colocar limites severos sobre o subespaço que pode ser explorado por um algoritmo de mineração de dados. Dessa forma, pode ser necessária uma redução da dimensão que se irá trabalhar. Um outro ponto que se deve levar em consideração, é o fato de que, alguns dados podem não estar no formato exigido pelos algoritmos que irão extrair o conhecimento.

Nesta etapa serão aplicadas diferentes estratégias e técnicas para transformação dos dados. A seguir será apresentado algumas idéias e abordagens:

1. *Agregação:* Consiste na combinação de dois ou mais objetos em um único. Existem diversas motivações para a agregação. Primeiro, os conjuntos de dados menores requerem menos memória e tempo de processamento, permitindo o uso de algoritmos mais custosos. Em segundo lugar, pode atuar como uma mudança de escopo ou escala, fornecendo uma visão de alto nível dos dados ao invés de baixo nível. Finalmente, o comportamento de grupos de objetos ou atributos é muitas vezes mais estável do que o de atributos ou objetos individuais. Uma desvantagem é a potencial perda de detalhes interessantes e no caso da detecção de fraude essa perda de detalhes pode interferir na identificação desses registros. Isso acontece porque os registros fraudulentos podem-se diferenciar devido a detalhes sutis e específicos.

2. *Amostragem:* É uma abordagem comumente usada para selecionar um subconjunto dos objetos de dados a serem analisados. Geralmente, usa-se amostragem porque a obtenção do conjunto inteiro de dados é custosa ou o processamento de todo o conjunto de dados pode consumir muito tempo ou ser até mesmo inviável. Os tipos de amostragem são: *amostragem aleatória simples* os dados são selecionados aleatoriamente, *amostragem estratificada* amostras de determinados grupos são selecionadas e *amostragem progressiva* é feita uma amostragem pequena e então é aumentado o tamanho da amostra até que ela possua um tamanho suficiente. A perda de informação também é uma desvantagem da amostragem.

Como geralmente os registros de fraude estão em menor quantidade e muitas vezes com uma frequência baixíssima não é de interesse a amostragem de registros dessa classe. Além do mais, isso fará com que informações e detalhes importantes dos registros de fraude sejam perdidos. Entretanto, uma amostragem apenas dos registros de não fraude pode ser interessante para solucionar o problema da baixa distribuição. No entanto, mais uma vez corre-se o risco de perder informações preciosas para se diferenciar registros não fraudulentos. Uma abordagem que pode se mostrar interessante, seria a eliminação de registros redundantes ou muito semelhantes das classes de não fraude. Isso poderia atenuar o problema da distribuição e do volume de registros sem uma perda considerável de informação.

3. *Construção de Características:* Às vezes as características nos conjuntos de dados originais têm as informações necessárias, mas não estão em uma forma apropriada para o algoritmo de mineração de dados. Nesta situação, uma ou mais características construídas a partir das características originais podem ser mais úteis do que estas. Por exemplo, por meio do campo data de nascimento é possível extrair a idade da pessoa que realizou determinada transação. Por meio desse campo pode ser possível identificar faixas etárias onde a fraude pode ocorrer com uma maior frequência.

4. *Discretização e Binarização:* Alguns algoritmos de mineração de dados requerem que os dados estejam na forma de atributos categorizados ou binários. *Discretização* consiste em transformar um atributo contínuo em categórico. Uma abordagem para essa técnica consiste em criar faixa de valores de tamanhos iguais para os atributos. A discretização pode trazer uma perda de informação o que no contexto de fraude pode representar uma perda significativa. Na *Binarização*, atributos contínuos ou discretos são transformados em um ou mais atributos binários. Uma técnica simples para binarizar atributos categorizados é: se houver m valores categorizados, então crie m atributos e atribua 1 para existência do valor e 0 para a ausência. Uma desvantagem é que com isso aumenta-se a dimensionalidade o que pode levar o problema da *maldição da dimensionalidade* descrito a seguir.

5. *Redução da Dimensionalidade:* Existem diversos benefícios na redução de dimensionalidade. Um deles é que muitos algoritmos de mineração de dados funcionam melhor se a dimensionalidade - o número de atributos - for menor. Além do mais, evita-se o problema da *maldição da dimensionalidade* que se refere ao fato de que quando a dimensionalidade aumenta, os dados se tornam cada vez mais dispersos

no espaço que eles ocupam. Para classificação, isto significa que não há objetos de dados suficientes para permitir a criação de um modelo que atribua de forma confiável uma classe a todos os objetos possíveis. Uma forma de reduzir a dimensionalidade segundo Tan et al. é usar apenas um subconjunto das características. Existem três abordagens padrão para a seleção de características:

a) Abordagens Internas: A seleção de características ocorre naturalmente como parte do algoritmo de mineração de dados. Especificamente, durante a operação o próprio algoritmo decide quais atributos usar e quais ignorar. Algoritmos para construir classificadores de árvore de decisão muitas vezes operam desta maneira.

b) Abordagens de Filtro: Características são selecionadas, antes que o algoritmo de mineração de dados seja executado, usando alguma abordagem que seja independente da tarefa de mineração de dados. Por exemplo, pode-se selecionar conjuntos de atributos cuja *correlação* dos pares seja tão baixa quanto possível. A *correlação* entre dois atributos é uma medida do relacionamento linear entre eles. Ela fica sempre em uma faixa de -1 e 1. A correlação de 1 indica que os atributos possuem um relacionamento linear positivo e correlação -1 indica um relacionamento linear negativo. Maiores detalhes podem ser encontrados em Magalhães & de Lima [2002].

c) Abordagens de Envoltório: Estes métodos usam o algoritmo de mineração de dados alvo como uma caixa preta para encontrar o melhor subconjunto de atributos. Nessa abordagem são experimentados vários subconjunto de atributos, mas geralmente sem enumerar todos os subconjuntos possíveis.

Dificilmente serão encontrados trabalhos que discutam sobre as variáveis utilizadas na prevenção à fraude, isso provavelmente se deve a questões de segurança. No Capítulo 2 foram apresentadas algumas variáveis utilizadas, sendo que algumas delas envolvem a criação de uma nova característica por meio de duas variáveis originais como, por exemplo, o indicador se o endereço de entrega é diferente do endereço do comprador. A seguir, é enumerado conforme apresentado por Gadi [2008], algumas das principais classes de informação utilizadas para detecção de fraude e para cada uma das classes são apresentados alguns exemplos de possíveis variáveis preditoras.

1. Demográficas do cliente:

   a) Variáveis associadas à cidade ou região de origem do cartão.

   b) Data de expiração do cartão.

c) Limite do cartão.

2. Demográficas do estabelecimento:

   a) Variáveis que se relacionam à cidade ou região.

   b) Categoria do estabelecimento.

   c) Porte do estabelecimento.

3. Perfil do Cliente:

   a) Lista de estabelecimentos mais utilizados pelo cliente.

   b) Valor médio das compras.

   c) Tempo de registro.

   d) Status do CPF.

   e) Número de casos de *chargeback*.

   f) Valor de pico nos últimos 12 meses.

   g) Indicador de uso anterior na Internet.

   h) Indicador de uso anterior internacional.

   i) Indicador *Transactor*(cliente que paga a fatura em sua totalidade)/*Resolver*(cliente que paga parte da fatura, entre o mínimo e o total, rolando o saldo com juros para o próximo mês).

4. Perfil de Fraude:

   a) O país/cidade pertence a uma lista de países em quarentena?

   b) É uma transação real seguida de um teste anterior?

5. Velocidade:

   a) Tempo desde a última transação.

   b) Número de transações (negadas/aprovadas) na última hora.

   c) Número de transações (negadas/aprovadas) no último dia.

   d) Número de transações (negadas/aprovadas) no último mês.

   e) Montante utilizado (aprovado/negado) na última hora.

   f) Montante utilizado (aprovado/negado) no último dia

g) Montante utilizado (aprovado/negado) no último mês.

Note que grande parte das variáveis previamente listadas são obtidas por meio da construção de novas características, tendo como base as características originais. Isso, mais uma vez, destaca a importância dessa etapa de transformação no processo de detecção de fraudes.

4.5 Passo 4: Mineração de Dados

Mineração de Dados é uma etapa do processo que se refere ao ato de extrair padrões ou modelos em grandes volumes de dados. Segundo Fayyad et al. [1996b], os objetivos básicos da mineração de dados são a *previsão* e a *descrição*. A *previsão* envolve o uso de dados existentes no banco de dados para prever valores desconhecidos ou futuros. A *descrição* se concentra em encontrar padrões interpretáveis pelo usuário que descrevam os dados. Embora as fronteiras entre previsão e descrição não são nítidas (alguns dos modelos preditivos podem ser descritivos, na medida em que eles são compreensíveis e vice-versa), a distinção é útil para a compreensão do objetivo da descoberta de conhecimento. No contexto de fraude, técnicas que sejam preditivas e descritivas são mais interessantes uma vez que deixam claro os padrões utilizados para a sua previsão. Além do mais, essa pode ser uma importante ferramenta para entendimento dos padrões associados à fraude.

É nesta etapa que é definida a tarefa de mineração de dados e as técnicas a serem utilizadas. De acordo com Amo [2004], é importante distinguir o que é uma *tarefa* e o que é uma *técnica* de mineração de dados. A *tarefa* consiste na especificação do que se pretende fazer com os dados, ou seja, qual o objetivo do processo. Dentre as tarefas pode-se citar: classificação, análise de associação, agrupamento, etc. A *técnica* consiste na escolha de métodos ou algoritmos que permitam que esses objetivos sejam alcançados. Dentre as técnicas existentes pode-se citar: árvore de decisão, classificador baseado em regras, classificadores bayesianos, meta aprendizagem, dentre outras. A seguir, é descrito algumas das principais tarefas de mineração de dados segundo Amo [2004].

1. Classificação e Predição. Classificação é o processo de construir um modelo que descreve e distingue classes ou conceitos, com o propósito de utilizar o modelo para predizer a classe de objetos que ainda não foram classificados. O modelo construído baseia-se na análise prévia de um conjunto de dados de treinamento contendo objetos corretamente classificados. Considere, por exemplo, o caso desta

pesquisa, onde deseja-se descobrir se uma transação de pagamento *online* feita na Internet é ou não fraudulenta. Um modelo de classificação poderia incluir a seguinte regra: "Compras acima de R$ 50,00, feitas por compradores com idade entre 50 e 60 e com período de registro no site menor que 15 dias são transações fraudulentas". Em algumas aplicações, o usuário está mais interessado em predizer alguns valores ausentes em seus dados em vez de descobrir classes de objetos. Isto ocorre sobretudo quando os valores que faltam são numéricos. Neste caso, a tarefa de mineração é denominada *Predição*.

2. Análise de Regras de Associação. Uma regra de associação é um padrão da forma $X \rightarrow Y$, onde X e Y são conjuntos de valores. Ela é usada para descobrir padrões que descrevam características altamente associadas aos dados. Considere, por exemplo, o caso da identificação de fraude onde o seguinte padrão: "Se aprovação de uma transação é anterior a uma reprovação" representa uma regra de associação que repete um padrão de comportamento de uma transação que pode ser fraudulenta. Descobrir regras de associação de transações fraudulentas e não fraudulentas é útil para entendimento dos padrões associados à fraude.

3. Análise de *Clusters* (Agrupamentos). Diferentemente da classificação e predição, onde os dados de treinamento estão devidamente classificados e os rótulos das classes são conhecidos, a análise de *clusters* trabalha sobre dados onde os rótulos das classes não estão definidos. A tarefa consiste em identificar agrupamentos de objetos, agrupamentos estes que identificam uma classe. Por exemplo, poderia se aplicar análise de *clusters* sobre o banco de dados de um site de vendas *online* a fim de identificar grupos de transações onde existam uma maior predisposição à fraude e assim identificar características comuns que podem auxiliar na sua detecção.

4. Análise de Padrões Sequenciais. Um padrão sequencial é uma expressão da forma $< I_1, ..., I_n >$, onde cada I_i é um conjunto de itens. A ordem em que estão alinhados estes conjuntos repete a ordem cronológica em que aconteceram os fatos representados por estes conjuntos. Assim, por exemplo, a sequência de status de transações $< \{reprovada\}, \{reprovada\}, \{aprovada\} >$ pode representar o padrão de um fraudador tentando identificar o limite disponível de um cartão de crédito para então realizar uma compra até esse limite. Descobrir tais padrões sequenciais em dados temporais pode ser útil na identificação de comportamentos fraudulentos.

5. Análise de *Outliers*. Um banco de dados pode conter dados que não apresentam o comportamento geral da maioria. Estes dados são denominados *outliers*(exceções). Muitos métodos de mineração de dados descartam estes *outliers* como sendo ruídos indesejados. Entretanto, em algumas aplicações, estes eventos raros podem ser mais interessantes do que eventos que ocorrem regularmente. Este é o caso da detecção de fraude, onde os casos fraudulentos geralmente desviam-se dos padrões associados aos casos não fraudulentos. Sendo assim, esses registros raros são mais importantes de serem identificados, uma vez que causam prejuízos a compradores e vendedores.

As várias tarefas de mineração de dados possuem contribuições significativas no processo de detecção de fraude. Entretanto, cada uma dessas tarefas envolvem pesquisas específicas nas suas áreas do conhecimento. A realização de pesquisas em cada uma dessas tarefas envolvem esforços que vão além do escopo deste trabalho. Além do mais, representam não uma, mas várias pesquisas associadas. Sendo assim, este trabalho se limita à tarefa de classificação, pois envolve a construção de modelos para previsão de registros fraudulentos. Além do mais, existem técnicas preditivas em que o modelo gerado permite uma descrição dos padrões encontrados. Essa característica possibilitará um maior entendimento dos casos de fraude.

Para a tarefa de classificação há uma grande variedade de técnicas associadas e a descrição e utilização de cada uma delas também envolvem um esforço além do escopo deste trabalho. Sendo assim, foram selecionadas algumas das técnicas consideradas mais adequadas para detecção de fraude. As técnicas selecionadas são listadas a seguir sendo feita uma justificativa para a sua utilização.

1. *Árvore de Decisão:* uma das grandes vantagens da árvore de decisão é que além de permitir uma modelagem preditiva do problema, ela também fornece uma modelagem descritiva, uma vez que por meio da árvore é possível identificar quais informações foram utilizadas para a classificação de um determinado registro. Essa característica será muito útil para se entender melhor os padrões associados a registros fraudulentos. Além do mais, os atributos utilizados para construção da árvore poderão ser analisados para uma possível redução de dimensionalidade direcionada para essa técnica. Esta técnica também é utilizada nos trabalhos de detecção de fraude realizados por [Philip K. Chan & Stolfo, 1999] e [Phua et al., 2004].

2. *Classificador Baseado em Regras:* a expressividade de um conjunto de regras é quase equivalente a uma árvore de decisão porque uma árvore pode ser represen-

tada por um conjuntos de regras completas e mutuamente excludentes. Dessa forma, assim como a árvore de decisão um classificador baseado em regras permitirá um maior conhecimento sobre os padrões existentes nos registros de fraude. Classificadores baseado em regras são utilizados na pesquisa de detecção de fraudes em cartão de crédito realizada por [Philip K. Chan & Stolfo, 1999]. Foram escolhidos um algoritmo para cada uma das abordagens existentes: abordagem de ordenação das regras representada pelo *RIPPER* e a abordagem de regras não ordenadas representado pelo *LAC*.

3. *Naive Bayes:* Phua et al. utilizam essa técnica em sua pesquisa de fraude e segundo eles, embora esse algoritmo seja simples, ele é muito eficaz em muitos conjuntos de dados do mundo real, pois pode fornecer uma melhor acurácia do que os métodos conhecidos como por exemplo árvores de decisão e rede neural [Domingos & Pazzani, 1996; Elkan, 2001]. Além do mais, ele é extremamente eficiente na medida em que aprende de uma forma linear usando técnicas de meta aprendizagem, tais como *bagging* e *boosting* para combinar as previsões do classificador [Elkan, 1997]. No entanto, quando os atributos são redundantes e não seguem uma distribuição normal a acurácia da previsão é reduzida.

4. *Boosting, Oversampling e Stacking:* essas técnicas foram escolhidas com o intuito de melhorar a qualidade da previsão dos classificadores definidos anteriormente. As duas primeiras focam na melhora da distribuição das classes do *dataset* de forma a solucionar o problema de desequilíbrio de classe. A segunda busca criar um meta classificador agrupando todos os classificadores utilizados e tentando extrair o melhor deles.

De uma forma geral, o critério para escolha dos classificadores foi selecionar aqueles em que o modelo gerado permita a sua interpretação e assim, ser possível um conhecimento maior dos padrões de fraude. Exceção a esse critério foi a utilização do classificador *Naive Bayes*, no entanto ele tem sido muito utilizado em técnicas de meta aprendizado, como *Stacking*. As técnicas de meta aprendizagem são utilizadas mais especificamente neste trabalho para lidar com o problema da baixa distribuição das classes de fraude e assim, fazer com que os classificadores selecionados apresentem melhores resultados. As próximas subseções irão apresentar noções básicas de cada uma das técnicas utilizadas e tem como base os livros publicados por Tan et al. e Written & Frank. Maiores detalhes ou exemplos de outras técnicas podem ser encontrados em [Kantardzic, 2002; Written & Frank, 2005; Han & Kamber, 2005; Veloso et al., 2006; Tan et al., 2009; Torgo, 2010].

4.5.1 Árvore de Decisão

Esta seção introduz um classificador de árvore de decisão, que é uma técnica de classificação simples, porém muito usada. De acordo com Tan et al. [2009], pode-se resolver um problema de classificação fazendo uma série de questões cuidadosamente organizadas sobre os atributos do registro de teste. Cada vez que uma resposta é recebida, uma questão seguinte é feita até que se chegue a uma conclusão sobre o rótulo da classe do registro. A série de questões e suas respostas possíveis podem ser organizadas na forma de uma árvore de decisão, com sua estrutura hierárquica consistindo de nós e arestas direcionadas. A Figura 4.5 mostra a árvore de decisão para o problema da classificação de vertebrados em mamíferos e não-mamíferos. A árvore possui três tipos de nós:

1. Um *nó raiz* que não possui arestas chegando e zero ou mais arestas saindo.

2. *Nós internos*, cada um possuindo exatamente uma aresta chegando e duas ou mais saindo.

3. *Nós folha ou terminal*, cada um possuindo exatamente uma aresta chegando e nenhuma saindo.

Em uma árvore de decisão, cada *nó folha ou terminal* recebe um rótulo de classe. Os *nós não terminais*, que incluem o *nó raiz* e outros *nós internos*, contêm condições de testes de atributo para separar registros que possuam características diferentes.

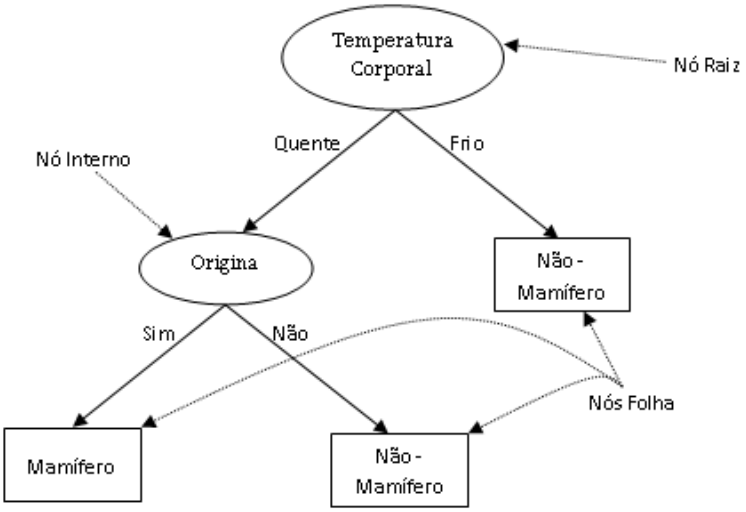

Figura 4.5. Uma árvore de decisão para o problema de classificação de vertebrados. Fonte: [Tan et al., 2009].

Um algoritmo de árvore de decisão cresce de uma forma recursiva pelo particionamento dos registros em sucessivos subconjuntos mais puros. Suponha que D_t seja o conjunto de registros de treino que estão associados ao nó t e $y = \{y_1, y_2, ..., y_c\}$ sejam os rótulos das classes. A seguir é feita uma definição recursiva do algoritmo:

- *Passo 1:* Se todos os registros em D_t pertencerem à mesma classe y_t então t é um nó folha rotulado como y_t.

- *Passo 2:* Se D_t contiver registros que pertençam a mais de uma classe, uma *condição de teste de atributo* é selecionada para particionar os registros em subconjuntos menores. Um nó filho é criado para cada resultado da condição de teste e os registros de D_t são distribuídos para os filhos baseados nos resultados. O algoritmo é então aplicado recursivamente a cada nó filho.

Um algoritmo de aprendizagem para induzir uma árvore de decisão, a cada iteração, deve selecionar um atributo *(condição de teste de atributo)* que melhor divida os registro e para isso são definidas algumas métricas. Estas métricas são consideradas em termos da distribuição da classe dos registros antes e depois da divisão.

As métricas desenvolvidas para selecionar a melhor divisão são muitas vezes baseadas no grau de impureza dos nós filhos. Quanto menor o grau de impureza, mais distorcida é a distribuição das classes. Por exemplo, um nó com uma distribuição de classe (0;1) possui impureza zero, enquanto que um nó com distribuição de classe uniforme (0,5;0,5) possui a maior impureza. Exemplos de métricas de impureza incluem:

$$Entropia(t) = -\sum_{i=0}^{c-1} p(i \setminus t) log_2 p(i \setminus t) \tag{4.2}$$

$$Gini(t) = 1 - \sum_{i=0}^{c-1} [p(i \setminus t)]^2 \tag{4.3}$$

$$Erro\,de\,Classificacao(t) = 1 - max_i[p(i \setminus t)]^2 \tag{4.4}$$

Onde $p(i \setminus t)$ denota a fração de registros que pertencem à classe i em um determinado nó t. A *Entropia* possui uma escala que varia entre 0 e 1, *Gini* e *Erro de Classificação* possuem uma escala que varia entre 0 e 0,5. Todas as três alcançam seu valor máximo quando a distribuição da classe é uniforme, ou seja, não é possível diferenciar uma classe de outra. Os valores mínimos para as métricas são atingidos quando todos os registros pertencem à mesma classe. A cada iteração, o algoritmo irá selecionar o atributo que possua o menor valor obtido pela métrica.

4.5.2 Classificador Baseado em Regras

Um classificador baseado em regras é uma técnica para classificar registros usando um conjunto de regras "se ... então". As regras para o modelo são representadas na forma $R = (r_1 \vee r_2 \vee ...r_k)$, onde R é conhecido como o *conjunto de regras* e os r_i's são as regras de classificação. Cada regra pode ser expressa da seguinte maneira:

$$r_1 : (Condicao) \rightarrow y_i \tag{4.5}$$

O lado esquerdo da regra é chamado de *antecedente da regra* ou *pré-condição*. Ele contém um conjunto de testes de atributo:

$$Condicao_i = (A_1 \; op \; v_1) \vee (A_2 \; op \; v_2) \vee ...(A_k \; op \; v_k) \tag{4.6}$$

Onde (A_j, v_j) é um par atributo-valor e op é um operador lógico escolhido do conjunto $\{=, \neq, <, >, \leq, \geq\}$. Cada teste de atributo $\{A_j \; op \; v_j\}$ é conhecido como um conjunto. O lado direito da regra é chamado de *consequência da regra*, que contém a classe y_i prevista. Um regra r cobre um registro x se a pré-condição de r corresponder aos atributos de x. r também é dita ser disparada sempre que cobrir um determinado registro.

A qualidade de uma regra de classificação pode ser avaliada usando-se medidas como a *cobertura* e a *precisão*. Dado um conjunto de dados D e uma regra de classificação $r : A \rightarrow y$, a cobertura da regra é definida como a fração de registros em D que disparam a regra r. Por outro lado, sua precisão ou fator de confiança é definida como a fração de registros disparados por r cujos rótulos de classe sejam iguais a y. As definições formais dessas medidas são:

$$Cobertura(r) = \frac{|A|}{|D|} \tag{4.7}$$

$$Precisão(r) = \frac{|A \cap y|}{|A|} \tag{4.8}$$

Onde $|A|$ é o número de registros que satisfazem ao antecedente da regra, $|A \cap y|$ é o número de registros que satisfazem tanto ao antecedente quanto ao consequente e $|D|$ é o número total de registros.

Um classificador baseado em regras classifica um registro de teste baseado na regra disparada pelo registro. Duas são as propriedades importantes do conjunto de regras gerado por um classificador: (1)*Regras Mutuamente Excludentes* - não devem haver duas regras que sejam disparadas pelo mesmo registro; (2) *Regras Completas* - cada

registro deve ser coberto por pelo menos uma regra. Muitos classificadores baseado em regras não possuem tais propriedades. Neste caso, se um conjunto de regras não for completo uma regra padrão deve ser adicionada e disparada quanto todas as outras falham. A classe atribuída por essa regra padrão é conhecida como *classe padrão* que geralmente é a classe majoritária. Caso as regras não forem mutuamente excludentes, há duas formas de resolver o problema: (1) *Regras Ordenadas* - as regras são ordenadas de forma decrescente de prioridade, que pode ser de diversas maneiras, por exemplo, precisão, cobertura, ordem que foi gerada dentre outras; (2) *Regras Não Ordenadas* - permite que um registro dispare múltiplas regras de classificação e considera cada uma como um voto(ponderado ou não) para uma determinada classe, o registro recebe a classe que tiver o maior número de votos.

4.5.2.1 RIPPER

O *RIPPER* é um algoritmo que usa regras ordenadas. Para problemas de duas classes, o algoritmo escolhe a classe majoritária como sua classe padrão e descobre as regras para detectar a classe minoritária. Para problemas com múltiplas classes, estas são ordenadas de acordo com suas frequências. Suponha que $(y_1, y_2, ..., y_c)$ sejam as classes ordenadas, onde y_1 é aquela com menor frequência e y_c a de maior frequência. Durante a primeira iteração, instâncias que pertençam a y_1 são rotuladas como exemplos positivos, enquanto que aquelas que pertencerem a outras classes são rotuladas como exemplos negativos. Os exemplos considerados positivos são retirados do treino. A seguir, *RIPPER* extrai regras que distinguem y_2 das outras classes restantes. Este processo é repetido até que se tenha y_c, que é atribuída a classe padrão.

4.5.2.2 LAC

O *Lazy Associative Classification (LAC)* é um algoritmo desenvolvido por Veloso et al. e trabalha com regras não ordenadas. Ele explora o fato de que, frequentemente, existem fortes associações entre os pares de atributo-valor e as classes. Tais associações são geralmente escondidas nos dados de treinamento e quando descobertas podem revelar aspectos importantes que podem ser usados na previsão das classes [Veloso et al., 2006].

Basicamente, o algoritmo produz uma função de classificação composta pelas regras $X \rightarrow c_i$ que indicam a associação entre os pares de atributo-valor X e a classe c_i (por exemplo, classe de fraude (*chargeback*) e classe de não fraude). Denomina-se como R um conjunto arbitrário de regras. Da mesma forma, é denominado como R_{c_i} um subconjunto de R que é composto pelas regras na forma $X \rightarrow c_i$ (ou seja, regras

que preveem a classe c_i). Uma regra $X \rightarrow c_i$ é dita associada a um registro x se $X \subseteq x$ (ou seja, se o registro x contém todos os pares atributo-valor de X) e essa regra é incluída em $R^x_{c_i}$. Sendo assim, $R^x_{c_i}$ é composto pelas regras que preveem a classe c_i e estão associadas ao registro x. Obviamente $R^x_{c_i} \subseteq R_{c_i} \subseteq R$.

O LAC aprende a função de classificação em duas etapas principais:

- Extração de Regra Sob Demanda: Para evitar explosão de regras o algoritmo as extrai sob demanda, no tempo de aprendizagem, a partir dos dados de treinamento. Ele projeta o espaço de busca para as regras de acordo com as informações dos registros no conjunto de teste permitindo a extração de regras com eficiência. Em outras palavras, o LAC projeta/filtra os dados de treinamento de acordo com os pares atributo-valor do registro x no conjunto de teste e extrai as regras dos dados de treinamento projetados que é indicado como D^x. Isso garante que somente as regras que carregam a informação sobre o registro x são extraídas dos dados de treinamento delimitando drasticamente o número de regras possíveis.

- Previsão: Naturalmente, existe uma ordenação total entre as regras no sentido de que algumas mostram associações mais fortes do que outras. Uma estatística amplamente utilizada chamada confiança (denotada como $\theta(X \rightarrow c_i)$) mede a força da associação entre X e c_i. De forma simples, a confiança da regra $X \rightarrow c_i$ é dada pela probabilidade condicional de c_i ser da classe do registro x dado que $X \subseteq x$.

  Utilizar uma regra única para prever a classe correta pode ser propenso a erros. Em vez disso, a probabilidade de c_i ser a classe do registro x é estimada através da combinação das regras em $R^x_{c_i}$. Mais especificamente, $R^x_{c_i}$ é interpretado como uma enquete, em que cada regra $X \rightarrow c_i \in R^x_{c_i}$ é um voto dado pelas características em X para a classe c_i. O peso de um voto $X \rightarrow c_i$ depende da força da associação entre X e c_i que é dado por $\theta(X \rightarrow c_i)$. O processo de estimar a probabilidade de c_i ser a classe do registro x começa com a soma dos pesos dos votos para c_i e então a média é obtida dividindo o número total de votos para c_i, conforme expresso pela função de pontuação *score function* $s(c_i, x)$ mostrada na Equação 4.9 (onde $r_j \subseteq R^x_{c_i}$ e $|R^x_{c_i}|$ é o número de regras em $R^x_{c_i}$). Assim, $s(c_i, x)$ fornece a confiança média das regras em $R^x_{c_i}$ (obviamente, quanto maior a confiança mais forte a evidência de pertencer à classe).

$$s(c_i, x) = \frac{\sum_{j=1}^{|R^x_{c_i}|} \theta(r_j)}{|R^x_{c_i}|} \tag{4.9}$$

A probabilidade estimada de c_i ser a classe do registro x, denotado como $\hat{p}(c_i|x)$, é obtida normalizando $s(c_i, x)$, como mostrado na Equação 4.10. Um valor mais alto de $\hat{p}(c_i|x)$ indica uma maior probabilidade de c_i ser a classe para o registro x. A classe associada com a maior probabilidade é, finalmente, prevista como a classe para o registro x.

$$\hat{p}(c_i|x) = \frac{s(c_i, x)}{\sum_{j=1}^{n} s(c_i, x)} \tag{4.10}$$

4.5.3 Naive Bayes

Suponha que $\mathbf{X}$ denote o conjunto de atributos e Y denote a variável de classe. Se esta tiver um relacionamento não determinístico com os atributos, então pode-se tratar $\mathbf{X}$ e Y como variáveis aleatórias e capturar seu relacionamento usando probabilisticamente $P(Y|\mathbf{X})$. Esta probabilidade condicional também é conhecida como *probabilidade posterior* de Y, em oposição à sua *probabilidade anterior*, $P(Y)$.

Durante a fase de treinamento, é preciso descobrir as probabilidades posteriores $P(Y|\mathbf{X})$ para cada combinação de $\mathbf{X}$ e Y baseada em informações coletadas a partir dos dados de treinamento. Conhecendo estas probabilidades, um registro de teste $\mathbf{X}'$ pode ser classificado encontrando-se a classe Y' que maximize a probabilidade posterior, $P(Y'|\mathbf{X}')$.

Em muitas aplicações, o relacionamento entre o conjunto de atributos e a classe é não determinístico. Ou seja, o rótulo da classe de um registro de teste pode não ser previsto com certeza embora seu conjunto de atributos sejam idênticos a alguns dos exemplos de treinamento. O classificador *Naive Bayes* avalia a probabilidade condicional da classe supondo que os atributos sejam condicionalmente independentes, dado o rótulo de classe y. A suposição de *independência condicional* pode ser declarada formalmente da seguinte maneira:

$$P(\mathbf{X}|Y = y) = \prod_{i=1}^{d} P(X_i|Y = y) \tag{4.11}$$

Onde cada conjunto de atributos $\mathbf{X} = \{X_1, X_2, ..., X_d\}$ consiste de d atributos. Com a suposição da independência condicional, em vez de se calcular a probabilidade condicional de classe para cada combinação de $\mathbf{X}$, apenas tem-se que estimar a probabilidade condicional de cada X_i, dado Y. Para classificar um registro de teste, o classificador calcula a probabilidade posterior para cada classe Y.

$$P(Y|\mathbf{X}) = \frac{P(Y)\prod_{i=1}^{d} P(X_i|Y))}{P(\mathbf{X})} \tag{4.12}$$

Já que $P(\mathbf{X})$ é fixo para cada Y, basta escolher a classe que maximiza o termo numerador, $P(Y)\prod_{i=1}^{d} P(X_i|Y)$.

4.5.4 Meta Aprendizagem

Esta seção apresenta técnicas para melhorar a precisão de classificação agregando as previsões de múltiplos classificadores. Estas técnicas são conhecidas como *combinação de classificadores* ou *meta aprendizado*. O método constrói um conjunto de *classificadores básicos* a partir dos dados de treinamento e executa a classificação recebendo um voto sobre as previsões feitas por cada um dos classificadores básicos.

4.5.4.1 Boosting

Boosting é um procedimento iterativo usado para alterar adaptativamente a distribuição de exemplos de treinamento de modo que os classificadores de base enfoquem exemplos que sejam difíceis de classificar. Ele atribui um peso a cada exemplo de treinamento que podem ser usados das seguintes maneiras:

1. Podem ser usados como uma distribuição de amostras para desenhar um conjunto de amostras de *bootstrap* a partir dos dados originais.

2. Podem ser usados pelo classificador de base para descobrir um modelo que tenha tendência na direção de exemplos de pesos mais altos.

Esta seção descreve um algoritmo que usa pesos de exemplos para determinar a distribuição de amostragem do seu conjunto de treinamento. Inicialmente, os exemplos recebem valores iguais, de modo que tenham a mesma probabilidade de serem escolhidos para treinamento. Uma amostra é desenhada de acordo com a distribuição de amostras dos exemplos de treino para se obter um novo conjunto de treinamento. A seguir, um classificador é induzido a partir do conjunto de treino e usado para classificar todos os exemplos dos dados originais. Os pesos dos exemplos de treinamento são atualizados no final de cada rodada de *boosting*. Exemplos que estejam classificados incorretamente terão seus pesos aumentados, enquanto que aqueles que estiverem classificados corretamente terão seus pesos diminuídos. Isto força que o classificador, nas iterações subsequentes, enfoque em exemplos que sejam difíceis de classificar.

4.5.4.2 Stacking

Ao contrário de *boosting*, *stacking* não é normalmente usado para combinar os modelos de mesmo tipo, por exemplo, um conjunto de árvores de decisão. Em vez disso, é aplicado a modelos construídos por meio de algoritmos de aprendizagem diferentes. Suponha que exista um indutor de árvore de decisão, um classificador *Naive Bayes*, e um classificador baseado em regras e que se deseja criar um classificador para um conjunto de dados fornecido. O procedimento usual seria estimar o erro esperado de cada algoritmo utilizando validação cruzada e escolher o melhor modelo de previsão em dados futuros. Em vez disso, *stacking* combina a saída de todos os três algoritmos com o objetivo de melhorar a previsão.

Uma maneira de combinar saídas é por meio de um mecanismo de votação. No entanto, votação (não ponderada) só faz sentido se os sistemas de aprendizagem executam comparativamente bem. A classificação ficaria errada, se dois dos três classificadores fazerem previsões que são grosseiramente incorretas. Em vez disso, o *stacking* introduz o conceito de um *metalearner*, que substitui o processo de votação. O problema com a votação é que não está claro em qual classificador confiar. *Stacking* tenta aprender qual dos classificadores são os de confiança, usando outro algoritmo de aprendizagem - o *metalearner* - para descobrir a melhor forma de combinar a saída dos classificadores base.

A entrada para o metamodelo, também chamado de modelo de nível 1, são as previsões dos modelos base, ou modelos de nível 0. Uma instância de nível 1, tem tantos atributos quanto o número de classificadores de nível 0 e os valores dos atributos fornecem a previsão do classificador para a instância correspondente no nível 0. Quando um classificado *stacking* é usado para a classificação, uma instância é primeiro alimentada nos modelos de nível 0 e cada um prevê um valor de classe. Estas previsões são alimentadas no modelo de nível 1, que as combinam para a previsão final. A Figura 2.6 mostra um exemplo do modelo *stacking*.

4.5.5 Oversampling

Conjunto de dados com distribuições de classes desequilibradas são bastante comuns em muitas aplicações reais. Exemplo, na detecção de fraudes em cartão de crédito, transações fraudulentas ocorrem em menor número que as legítimas. Sendo assim, há um número desproporcional de instâncias que pertencem a classes diferentes. Apesar disso, uma classificação corretada da classe rara possui um valor maior do que uma classificação correta da classe majoritária. No entanto, a maioria dos classifica-

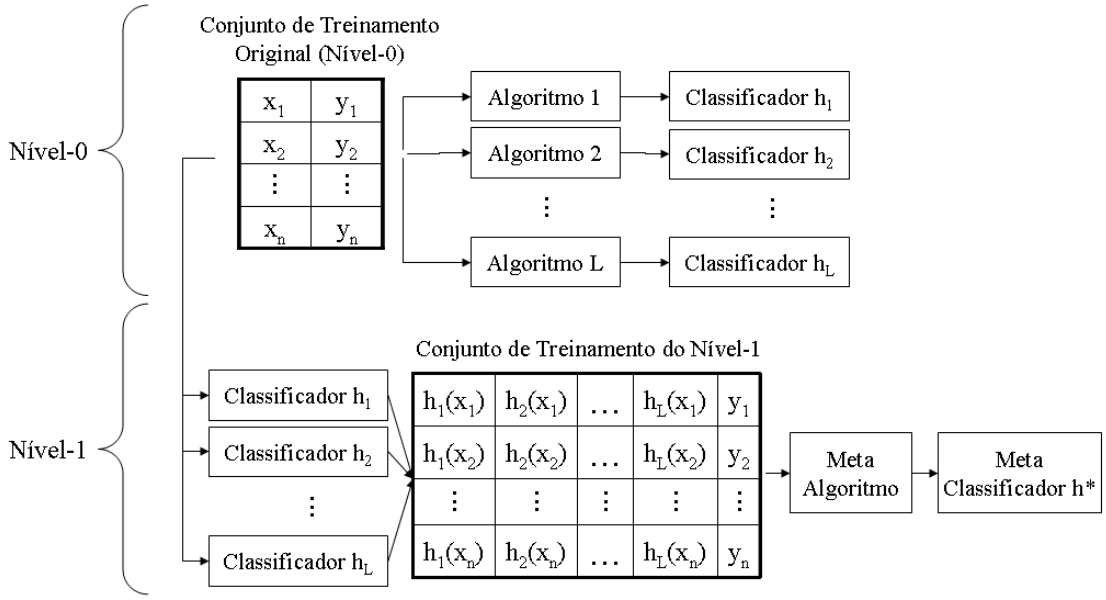

Figura 4.6. Modelo de Stacking. Fonte:[Maria Izabela R. Caffé, 2011].

dores tendem a focar a classificação nos exemplos majoritários o que pode causar a completa falha do classificador em relação ao objetivo desejado.

Oversampling é uma técnica simples para lidar com o desequilíbrio de classe. Ela consiste em sobre amostrar os exemplos das classes minoritárias nos dados de treinamento. Isto cria um conjunto de dados mais equilibrado, o que pode contribuir para a criação de um melhor modelo de classificação. A amostragem pode ser aleatória ou baseada em algum critério, como por exemplo, registros da classe minoritária que possuem maiores erros de classificação [Japkowicz & Stephen, 2002]. Entretanto, essa sobre amostragem pode resultar em um enorme problema, pois irá aumentar ainda mais o conjunto de dados o que pode fazer com que alguns algoritmos não consigam lidar com esse volume maior de informação.

4.6 Passo 5: Interpretação e Avaliação dos Resultados

Esta etapa consiste em interpretar e avaliar os padrões obtidos na mineração de dados. Os modelos gerados são analisados para verificar o quão bem eles atendem aos objetivos do processo. Este passo pode envolver também a visualização dos padrões extraídos, que permite que os analistas explorem os dados e os resultados da mineração, a partir de uma diversidade de pontos de vista. O usuário também pode definir medidas para avaliar os padrões extraídos baseado no conhecimento do domínio da aplicação

por exemplo, no caso da detecção de fraude pode-se criar medidas para calcular o montante de dinheiro economizado.

Algumas medidas de avaliação do modelo são: validação cruzada, matriz de confusão, precisão, revocação e medida F1. Outras medidas como acurácia e taxa de erro são utilizadas principalmente, para *datasets* que possuem uma distribuição relativamente semelhante das classe. No caso de detecção de fraude, as classes possuem uma distribuição desbalanceada sendo assim, serão descritas algumas das medidas para esse tipo de problema. Importante observar que precisão, revocação e medida F1 são calculadas por classe, neste trabalho, a classe a ser avaliada é a classe de fraude, sendo assim, essas medidas serão calculadas para essa classe.

1. *Validação Cruzada:* muitas vezes é útil medir o desempenho do modelo em um conjunto de dados de teste, uma vez que tal medição fornece uma avaliação imparcial do erro e da generalização. O erro estimado auxilia o algoritmo de aprendizagem a executar a seleção do modelo. O método de validação cruzada segmenta os dados em k partições de tamanho igual. Durante cada execução, uma das partições é escolhida para teste, enquanto que as outras são usadas para treinamento. Este procedimento é repetido k vezes de modo que cada partição seja usada para teste exatamente uma vez. O erro total é encontrado pela soma dos erros de todas as k execuções.

2. *Matriz de confusão:* avalia o desempenho de um modelo de classificação baseada na contagem de registros de testes previstos corretamente e incorretamente. Estas contagens são tabuladas em uma tabela que é conhecida como matriz de confusão. A Tabela 4.1 mostra a matriz de confusão para problemas de classificação binária na qual as classes não são igualmente importantes. Nesses casos, a classe rara muitas vezes denota a classe positiva, enquanto que a majoritária é denotada como classe negativa. A terminologia frequentemente usada é:

   a) *Verdadeiro Positivo (TP) ou* f_{++}*:* registros positivos previstos corretamente pelo modelo.

   b) *Falso Negativo (FN) ou* f_{+-}*:* registros positivos previstos erroneamente como negativos pelo modelo.

   c) *Falso Positivo (FP) ou* f_{-+}*:* registros negativos previstos erroneamente como positivos pelo modelo.

   d) *Verdadeiro Negativo (TN) ou* f_{--}*:* registros negativos previstos corretamente pelo modelo.

3. *Precisão e Revocação:* são duas métricas amplamente usadas em aplicações onde a detecção bem sucedida de uma classe é considerada mais significativa do que a de outra classe. A *precisão*, calculada conforme Equação 4.13, determina a fração de registros que realmente são positivos no grupo que o classificador declarou como classe positiva. A *revocação*, calculada conforme Equação 4.14, mede a fração de exemplos positivos previstos corretamente pelo classificador.

$$Precisao, p = \frac{TP}{TP + FP} \tag{4.13}$$

$$Revocacao, r = \frac{TP}{TP + FN} \tag{4.14}$$

4. *Medida F1:* A precisão e a revocação podem ser resumidas em outra métrica conhecida como *medida F1*. A princípio, essa medida representa a média harmônica entre precisão e revocação. Sendo assim, um valor alto da *medida F1* assegura que tanto a precisão quanto a revocação sejam razoavelmente altas. A Equação 4.15 representa sua definição formal:

$$F1 = \frac{2rp}{r + p} = \frac{2TP}{2TP + FP + FN} \tag{4.15}$$

Tabela 4.1. Matriz de confusão para um problema de classificação binária na qual as classes não são igualmente importantes

		Classe Prevista	
		+	-
Classe Real	+	f_{++} (TP)	f_{+-} (FN)
	-	f_{-+} (FP)	f_{--} (TN)

A visualização também é muito utilizada nesta etapa e, como já mencionado, consiste na exibição dos resultados na forma de gráficos e tabelas. O objetivo é transformar o conhecimento derivado em um formato que seja fácil para os humanos entenderem(como imagens ou gráficos) e em seguida, contar com a velocidade e capacidade altamente evoluída do sistema visual humano para perceber o que é interessante [Fayyad, 1996].

No contexto da fraude, existem perdas financeiras associadas a erros de classificação. Assim, não só a cobertura da fraude deve ser levada em consideração, mas também, a eficiência econômica do resultado. O conceito de *Eficiência Econômica* trata-se da avaliação financeira dos resultados obtidos. Ou seja, com a utilização de

um determinado modelo de detecção de fraude, foram obtidos maiores ganhos financeiros em relação à situação onde esse modelo não foi utilizado? Dessa forma, é proposta uma fórmula para que seja possível o cálculo da eficiência econômica dos resultados obtidos com os modelos de detecção de fraude. Esta fórmula é baseada na contabilização dos ganhos e das perdas financeiros de acordo com os estados definidos na *matriz de confusão*. O cálculo de Eficiência Econômica *(EE)* é apresentado na Equação 4.16.

$$EE = \sum_{j=1}^{n} G * gr - L * lr \tag{4.16}$$

Onde n representa o número de transações existentes, G representa o valor financeiro das transações verdadeiras positivas *(TP)*, gr é o percentual de ganho da empresa em uma transação bem sucedida, L é o valor financeiro das transações falsas negativas *(FN)* e lr é o percentual de perda da empresa em uma transação de fraude classificada como não fraude. Aplicando esta fórmula para a classificação fornecida pelas técnicas, é possível verificar o lucro ou prejuízo obtido em cada algoritmo.

Para que seja possível uma melhor avaliação do resultado obtido com a Eficiência Econômica *(EE)*, são definidos três limites a serem comparados. O primeiro limite é a Eficiência Econômica Máxima *(EE_ Max)*, onde o termo $(L*lr)$ da Equação 4.16 é igual a 0, que representa o ganho máximo a ser obtido pela empresa caso todas a fraudes sejam detectadas e nenhuma transação legítima seja considerada fraude. A Equação 4.17 representa esse cálculo. O segundo limite consiste na Eficiência Econômica Real *(EE_ Real)*, que consiste no cálculo da Eficiência Econômica (EE) sem a utilização de nenhuma das técnicas propostas considerando apenas a eficiência real existente. Neste caso, para que as técnicas tragam algum benefício econômico é necessário que seus resultados fiquem acima deste limite. O último limite consiste na Eficiência Econômica Mínima *(EE_ Min)*, onde o termo $(G*gr)$ da Equação 4.16 é igual a 0, que representa a pior situação possível, em que todas as transações legítimas foram consideradas fraudes e as transações de fraudes foram consideradas legítimas. A Equação 4.18 representa esse cálculo.

$$EE\_Max = \sum_{j=1}^{n} G * gr \tag{4.17}$$

$$EE\_Min = \sum_{j=1}^{n} -L * lr \tag{4.18}$$

Segundo Júnior et al. [2011], há uma relação de custo benefício entre cobrir todas as fraudes e ao mesmo tempo bloquear transações legítimas. Cada vez que uma fraude

é detectada há uma vantagem financeira associada a essa detecção, já que não existe nenhum prejuízo nessa situação. No entanto, existe uma perda financeira cada vez que uma transação legítima é rotulada como transação de fraude. Além do mais, isso também provoca uma insatisfação do cliente já que este não conseguiu realizar a compra desejada. Sendo assim, a Equação 4.16 deve ser adaptada para considerar uma penalização nos casos em que transações legítimas são erroneamente classificadas como ilegítimas. Esta equação é chamada de Eficiência Econômica com Penalização *(EE_ CP)* e é apresentada pela Equação 4.19.

$$EE\_CP = \sum_{j=1}^{n} G * gr - L * lr + NG * p \tag{4.19}$$

Onde NG representa o valor dos registros legítimos classificados como fraude ou seja, os falsos positivos *(FP)* da *matriz de confusão*. p representa o percentual desse valor que será considerado como perda. Para avaliar o percentual de ganho obtido em relação ao ganho máximo possível deve-se utilizar Equação 4.20 e Equação 4.21. Elas fornecem o ganho relativo em que 100% representa o ganho máximo possível *(EE_ Max)* e 0% o cenário real *(EE_ Real)*. A Equação 4.20 representa o ganho alcançado considerando a Eficiência Econômica sem penalidade *(EE)* e a Equação 4.21 representa o ganho alcançado considerando a Eficiência Econômica com penalidade *(EE_ CP)*.

$$Ganho\_SP = \frac{EE - EE\_Real}{EE\_Max - EE\_Real} \tag{4.20}$$

$$Ganho\_CP = \frac{EE\_CP - EE\_Real}{EE\_Max - EE\_Real} \tag{4.21}$$

Assim como as ferramentas de pontuação de risco descritas no Capítulo 2, os classificadores selecionados fornecem como resultado a probabilidade de um registro ser fraudulento ou não. Sendo assim, como forma de maximizar os resultados obtidos com o classificador, é proposta a geração de um *ranking* considerando a probabilidade do registro ser fraudulento. Dessa forma, a Eficiência Econômica deverá ser analisada ao longo do *ranking* de forma a encontrar um ponto onde os resultados são maximizados. Essa abordagem se mostra interessante, uma vez que os classificadores podem considerar um registro fraudulento como não fraudulento, entretanto a probabilidade de ser fraudulento pode ser relativamente grande. Utilizando o *rank* esse problema pode ser atenuado. A Figura 4.9, mostra o resultado de um classificador para um *dataset* contendo 20 registros em que 5 deles são fraudulentos. Perceba que pela tabela da esquerda nenhum dos casos fraudulentos foi classificado como fraude, uma vez que sua probabilidade de não fraude é maior. No entanto, ordenando-se de forma decres-

cente pela probabilidade de fraude, percebe-se que existe um ponto no *ranking* onde as fraudes são 100% cobertas. Dessa forma, possibilitou-se maximizar os resultados do classificador. A Figura 4.7 mostra em cinza a região do gráfico onde espera-se que os resultados dos classificadores estejam para que seja possível considerar um ganho financeiro.

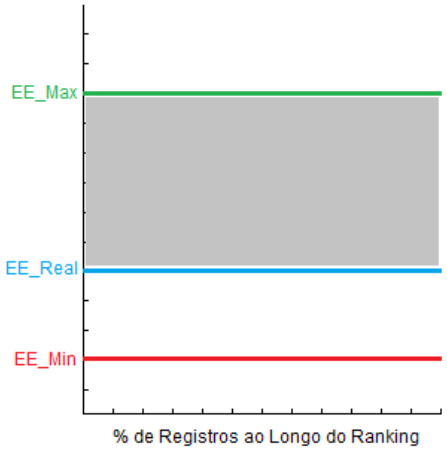

Figura 4.7. Eficiência econômica ao longo do *ranking*. A região em cinza indica que ganhos são obtidos em relação aos ganhos reais.

A Figura 4.8 sintetiza o processo de avaliação de um classificador considerando a abordagem proposta. Com isso, espera-se tanto avaliar o modelo gerado baseado nas medidas já existentes, quanto verificar o retorno financeiro obtido com a técnica.

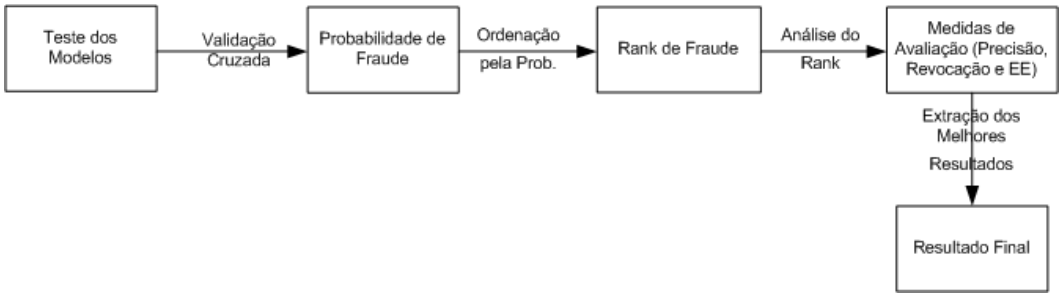

Figura 4.8. Metodologia para avaliação dos classificadores.

4.7 Considerações Finais

Este capítulo apresentou uma metodologia para detecção de fraude baseada no processo de Descoberta de Conhecimento em Banco de Dados. Essa metodologia é

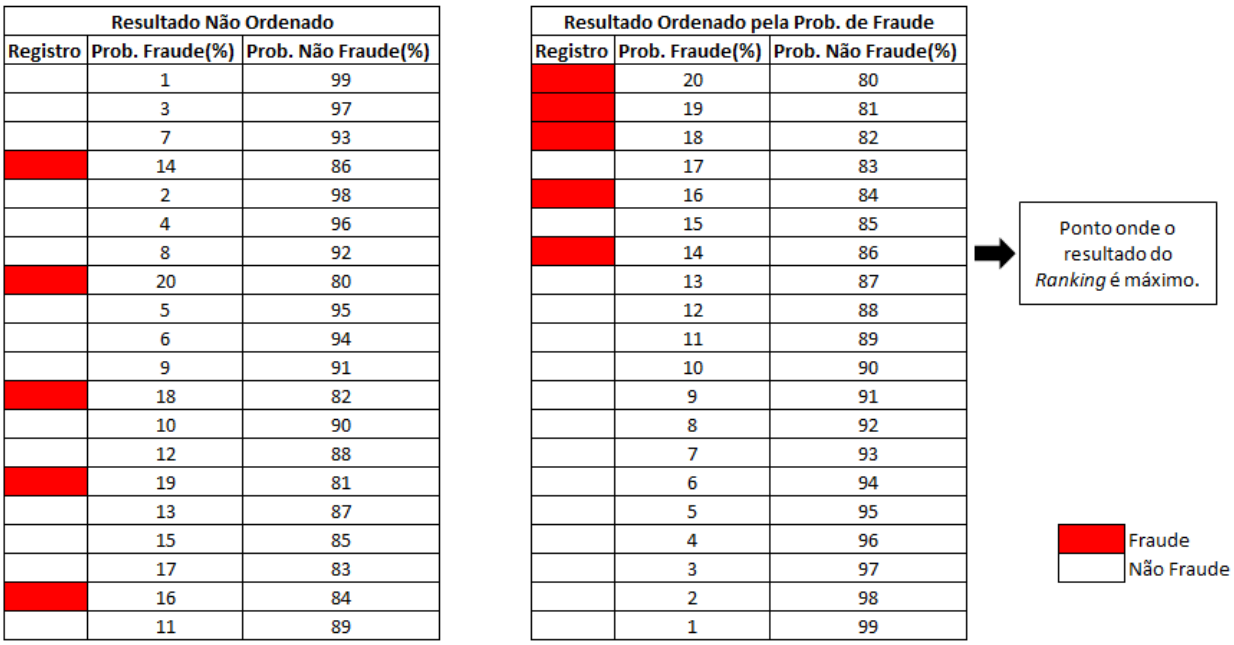

Figura 4.9. Geração do *ranking* a partir dos resultados de um classificador

composta por cinco passos com o objetivo de auxiliar o processo de detecção de fraude desde a obtenção dos dados para a formação de um *dataset*, passando pela escolha de técnicas mais promissoras, até a avaliação dos resultados obtidos. Em cada uma dos passos são apresentadas metodologias e/ou orientações para sua realização, considerando o contexto de fraude, fornecendo uma boa orientação para quem deseja realizar pesquisas nessa área. O próximo capítulo apresentará um estudo de caso onde a metodologia aqui apresentada será instanciada para a realização do processo de detecção de fraude em um conjunto de dados real de uma das maiores empresas no Brasil de serviços de pagamentos eletrônicos o *UOL PagSeguro*.

Capítulo 5

Estudo de Caso - UOL PagSeguro

Os capítulos anteriores permitiram uma ampla compreensão dos conceitos abordados nesta pesquisa. As fraudes em transações com cartão de crédito foram extensamente discutidas, tanto em questões de como estas são realizadas quanto como elas são combatidas. Uma metodologia para detecção de fraude e as técnicas utilizadas foram apresentadas e descritas. Este capítulo apresenta um estudo de caso usando dados reais, onde a metodologia proposta é instanciada com o objetivo de auxiliar na detecção de fraudes em transações eletrônicas, mais especificamente nas operações de cartão de crédito para identificação de *chargeback*.

5.1 Visão Geral

No Capítulo 2 foi apresentado o conceito de *chargeback* e como ele afeta cada uma das partes envolvidas na transação. Dessa forma, devido aos prejuízos causados pelas fraudes, surgiram na *Web* empresas que buscam fornecer soluções para pagamentos *online* com o objetivo de tornar as transações de compra e venda na Internet mais seguras. Essas empresas trabalham como um intermediário das transações e assumem os riscos associados, obviamente, cobrando por esse serviço. Elas asseguram que tanto a compra quanto a venda serão realizadas com sucesso, caso contrário, o cliente ou o vendedor serão reembolsados. Pode-se perceber que tanto o comprador quanto o vendedor transferem todo o risco da transação para essas empresas, mostrando ser esse um serviço de altíssimo risco. Dessa forma, investimentos em tecnologias para detecção de fraudes na *Web* são de extrema importância para essas empresas. Para avaliar a eficiência da metodologia proposta, este trabalho utiliza como cenário uma das maiores empresas no Brasil no serviço de pagamentos eletrônicos, o *UOL PagSeguro*[1].

[1] https://pagseguro.uol.com.br/

De acordo com o seu site, o *UOL PagSeguro* é uma empresa de soluções em pagamentos *online* que busca garantir a segurança de quem compra e de quem vende na *Web*. Quem compra tem a garantia de produto ou serviço entregue ou o dinheiro de volta. Quem vende fica livre de fraudes e perdas em vendas online. A Figura 5.1 mostra a página inicial do site do *UOL PagSeguro*.

Figura 5.1. Página inicial do site do *UOL PagSeguro*.

Como um breve histórico, pode-se citar que a empresa surgiu inicialmente em janeiro de 2007 quando o *UOL* adquiriu a *BRpay*. Em abril de 2007, a *BRpay* foi escolhida pela *InfoExame* como a melhor solução para pagamentos *online*. Em julho de 2007, o *UOL* lançou o *PagSeguro*, dobrando a oferta de meios de pagamento, acrescentando parcelamento em todos os cartões, diminuindo as taxas e adicionando a central de atendimento telefônico. Terminou 2007 com pouco mais de 100 parceiros de desenvolvimento, 8 mil lojas e 1 milhão de compradores. Em novembro de 2008 o *PagSeguro* foi escolhido pelos leitores da revista *InfoExame* como o site do ano na categoria comércio eletrônico, e terminou o ano com mais de 500 parceiros de desenvolvimento, 20 mil lojas ativas e 3 milhões de compradores.

Esta pesquisa utiliza o cenário do *UOL PagSeguro* para avaliação da abordagem proposta. Inicialmente, o objetivo seria a utilização de técnicas de mineração de dados para auxiliar na identificação de fraude. No entanto, para realização dessa tarefa foi necessário um esforço muito maior do que o esperado de entendimento do problema e preparação de uma base de dados com atributos relevantes. Isso fez com que a pesquisa se tornasse ainda mais abrangente envolvendo conhecimentos que vão além das técnicas

de mineração de dados. Dessa forma, o processo de descoberta do conhecimento, descrito na Seção 4.1, foi base fundamental para desenvolvimento de uma metodologia para a resolução desse problema. Essa metodologia lida com o problema de uma forma ampla, uma vez que suas abordagens e técnicas permitem resolver questões que vão desde a criação de uma base da dados, passando pela mineração de dados, até a avaliação dos resultados obtidos. As próximas seções irão descrever cada uma das tarefas associadas às etapas da metodologia e como elas foram realizadas para cumprir com seus objetivos.

5.2 Seleção dos Dados

Nesta etapa os dados que serão usados para detecção de fraude são selecionados para que sejam utilizados pelas técnicas de mineração de dados. Entretanto, é necessária uma compreensão do domínio do problema e uma análise e entendimento dos dados para seleção dos melhores atributos. O objetivo geral do processo já se encontra definido que consiste em fazer uso das abordagens e técnicas definidas pela metodologia de detecção de fraude para auxiliar na identificação das transações que podem causar *chargeback*. Definido o objetivo geral, uma ampla pesquisa na área de fraudes em pagamentos *online* foi realizada para compreensão do domínio do problema. Isso foi apresentado nos Capítulos 2 e 3, que descrevem todo o conhecimento adquirido para detecção de fraude. Além do mais, foram fornecidas pelo *UOL PagSeguro* documentações utilizadas no treinamento dos responsáveis pela análise manual do risco das transações. Todo esse conhecimento acumulado permitiu uma preparação para a escolha das informações que serão relevantes na identificação de fraude.

Um entendimento dos dados também é necessário e isso envolve a compreensão da fonte de dados e ferramentas para sua manipulação. É importante também o conhecimento da informação semântica fornecida pelos domínios, tipos, faixas de valores dos atributos e as relações entre eles. Para análise dos dados, o *UOL PagSeguro* disponibilizou uma base no formato *DUMP* do *Oracle*[2] com um tamanho de 32 *GigaBytes (GB)*. Essa base compreende o período entre janeiro de 2006 a início de janeiro de 2011 e foi extraída da instância de *Quality Assurance (QA)* que é a instância utilizada para validação final do sistema, onde, teoricamente, os dados estarão semelhantes aos dados de produção com pouca ou nenhuma perda de realismo. Para importação desse *DUMP* é necessária a instalação de um gerenciador de banco de dados *Oracle* que pode ser

[2]http://www.oracle.com.br

obtido pelo site da empresa desde que usado para fins acadêmicos. A versão obtida foi a *10g*, que utiliza uma menor quantidade de recursos da máquina.

A importação do *DUMP* durou aproximadamente 8 horas e a base de dados extraída possuía centenas de tabelas e milhares de registros. Importante ressaltar que, informações altamente confidenciais, como por exemplo, número do cartão, CPF, ou até mesmo e-mail são criptografados pelo *UOL PagSeguro* de forma que os dados fornecidos preservam a segurança de quem utiliza esse serviço. Além do mais, termos de confidencialidade devem ser assinados por todos os envolvidos na pesquisa que lidam diretamente com os dados. Isso garante ainda mais a segurança da informação existentes no *UOL PagSeguro*, já que além da cobertura técnica existe uma cobertura jurídica.

Para entendimento do banco de dados, o *UOL PagSeguro* forneceu um documento com a descrição completa das tabelas e atributos, bem como, do domínio dos atributos. Além do mais, foi fornecido o modelo de entidade e relacionamento do banco de dados no formato *pdf*. Entretanto, a navegação entre as entidades e seus relacionamentos era difícil, o que comprometia o entendimento dos dados. Sendo assim, foi utilizado o software *Power Designer* em sua versão *15.1* desenvolvido pela empresa *Sybase*[3]. Por meio desse software, foi feita a engenharia reversa do banco de dados gerando um modelo entidade-relacionamento no formato do *Power Designer*. Esse modelo permitiu uma maior agilidade na manipulação das tabelas e seus relacionamentos, facilitando o entendimento dos dados.

Este trabalho de compreensão da informação fornecida, se estendeu por meses onde várias questões foram levantadas e enviadas para especialistas do *UOL PagSeguro*. Após o entendimento desses dados, foi utilizada a metodologia de extração da informação do banco de dados definida no Capítulo 4. A Figura 5.2 ilustra a instanciação da metodologia para o banco de dados *Oracle* existente no *UOL PagSeguro*.

Conforme sugerido no *passo 1* da Seção 4.2, foi selecionado um conjunto de tabelas que contêm informações que auxiliassem na detecção de fraude. Esta seleção também contou com a ajuda de especialistas do *UOL PagSeguro* e reduziu o problema de seleção para algumas dezenas de tabelas.

Passando para o *passo 2*, as tabelas selecionadas até então, foram agrupadas para formação de um *dataset*. Esse processo de agrupamento consistiu em criar consultas *SQL* que permitam a união das tabelas, por meio de junções, formando os grupos. Neste caso, os grupos definidos foram:

1. *Transação:* contém informações de tabelas ligadas diretamente à tabela que ar-

[3]http://www.sybase.com.br

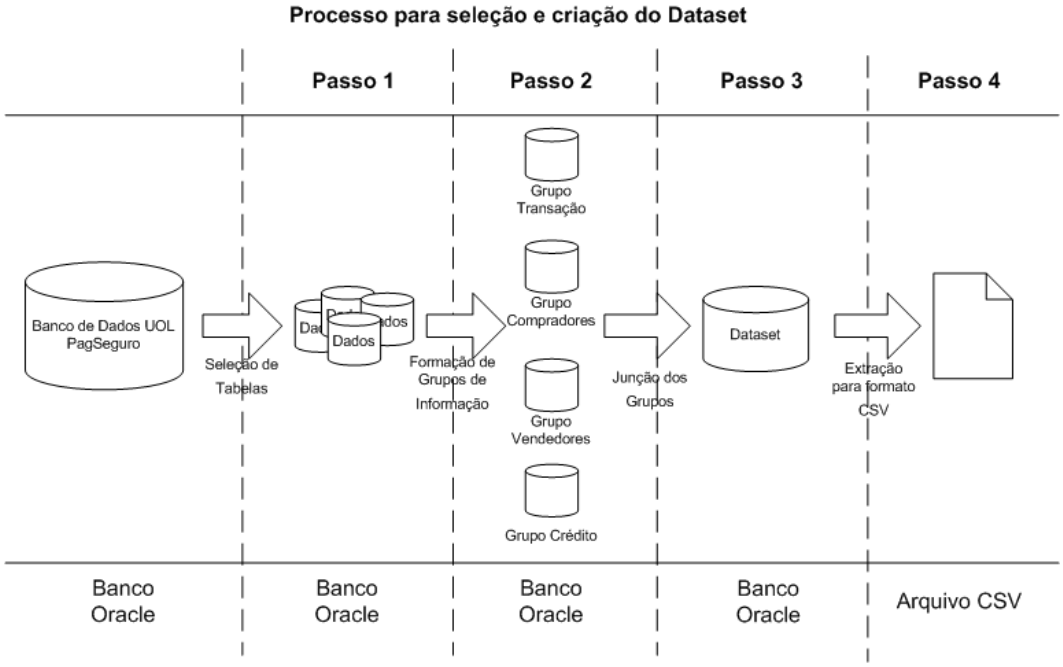

Figura 5.2. Processo de seleção dos dados.

mazena informações sobre a transação efetuada no *UOL PagSeguro*.

2. *Compradores:* contém informações sobre os usuários que atuaram como compradores na transação.

3. *Vendedores:* contém informações sobre os usuários que atuaram como vendedores na transação.

4. *Crédito:* contém informações sobre os cartões de crédito utilizados nas transações ou informações relacionadas ao pagamento.

Em cada grupo foi definido uma *tabela base* e foi criado um documento que mapeia cada uma das tabelas do grupo e quais os campos deverão ser utilizados para uní-las. Foi criado também um modelo entidade-relacionamento contendo apenas as tabelas selecionadas. Estes documentos serviram de referência para a realização das junções. O processo de auditoria dos dados definido no Capítulo 4 foi utilizado como forma de garantir que essas uniões não introduzissem erros no *dataset*.

Concluídas as junções de cada um dos grupos, os scripts gerados foram utilizados na criação de *views*. Sendo assim, foram criadas 4 *views* correspondendo a cada um dos grupos definidos. Seguindo para o *passo 4* os grupos criados foram unidos em um único *dataset*. Para isso foi utilizado o mesmo processo de auditoria mostrado na Figura 4.4. Neste caso, a *tabela base* será a *view* transação e o script gerado para junção

será utilizado para criar a *view* que conterá o *dataset* completo. O passo final, *passo 5*, consiste em extrair o *dataset* criado para um arquivo texto no formato *CSV*. Para isso foi utilizado o comando *SPOOL* do *Oracle* que permite a extração dos resultados de uma consulta para um arquivo texto, maiores detalhes sobre esse comando pode ser encontrado em Thomas [2009].

Concluída a extração do *dataset*, optou-se por realizar uma análise inicial das informações extraídas para verificar se estão adequadas para os passos seguintes. Um dos principais motivadores para a análise inicial da qualidade dos dados deve-se ao fato de que o *DUMP* fornecido não ter sido obtido diretamente dos dados de produção. Isso poderá propiciar a existência de muitas informações não legítimas e altamente inconsistentes inviabilizando qualquer trabalho de mineração de dados. A realização dessa análise teve como base a verificação de atributos com relação a valores nulos e inconsistentes. Os dados também foram contabilizados com base em determinados atributos, como, por exemplo, número de casos de *chargeback* e *não chargeback*. Foram verificados também máximos e mínimos de determinados atributos. As estatísticas extraídas dessa análise não serão apresentadas devido à confidencialidade das informações extraídas. Por meio dessa análise, pôde-se verificar várias inconsistências nos dados. Entretanto, o que mais despertou a atenção foi a frequência extremamente baixa dos casos de *chargeback*, muito abaixo do esperado e se concentrando nos 3 meses finais do *dataset*. Pode-se perceber então que grande parte dos dados não poderia ser utilizada, talvez apenas os 3 meses finais pois mostravam-se relativamente consistentes.

Devido ao problema de qualidade dos dados, optou-se por não utilizá-los oficialmente para desenvolvimento da pesquisa, no entanto, eles podem ser úteis para alguns testes preliminares. Dessa forma, foi verificado junto ao *UOL PagSeguro* a possibilidade de extração das informações desejadas diretamente do banco de produção. Para que isso fosse possível, era necessário o desenvolvimento de uma solução de extração que não comprometesse significativamente o desempenho do banco e que não extraísse dados sigilosos que deveriam passar por alguma criptografia. A solução proposta fez uso de toda a metodologia descrita, onde foi gerado um script único que realiza todos os passos da metodologia. Inicialmente, ele cria as *views* de transação, comprador, vendedor, crédito e em seguida a *view* que une todas a 4 anteriores. O próximo passo é a extração dos dados para um arquivo no formato *CSV*, que é feito via comando *SPOOL*. Para atender ao requisito de não onerar o banco de produção o script foi parametrizado de forma que o volume de dados a serem extraídos pode ser definido pelo usuário. Para reduzir ainda mais a necessidade de uso do servidor de banco de dados de produção, as *views* poderiam ser criadas em um outro banco da rede do *UOL PagSeguro* e os dados do banco de produção poderiam ser acessados via *DBLINK*. De uma forma resumida,

DBLINK permite que dois bancos de dados se comuniquem de forma que suas tabelas possam ser compartilhadas. Os atributos selecionados não fazem uso direto de dados que requerem criptografia, maiores detalhes serão vistos nas etapas posteriores uma vez que várias tarefas dessas etapas foram antecipadas devido às restrições impostas para obtenção dos dados.

O script foi fornecido ao *UOL PagSeguro*, eles optaram por realizar extrações mensais dos dados e, dessa forma, não comprometer o desempenho do banco de produção. O período extraído foi de agosto de 2010 a julho de 2011 separados em arquivos no formato *CSV* mensais, cada mês fornecido possui milhões de registros. Novamente, uma análise da consistência dos dados foi feita, conforme descrita anteriormente e os mesmos se mostraram adequados para a realização das próximas etapas.

Como já dito anteriormente, uma consideração importante a ser feita é que o objetivo geral do processo para detecção de fraude foi definido que é fazer uso de suas abordagens e técnicas para auxiliar na identificação das transações que podem causar *chargeback*. No entanto, a detecção de fraude está associada a duas vertentes. Essas vertentes consistem na identificação de padrões intra ou inter-transações. A identificação de padrões intra-transações tem como objetivo classificar uma transação como fraudulenta ou não, baseada em seus atributos. O padrão inter-transações consiste em analisar uma sequência de transações e a partir disso identificar a existência de fraude. Esta pesquisa trabalha com a identificação de fraude na busca de padrões intra-transações, ou seja, dado como entrada uma transação com um conjunto de atributos, um modelo deverá classificar essa transação como fraudulenta *(chargeback)* ou não. Grande parte desse foco se deve às características dos dados obtidos, além do mais, as duas vertentes existentes podem envolver estudos diferentes, o que ampliaria ainda mais o universo de pesquisa.

5.3 Pré-Processamento

Como dito no Capítulo 4 e observado neste capítulo na etapa de seleção, não é realista esperar que os dados estejam perfeitos. Podem existir vários problemas como valores faltando ou ilegítimos, objetos duplicados ou não representativos do problema. Nesta etapa, o *dataset* gerado será analisado com o objetivo de se realizar uma limpeza nos dados de forma a eliminar ruídos. Para isso, estratégias de visualização como gráficos de dispersão e histogramas serão utilizados. Serão feitas algumas caracterizações dos dados considerando valores máximos, mínimos e médios. Várias análises foram feitas no *dataset*, entretanto, devido à confidencialidade das informações, somente al-

gumas serão mostradas. Foram selecionadas aquelas em que é possível a sua exibição sem que se comprometa o sigilo dos dados, além do mais, apenas valores relativos serão considerados. As ferramentas utilizadas nesse processo foram:

1. *Weka:* ferramenta desenvolvida em Java que, segundo Hall et al. [2009], consiste em uma coleção de algoritmos de aprendizado de máquina para tarefas de mineração de dados. Os algoritmos podem ser aplicados diretamente a um conjunto de dados ou chamados a partir de um código Java. Weka contém ferramentas para pré-processamento, classificação, regressão, agrupamento, regras de associação e visualização.

2. *AWK:* linguagem para manipulação de arquivos de texto. Nessa etapa foi utilizado para fazer a consolidação dos dados para geração dos gráficos.

3. *Planilhas eletrônicas:* usada para agrupamento e organização das informações coletadas e também geração de gráficos.

4. *GNUPlot*[4]*:* consiste em um utilitário em linha de comando para plotagem de gráficos.

Como primeira análise, alguns atributos foram escolhidos e os registros agrupados em relação ao atributo escolhido. Foi verificado que a transação pode assumir dezenas de estados possíveis, entretanto, para essa pesquisa, apenas os estados *chargeback* e *concluído com sucesso* serão utilizados. Sendo assim, um número significativo de registros puderam ser eliminados. Verificando o atributo de meio de pagamento, percebe-se que o *UOL PagSeguro* permite várias formas de pagamento. No entanto, apenas podem gerar *chargeback* as transações em que o meio de pagamento é o cartão de crédito, o que permite mais algumas eliminações de registros que não se adequam aos objetivos desta avaliação.

Para geração dos gráficos de dispersão foi utilizado o Weka e o objetivo é identificar anomalias nos valores dos dados e também padrões para os dados de *chargeback* e *não chargeback*. Em todos os gráficos o eixo X representa o *status* da transação, onde *chargeback* é representado pela cor vermelha e *não chargeback* pela cor azul e isso não sofrerá variação. O eixo Y irá variar de acordo com o campo escolhido para análise da dispersão. Os gráficos de dispersão possíveis de serem discutidos são listados a seguir e discutidos seus pontos mais importantes:

[4]http://www.gnuplot.info

1. A Figura 5.3 mostra a distribuição dos dados em relação à unidade federativa. Alguns estados parecem concentrar um número maior de *chargebacks* como Rio de Janeiro, São Paulo e Santa Catarina mas isso não fica muito claro visualmente. Um ponto a ser observado nesse gráfico é que no topo do eixo Y existem dois valores anômalos para as Unidades Federativas, esses registros foram eliminados.

2. A Figura 5.4 mostra a distribuição em relação ao tempo de registro do usuário, onde é possível perceber que a maior parte dos *chargebacks* ocorrem com usuários que estão registrados há pouco tempo.

3. A Figura 5.5 mostra a distribuição em relação aos dias da semana. Por esse gráfico não é identificado nenhum dia em que a transação de *chargeback* ocorre com maior frequência.

4. A Figura 5.6 mostra a distribuição dos dados em relação às horas do dia e pode-se perceber uma distribuição menor das transações no período da madrugada.

5. A Figura 5.7 mostra a distribuição dos dados em relação ao tempo de registro do vendedor. Assim como no caso dos compradores, vendedores com tempo de registro mais recentes são mais propensos a *chargeback*.

6. A Figura 5.8 mostra a distribuição dos dados em relação às categorias de produtos vendidos. Importante notar nesse gráfico que existem categorias onde a distribuição das fraudes é maior do que as outras.

7. A Figura 5.9 mostra a distribuição dos dados em relação ao número de parcelas realizadas na compra. Pode-se perceber que as fraudes se concentram em vendas feitas com um número pequeno de parcelas, geralmente uma ou duas.

Os gráficos analisados mostram alguns padrões nos registros de fraude considerando os atributos selecionados. Conforme informado no Capítulo 4, isso pode ser um bom indicativo que grande parte dos atributos escolhidos podem ser significativos para detecção de fraude.

Continuando a análise da qualidade dos dados, foram feitos histogramas para verificar a distribuição dos dados no *dataset*. Novamente, devido à confidencialidade das informações, apenas alguns gráficos serão exibidos. Todos aqueles apresentados a seguir mostram a distribuição relativa dos registros conforme Equação 4.1 do Capítulo 4 e não são exibidos valores absolutos. Três categorias de informações são observadas: Total - que considera o número total de registros, *Chargeback* - que considera apenas o número total de *chargeback* e *Não-Chargeback* que considera os registros efetivados

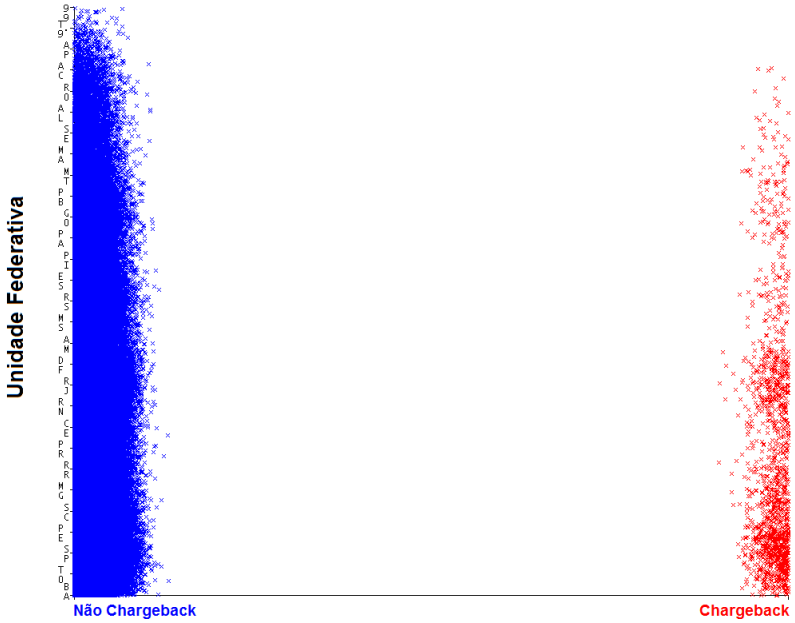

Figura 5.3. Dispersão dos dados em relação à unidade federativa. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa a unidade federativa.

com sucesso. O somatório de cada uma das categorias irá fechar em 100%. Os gráficos gerados são listados e discutidos abaixo:

1. A Figura 5.10 mostra a distribuição relativa dos registros em relação aos dias da semana. No eixo X o código 1 indica domingo e mantendo a sequência até o código 7 que indica sábado. Note que a distribuição relativa indica uma ocorrência maior dos *chargebacks* nos finais de semana.

2. A Figura 5.11 mostra a distribuição relativa dos registros em relação aos horários do dia. Pode-se perceber que as fraudes possuem uma maior ocorrência no período da madrugada e da noite.

3. A Figura 5.12 mostra a distribuição relativa dos registros em relação à idade do comprador. Nota-se que as transações se concentram nas faixas entre 20 e 40 anos. Percebe-se que a distribuição relativa das fraudes se sobressai a partir de idades maiores que 40 anos.

4. A Figura 5.13 mostra a distribuição relativa dos registros em relação às bandeiras de cartão de crédito. Conforme observado no gráfico de dispersão, existem bandeiras em que não há *chargeback*. Esses registros foram retirados da base de dados.

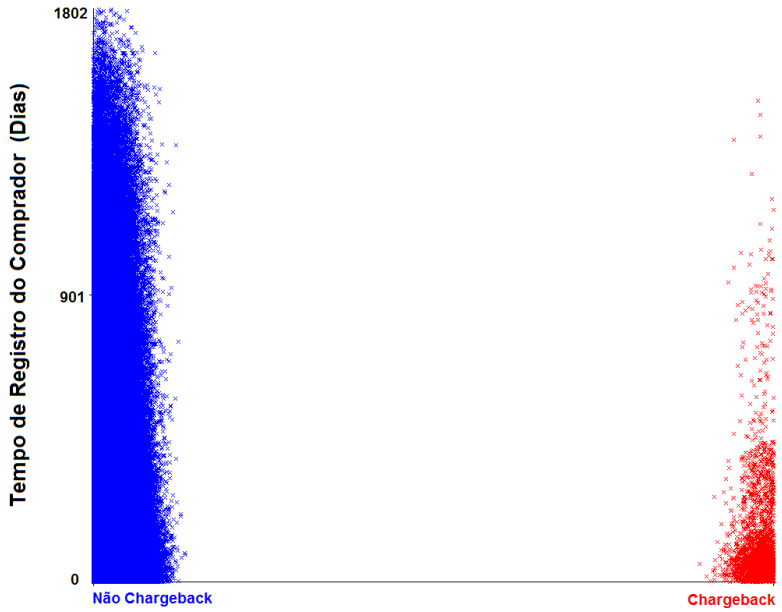

Figura 5.4. Dispersão dos dados em relação ao tempo de registro do comprador. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o tempo de registro do comprador.

5. A Figura 5.14 mostra a distribuição relativa dos registros em relação aos *chargebacks* ocorridos ao longo dos meses. O que mais se destaca no gráfico é que o mês de agosto de 2010 apresenta uma alta concentração de *chargebacks*. Foi verificado com o *UOL PagSeguro* o motivo dessa concentração maior de fraudes em agosto. Identificado o motivo, que nesse caso também é confidencial, os registros do mês de agosto foram desconsiderados e por segurança os de setembro também. No Capítulo 2 foi informado que as notificações de ocorrências de *chargeback* podem ser superiores a 72 dias. Existem casos na base de *UOL PagSeguro* que o registro de *chargeback* ocorreu após 6 meses. Sendo assim, considerando o período de recebimento da base, foram retirados os meses de janeiro de 2011 a julho de 2011 devido ao risco de existirem registro de fraude que ainda não foram notificados. Ao final, a base utilizada será composta apenas pelos meses de outubro, novembro e dezembro de 2010.

Foi verificado também campos que possuíam muitos valores nulos, acima de 90%. Aqueles que possuíam essa característica também foram retirados da base de dados. Também foram retirados campos de IDs, como, por exemplo, ID da transação, ID do comprador, ID do Vendedor, entre outros. Sendo assim, ao final desta etapa espera-se ter retirado a maior quantidade de ruídos possível deixando apenas informações que

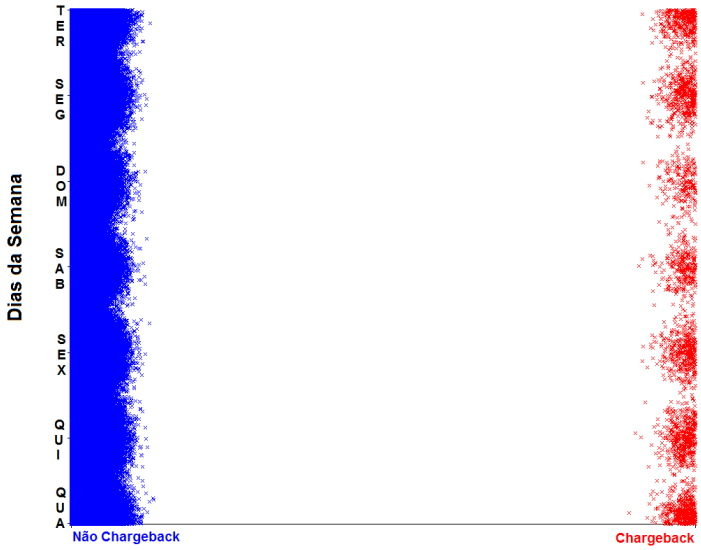

Figura 5.5. Dispersão dos dados em relação aos dias da semana. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa os dias da semana.

irão auxiliar na descoberta de padrões de fraudes. Um ponto relevante nesta etapa é que foi possível verificar como os dados se distribuem no *dataset*. Além do mais, os gráficos mostraram alguns padrões para os *chargebacks*, o que fornece bons indícios de resultados relevantes na etapa de mineração de dados. Uma das características do *dataset* é apresentar uma probabilidade baixíssima de fraude, muito inferior a 1%. Entretanto, mesmo com uma probabilidade baixa de fraude, os prejuízos financeiros são consideráveis e caso não sejam monitorados, podem causar o colapso da empresa.

5.4 Transformação

Nesta etapa focou-se principalmente na criação de novos atributos, derivados daqueles existentes no *dataset*. Foi realizada uma redução de dimensionalidade com o objetivo de viabilizar a etapa de mineração de dados. Optou-se por não fazer uma amostragem do *dataset* gerado para que não ocorra perda de informação, apenas foi feita uma separação por meses.

Alguns atributos passaram por transformações com o objetivo de torná-los viáveis para as técnicas de mineração de dados, uma vez que alguns formatos não permitiriam ganho de informação mesmo sendo relevantes. Sendo assim, os atributos de data que estavam no formato *DD/MM/AAAA HH:MM* foram separados gerando 4 atributos: dia da semana, semana, mês e hora. Acredita-se que esse formato seja mais inteligível para as técnicas de mineração. O campo *data de nascimento do titular do cartão* foi

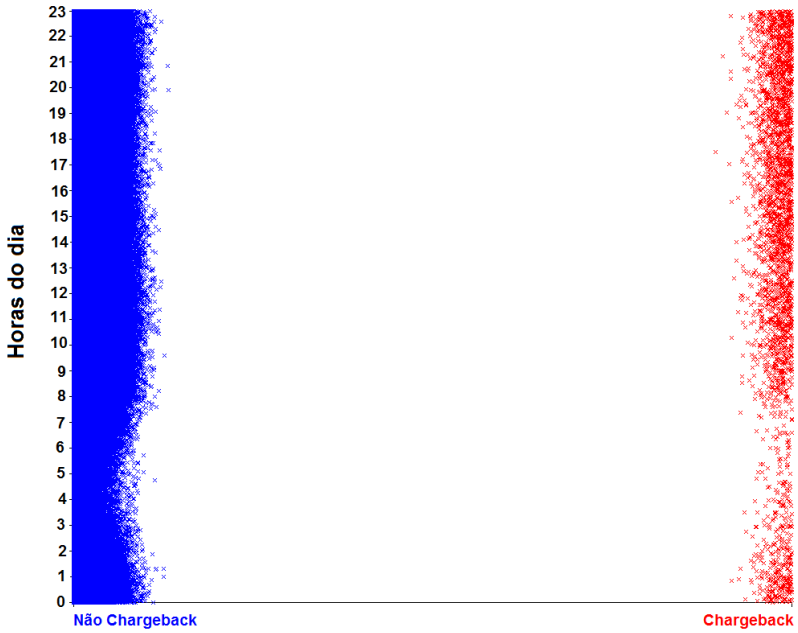

Figura 5.6. Dispersão dos dados em relação às horas do dia. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa as horas do dia.

transformado em idade, realizando-se o cálculo baseado na *data de criação da transação*. Dessa forma, o campo *idade do proprietário do cartão* irá indicar sua idade no momento em que realizou a transação.

Por meio dos campos originais foram gerados campos derivados com o objetivo de extrair mais informação que possa indicar a existência de fraude. Para criação desses novos campos foi considerado o conhecimento obtido no estudo do Capítulo 2, bem como as orientações apresentadas no Capítulo 4 no passo de transformação. Os novos campos gerados são listados e descritos abaixo:

1. *Período de registro do comprador:* tempo em que o comprador se encontra registrado no *UOL PagSeguro* até o momento da realização da transação.

2. *Período de registro do vendedor:* tempo em que o vendedor se encontra registrado no *UOL PagSeguro* até o momento da realização da transação.

3. *Flag usuário registrado:* indica se o comprador que está realizando a transação está registrado no *UOL PagSeguro*.

4. *Flag CPF:* indica se o número do CPF do titular do cartão é diferente do CPF do usuário cadastrado no *UOL Pagseguro*. Esse flag foi criado uma vez que dessa

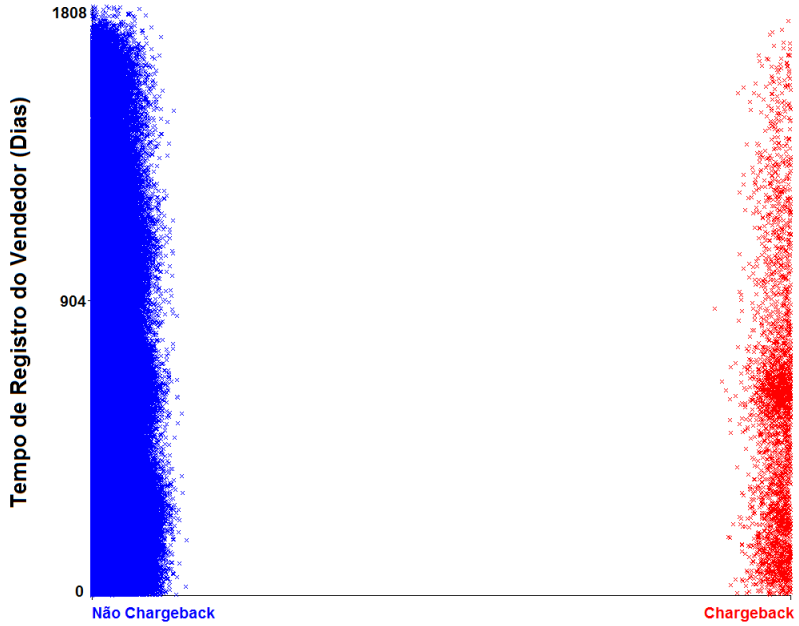

Figura 5.7. Dispersão dos dados em relação ao tempo de registro do vendedor. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o tempo de registro do vendedor.

forma não se revela o número do CPF do usuário. Além do mais, esse era um dos requisitos para se obter os dados diretamente de produção.

5. *Flag DDD:* indica se o número do DDD do titular do cartão é diferente do DDD do usuário cadastrado no *UOL PagSeguro*.

Para se evitar o problema da *maldição da dimensionalidade*, descrito no Capítulo 4, e também, considerando o fato de que muitos algoritmos de mineração de dados funcionam melhor com um número menor de atributos, foram utilizadas técnicas para avaliar a qualidade dos atributos selecionados. Por meio dos valores obtidos com essas técnicas será verificada a possibilidade de exclusão de atributos considerados irrelevantes para classificação dos registros. Dentre as várias técnicas de seleção de atributos (descritas no Capítulo 4), optou-se pela abordagem de filtro, uma vez que ela é independente do algoritmo de classificação e também é computacionalmente menos custosa, já que não é necessária a execução de algum algoritmo de mineração de dados. Além do mais, como o objetivo é excluir atributos irrelevantes, optou-se por utilizar técnicas de *ranking* dos atributos. Estas técnicas fazem uso de uma métrica para avaliar a qualidade de cada atributo individualmente. Dessa forma, é possível ordenar o conjunto de atributos formando um *ranking*, que então será utilizado para selecionar aqueles

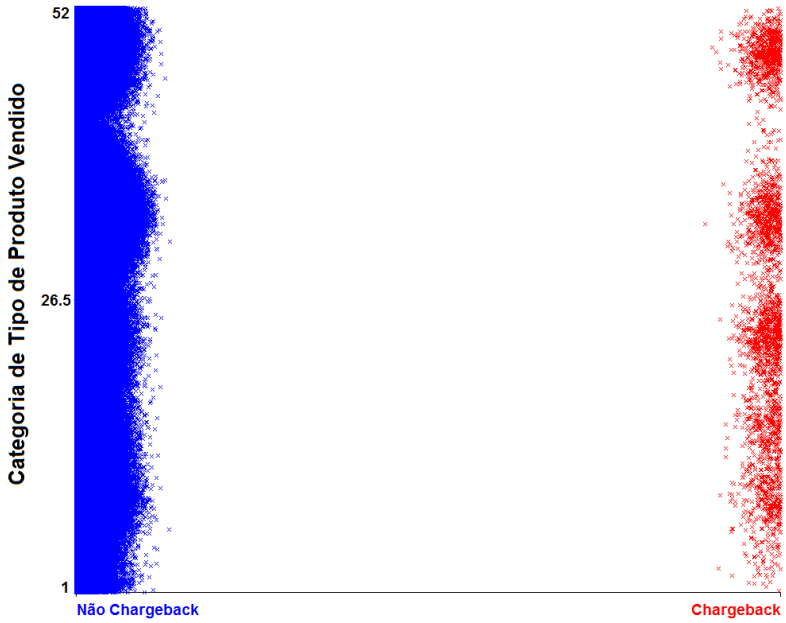

Figura 5.8. Dispersão dos dados em relação a categoria de tipo de produto vendido pelo vendedor. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa a categoria de tipo de produto vendido pelo vendedor.

em que o valor da métrica esteja acima de um limiar. As métricas mais comumente utilizadas são *entropia* e estatística *Chi-Quadrado*.

Para realização dessa tarefa utilizou-se a ferramenta Weka e os algoritmos selecionados, considerando as escolhas definidas acima, foram *InfoGainAttributeEval* que avalia os atributos medindo ganho de informação(entropia) com respeito à classe e *ChiSquaredAttributeEval* que avalia os atributos através do cálculo da estatística *Chi-Quadrado* com respeito à classe. Maiores detalhes sobre esses algoritmos podem ser encontrados em Written & Frank [2005]. Os resultados obtidos com os algoritmos foram comparados e aqueles atributos em que os valores da métrica eram muito baixos (abaixo de 0) foram analisados quanto à relevância semântica e excluídos. Para evitar-se uma exposição desnecessária dos atributos existentes na base de dados do *UOL PagSeguro*, as saídas das técnicas não serão exibidas.

A ferramenta Weka também fornece uma visualização gráfica da correlação entre os atributos, sendo assim, uma análise visual da correlação entre eles foi feita no sentido de verificar a existência de atributos altamente correlacionados. Não foi encontrado nenhum caso desse tipo. Ao final de toda essa análise, foram selecionados 21 atributos candidatos a serem utilizados pelas técnicas. Estes atributos também passaram por uma avaliação dos especialistas do *UOL PagSeguro* com relação à sua relevância. A Tabela 5.1 lista e descreve cada um deles.

Tabela 5.1. Lista de atributos selecionados

Atributo	**Descrição**
Valor	Atributo numérico que representa o valor da transação.
Indicador de Pontuação	Indica se a transação gerou pontuação.
Dia da Transação	Dia da semana em que a transação foi criada. Exemplo: Segunda, Terça, Quarta, etc.
Hora da Transação	Horário em que a transação foi iniciada.
Tipo Usuário Comprador	Define o tipo de conta do usuário.
Período de Registro do Comprador	Número de dias em que o comprador foi registrado até o momento da transação.
Número de Parcelas	Número de parcelas selecionadas para o pagamento.
Pontos do Comprador	Número de pontos absolutos, acumulados pelo usuário, baseado nas transações bem sucedidas às quais ele realizou.
Período de Registro do Vendedor	Número de dias em que o vendedor foi registrado até o momento da transação.
Categoria da Loja	Categoria principal de produto ou serviço oferecido pelo vendedor.
Operadora de Cartão de Crédito	Identificador da operadora de cartão de crédito.
Idade Proprietário Cartão	Idade do proprietário do cartão na data de criação da transação.
Flg DDD	Indica se o número do DDD do titular do cartão no serasa é diferente do DDD do usuário cadastrado no *UOL PagSeguro*.
Estado	Sigla do estado brasileiro do usuário.
Flg Leitura	Indica se o comprador visualizou os detalhes da transação.
Classe	Indica a classe ao qual a transação pertence. Ou seja, *chargeback* e *não chargeback*.
Flg Usuário Registrado	Indica se o comprador que está realizando a transação está registrado no *UOL PagSeguro*.
Flg CPF	Indica se o número do CPF do titular do cartão é diferente do CPF do usuário cadastrado no *UOL Pagseguro*.
Informação Válida	Indica se as informações fornecidas pelo Serasa estão compatíveis com as informações fornecidas pelo comprador.
Status Serasa	Identificador do status no Serasa.
CPF Válido	Indicador do Serasa se o CPF é válido.

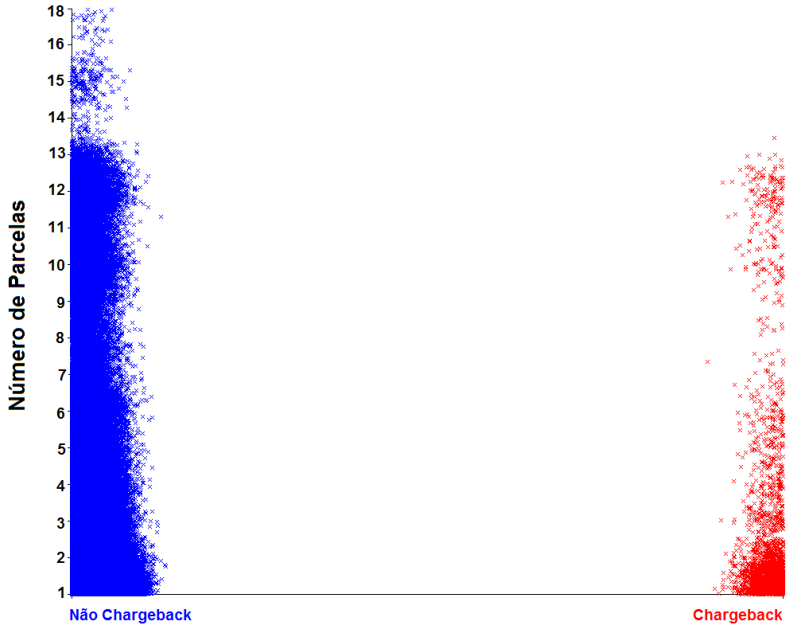

Figura 5.9. Dispersão dos dados em relação ao número de parcelas. O eixo X a esquerda em azul representa os registros de não *chargeback* e a direita em vermelho os registros de *chargeback*. O eixo Y representa o número de parcelas na compra do produto.

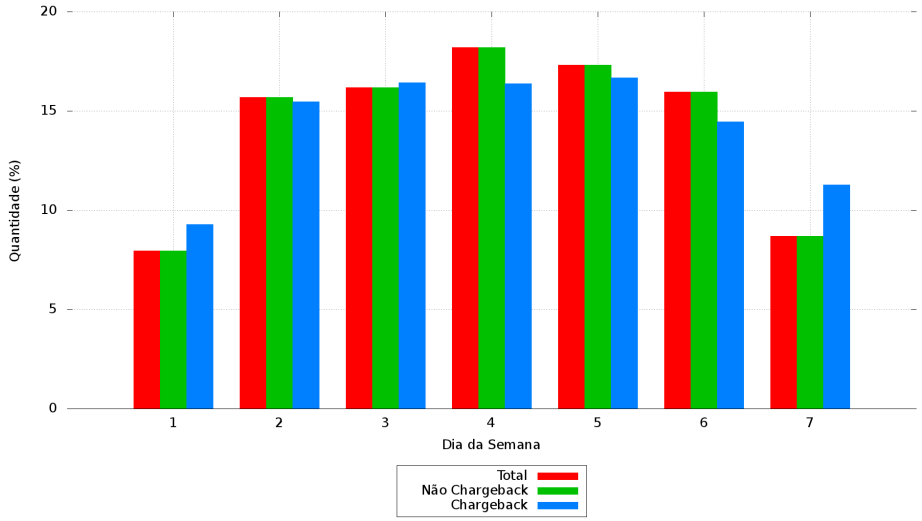

Figura 5.10. Distribuição dos registros em relação ao dia da semana.

5.5 Mineração de Dados

Nesta etapa deve ser definida a tarefa de mineração de dados a ser realizada, bem como as técnicas utilizadas nessa tarefa. Como o objetivo do processo de detecção de fraude é, dado um conjunto de atributos, definir se uma transação é fraudulenta ou não,

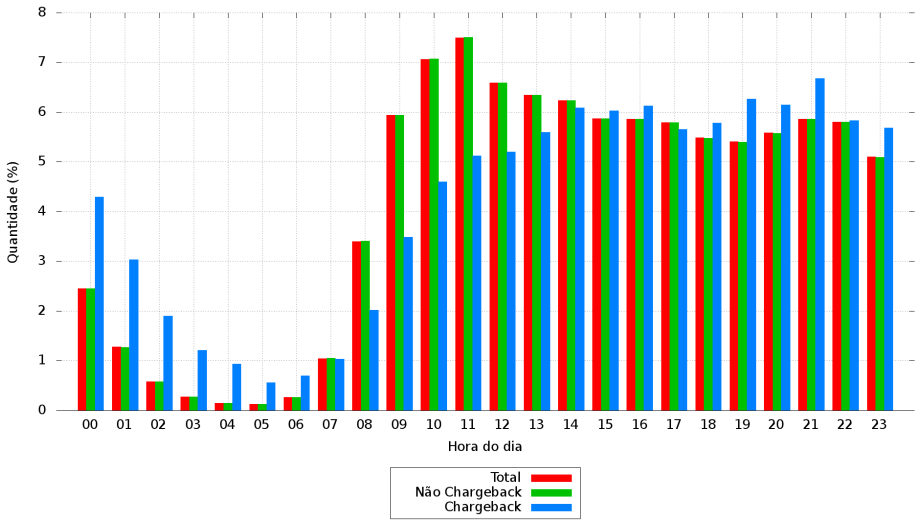

Figura 5.11. Distribuição dos registros em relação às horas do dia.

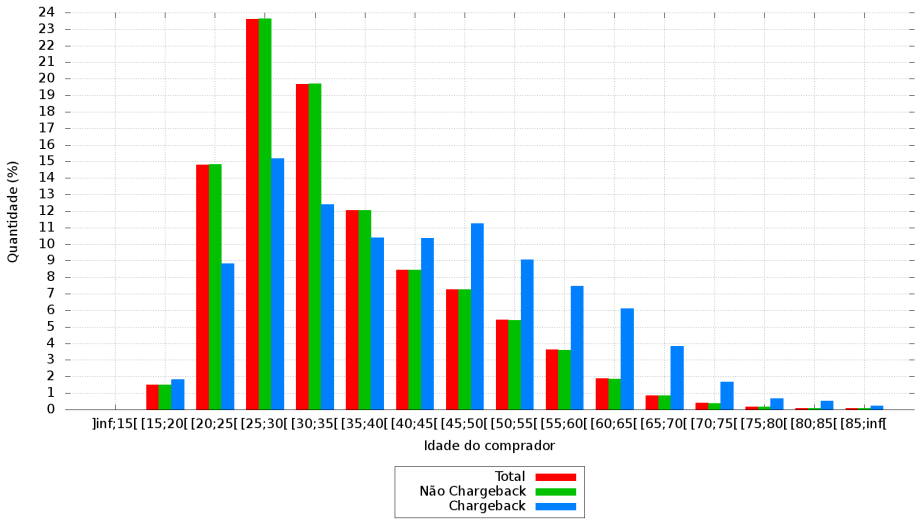

Figura 5.12. Distribuição dos registros em relação a idade do comprador.

a tarefa de mineração de dados mais apropriada para esse objetivo é a classificação. Os algoritmos selecionados são aqueles definidos e descritos no Capítulo 4. A seguir são feitas algumas considerações.

1. *Árvore de Decisão:* O Algoritmo utilizado foi o *C4.5* que consegue lidar também com atributos contínuos. Segundo Phua et al., em termos de precisão da previsão, *C4.5* é um pouco melhor do que *CART* e *ID3*. Uma discussão mais profunda do algoritmo é apresentada por Quinlan [1993].

2. *Classificador Baseado em Regra:* Conforme apresentado no Capítulo 4, foi es-

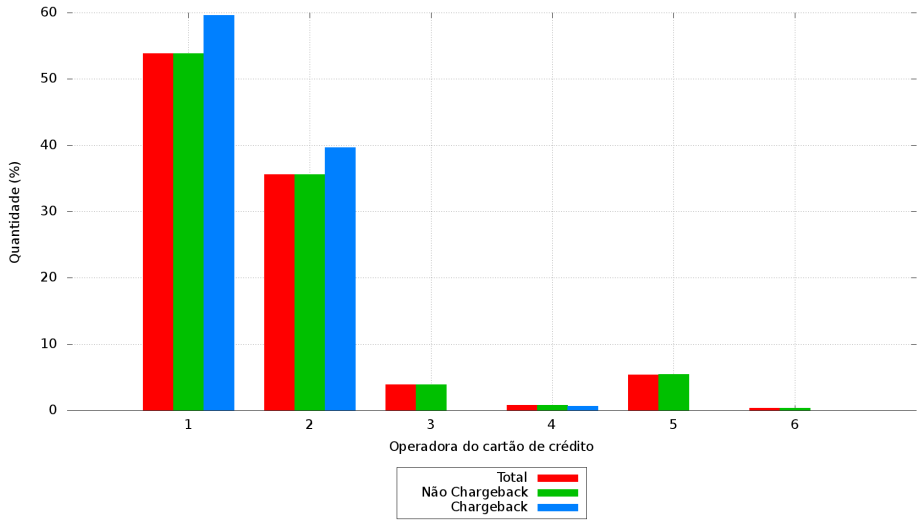

Figura 5.13. Distribuição dos registros em relação a bandeira de cartão de crédito.

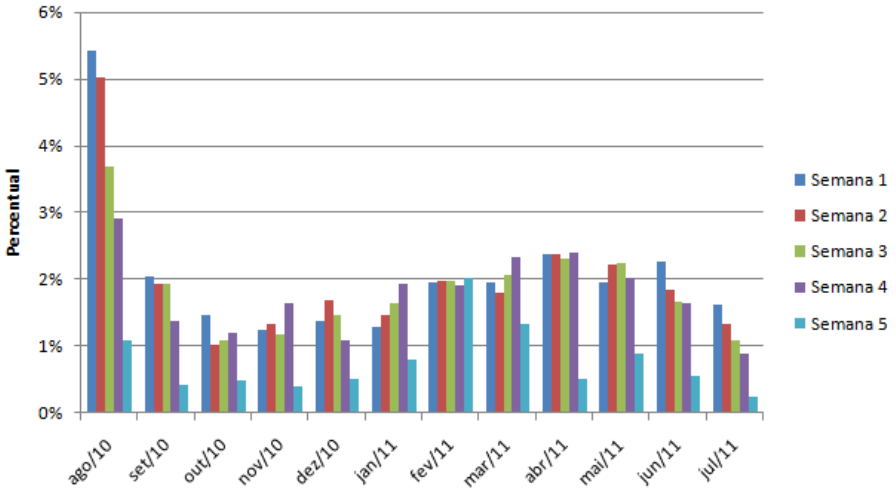

Figura 5.14. Distribuição dos registros em relação aos *chargebacks*.

colhido um algoritmo para cada uma das abordagens existentes: abordagem de ordenação das regras representada pelo *RIPPER* e a abordagem de regras não ordenadas representado pelo *LAC*. Maiores detalhes sobre os algoritmos podem ser encontrados respectivamente, em Cohen [1995] e Veloso et al. [2006].

3. *Naive Bayes:* diferente das abordagens de árvore de decisão e baseado em regras, considera que o relacionamento entre a classe e o atributo é não determinístico. Por considerar os atributos independentes, é um algoritmo que possui um baixo custo de processamento. Detalhes sobre o algoritmo pode ser encontrado em

Langley et al. [1992].

4. *Boosting, Oversampling e Stacking:* O algoritmo utilizado para *boosting* foi o *AdaBoost* desenvolvido por Freund & Schapire [1995], este algoritmo também é utilizado na pesquisa de detecção de fraudes em cartão de crédito realizada por Philip K. Chan & Stolfo [1999]. A técnica de *oversampling* consistiu em realizar sobreamostragens da classe minoritária (replicação das classes minoritárias) criando *datasets* com distribuição de classe de 10% até 50% variando de 10 e 10, conforme realizado no artigo Phua et al. [2004]. A combinação de algoritmos para a técnica *Stacking* foi *C4.5*, *RIPPER* e *Naive Bayes* tendo como *meta learner* o classificador *Naive Bayes*, essa combinação foi uma adaptação da combinação definida no artigo Phua et al. [2004]. Detalhes sobre a técnica de *stacking* podem ser encontrados em Written & Frank [2005].

A ferramenta Weka foi utilizada para a execução de todos os algoritmos selecionados, com exceção do algoritmo *LAC*, que neste caso, foi utilizada a implementação disponibilizada pelo autor[5]. Para a execução no ambiente do Weka o arquivo no formato *CSV* deve ser transformado para o formato do Weka cuja a extensão é *ARFF*. Essa conversão é necessária pois garante um melhor desempenho do algoritmo quando se usa essa ferramenta. No caso da implementação da técnica de *boosting* foi utilizado o ambiente Weka, uma vez que a mesma já se encontra implementada nessa ferramenta. Sendo assim, os testes com o *boosting* serão feitos apenas com os algoritmos disponíveis no Weka que são *J48* (implementação do algoritmo *C4.5*), *RIPPER* e *Naive Bayes*.

O algoritmo *LAC* trabalha apenas com atributos discretos, sendo assim, para sua utilização alguns atributos tiveram que ser discretizados. Para isso, foi utilizada a ferramenta Weka onde os atributos numéricos foram categorizados em 10 bins. Também foi desenvolvido um *script* para transformação do *dataset* do formato *CSV* para o formato reconhecido pelo *LAC*.

Nesta etapa cada um dos algoritmos selecionados serão executados para que seus resultados sejam avaliados na próxima etapa. Como mencionado, os meses utilizados foram outubro, novembro e dezembro de 2010, sendo que a execução desses algoritmos foi feita em cada um dos meses separadamente. A seguir serão discutidos os resultados alcançados.

[5]http://homepages.dcc.ufmg.br/∼adrianov/software/ddac.htm

5.6 Interpretação e Avaliação dos Resultados

Esta etapa consiste na avaliação dos resultados obtidos na etapa de mineração de dados. Para isso, conforme mostrado no Capítulo 4, são utilizados um conjunto de métricas para indicar a qualidade dos resultados obtidos nas diversas técnicas. A execução dos algoritmos foi feita considerando a metodologia de avaliação proposta no Capítulo 4 e ilustrada pela Figura 4.8. Sendo assim, para a validação cruzada, cada um dos meses foram divididos em 5 partições, sendo que, 4 são utilizadas para treino e 1 para teste. A execução é repetida até que todas as partições tenham sido testadas. As métricas são calculadas considerando uma média dos resultados obtidos em cada partição de teste. Dessa forma, para cada mês são realizadas 5 execuções de cada algoritmo.

Os resultados obtidos nesta etapa são mostrados a seguir e em todos os casos as tabelas apresentadas representam o ponto no *ranking* de probabilidade de fraude onde os resultados são maximizados. Nessas tabelas são apresentadas as seguintes medidas: *Precisão*, *Revocação*, Eficiência Econômica Sem Penalidade *(EE_ SP)*, Eficiência Econômica Com Penalidade *(EE_ CP)*, Ganho Sem Penalidade *(Ganho_ SP)*, Ganho Com Penalidade *(Ganho_ CP)*, Percentual do *Ranking* onde o resultado é maximizado *(Perc. Rank.)* e *Medida F1*. Para facilitar a identificação, o melhor resultado de cada medida é destacado em **negrito** e assinalado com um (†). Após a apresentação das tabelas, com o objetivo apenas de ilustrar os resultados dos classificadores ao longo do *ranking*, serão exibidos três gráficos para a técnica que apresentou o melhor resultado de eficiência econômica no geral. O mês a ser exibido será aquele que possuir a melhor eficiência econômica considerando a técnica vencedora. Esse critério foi adotado para se evitar uma exposição desnecessária de uma quantidade muito grande de gráficos, uma vez que as tabelas a serem apresentadas já filtram as informações mais importantes para exibição. Os gráficos a serem exibidos são: *Precisão X Revocação*, *Eficiência Econômica* e *Percentual de ganho em relação ao ganho máximo*. Com relação ao percentual do *ranking* onde os melhores resultados são obtidos, espera-se que o máximo seja alcançado com a menor porcentagem do *ranking*. Caso os registros considerados como fraude sejam encaminhados para uma revisão manual, aqueles classificadores que obtiverem os melhores resultados com um menor percentual do *ranking* conterão um número menor de registros para análise manual, o que reduzirá os custos dessa análise.

A Tabela 5.2 mostra o resultado em percentual das técnicas *J48*, *RIPPER (RIP.)*, *LAC*, *Naive Bayes (NB)* sem alteração na distribuição e sem combinação de classificadores. Na maioria dos casos, o *RIPPER* apresentou um melhor modelo considerando a Medida F1, onde obteve uma alta precisão, em média 69%, e uma boa cobertura

Tabela 5.2. Resultado em percentual das técnicas selecionadas sem alteração na distribuição e sem combinação de classificadores.

Mês	Medida	J48	RIP.	LAC	NB
Out.	**Precisão**	5,19	**62,25**†	6,68	53,00
	Revocação	**18,14**†	10,75	14,43	2,30
	EE_SP	**3,00**†	2,00	2,00	**3,00**†
	EE_CP	2,00	2,00	1,00	**3,00**†
	Ganho_SP	7,73	5,38	6,06	**8,54**†
	Ganho_CP	5,16	5,34	4,46	**8,41**†
	Perc. Rank.	1,14	0,06	0,70	**0,05**†
	Medida F1	8,07	**18,33**†	9,13	4,41
Nov.	**Precisão**	15,93	**77,00**†	23,00	57,50
	Revocação	**19,88**†	15,00	6,27	4,45
	EE_SP	5,00	5,00	2,00	**6,00**†
	EE_CP	5,00	5,00	2,00	**6,00**†
	Ganho_SP	15,83	14,85	7,26	**18,06**†
	Ganho_CP	14,98	14,78	6,65	**17,90**†
	Perc. Rank.	0,40	0,06	0,09	**0,03**†
	Medida F1	17,69	**25,11**†	9,85	8,26
Dez.	**Precisão**	18,52	**67,33**†	25,87	54,00
	Revocação	**8,45**†	7,49	5,20	4,00
	EE_SP	**8,00**†	7,00	**8,00**†	7,00
	EE_CP	**8,00**†	7,00	5,00	7,00
	Ganho_SP	**18,32**†	15,90	17,37	15,22
	Ganho_CP	**17,21**†	15,82	12,39	15,04
	Perc. Rank.	0,79	0,04	1,79	**0,03**†
	Medida F1	11,61	**13,48**†	8,66	7,45

de fraudes, uma média de 11%. No entanto, o *Naive Bayes*, no geral apresentou uma melhor eficiência econômica, isso porque as fraudes detectadas por esse classificador apresentavam valores muito maiores do que as fraudes obtidas pelo *RIPPER*. Isso mostra que nem sempre os melhores modelos irão obter a melhor eficiência econômica, sendo necessária a avaliação em conjunto com outras métricas. O mês de dezembro representou uma melhora para o classificador *J48*, que obteve a melhor eficiência econômica e uma Medida F1 comparável à do *RIPPER*. Isso ocorreu porque esse mês possui uma ligeira melhora na distribuição dos registros de fraude, o que pode ter favorecido esse classificador. O resultado não tão bom com o *LAC* provavelmente está associado à necessidade de discretização de algumas variáveis para utilização desse algoritmo, o que representa uma perda de informação. Além do mais, como foi feita uma discretização com intervalos fixos de valores isso pode ter contribuído, ainda mais, para a piora do desempenho. Com relação ao percentual do *ranking* necessário para se maximizar os resultados, o *Naive Bayes* apresentou os menores percentuais para se atingir esse objetivo. Nesta avaliação, conforme já mencionado, a melhor eficiência econômica foi obtida com o algoritmo *Naive Bayes*, sendo o melhor valor obtido no mês de novem-

bro. As Figuras 5.21, 5.22 e 5.22 mostram os gráficos que ilustram o comportamento dos resultados desse classificador ao longo do *ranking*. Cabe ressaltar que os melhores resultados são obtidos na porção inicial dos gráficos (topo do *ranking*) e ocorre uma queda significativa no decorrer do *ranking*. Isso pode ser explicado pela diminuição rápida da precisão à medida que se "caminha" no *ranking*.

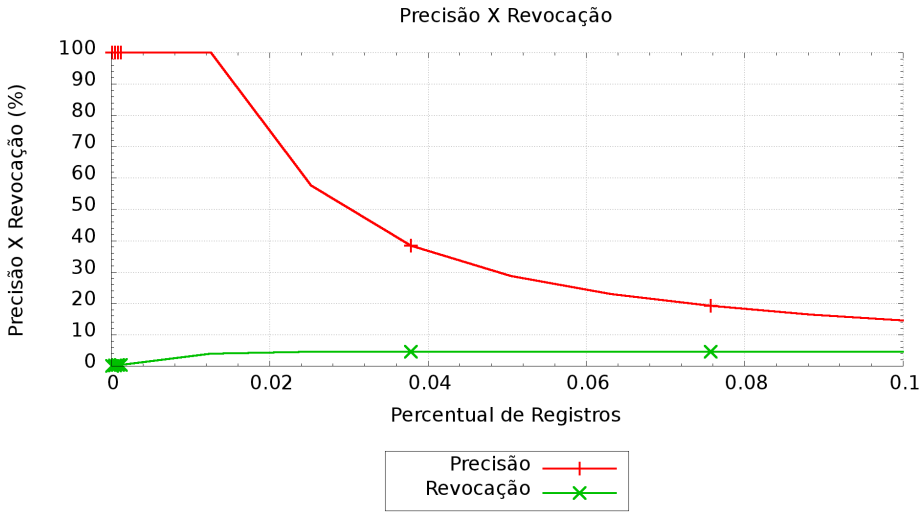

Figura 5.15. *Precisão X Revocação* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores.

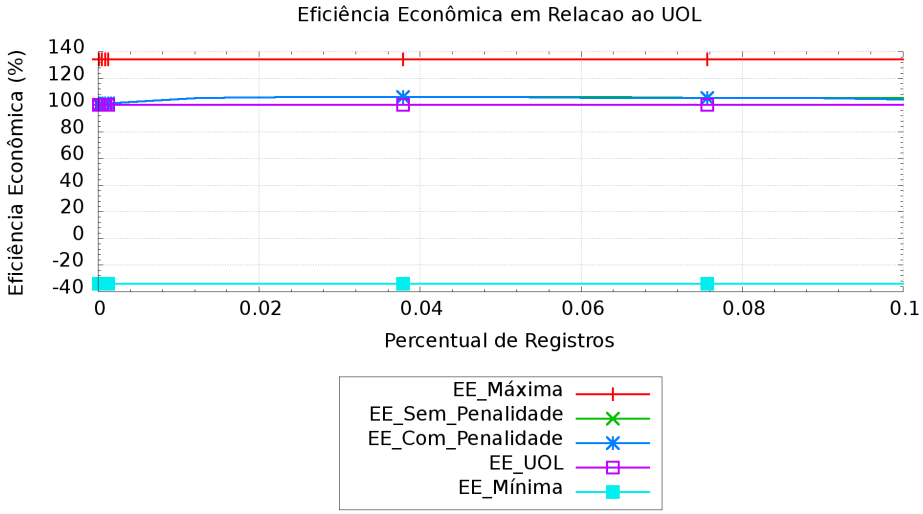

Figura 5.16. *Eficiência Econômica* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores.

A Tabela 5.3 mostra o resultado das técnicas de meta aprendizagem, o objetivo com esse experimento é verificar se as abordagens definidas por essas técnicas podem

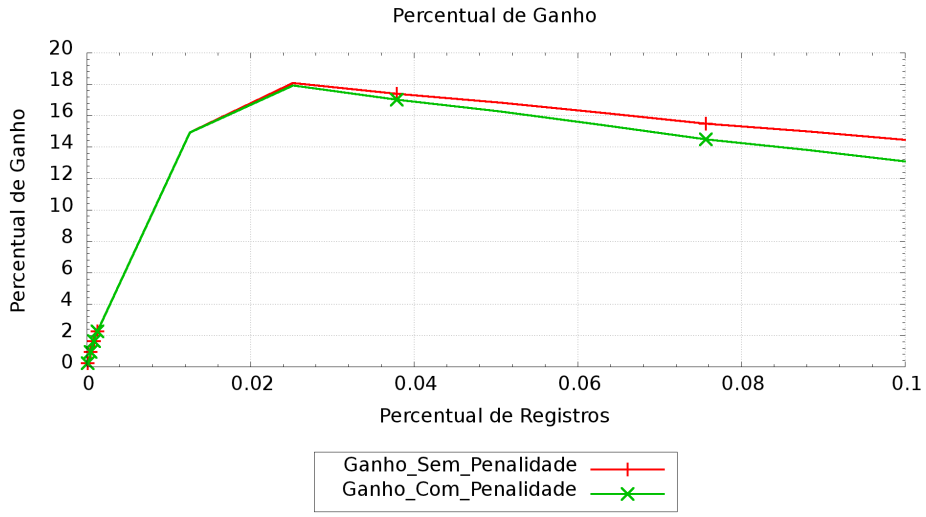

Figura 5.17. *Ganho em relação ao máximo* dos resultados obtidos em novembro com o *Naive Bayes* sem alteração na distribuição e sem combinação de classificadores.

melhorar a qualidade dos resultados dos classificadores definidos no primeiro experimento. A técnica de *boosting*, que reforça os exemplos classificados incorretamente, permitiu uma melhora significativa no algoritmo *J48*, apresentando uma excelente qualidade do modelo em todos os meses, com uma boa eficiência econômica. Para o *RIPPER* isso representou uma piora na precisão, no entanto foi obtida uma melhora considerável na revocação e na eficiência econômica. Isso pode ter ocorrido devido ao reforço nos registros das classes de fraude promovido pelo *boosting*. No entanto, essa técnica piorou consideravelmente o resultado do algoritmo *Naive Bayes*, uma vez que a piora na precisão não refletiu em uma melhora considerável da revocação, porque as amostras criadas por essa técnica não apresentaram um reforço significativo nas classes de fraude. O *Stacking* pode ser considerado o melhor resultado nesse experimento, apresentando um modelo com uma qualidade comparável e, no caso de dezembro, superior ao do *J48*. O mais importante é a obtenção de um ganho em eficiência de 46,46% sem considerar a penalidade e 46,44% considerando a penalidade no erro, ou seja, resultado praticamente igual devido à excelente precisão de 97,71% em um pequeno número de ocorrências próximas ao topo do *ranking*. Isso representa uma enorme economia no caso dos registros assinalados como fraude serem direcionados para a análise manual, uma vez que um baixo número de transações serão enviadas para essa etapa. Quanto menor esse número, maior a economia nessa etapa e, além do mais, com a análise manual pode-se maximizar, ainda mais, o resultado obtido com a técnica. Os melhores resultados obtidos com o *Stacking* foram no mês de dezembro, sendo assim, as Figuras 5.18, 5.19 e 5.20 mostram o comportamento dos resultados desse modelo na parte

Tabela 5.3. Resultado em percentual das técnicas de meta aprendizagem.

Mês	Medida	Boost J48	Boost RIP.	Boost NB	Stacking
Out.	**Precisão**	56,54	10,91	8,50	**81,00**†
	Revocação	31,74	**33,48**†	4,40	24,49
	EE_SP	7,00	7,00	1,00	**10,00**†
	EE_CP	7,00	6,00	1,00	**10,00**†
	Ganho_SP	21,08	22,44	2,48	**31,57**†
	Ganho_CP	20,81	19,09	2,10	**31,33**†
	Perc. Rank.	0,18	1,00	0,17	**0,10**†
	Medida F1	**40,66**†	16,46	5,80	37,61
Nov.	**Precisão**	64,64	11,24	5,60	**93,75**†
	Revocação	35,28	**38,56**†	6,55	29,23
	EE_SP	10,00	8,00	1,00	**15,00**†
	EE_CP	10,00	7,00	1,00	**15,00**†
	Ganho_SP	29,76	24,61	5,02	**43,87**†
	Ganho_CP	29,53	20,68	3,97	**43,83**†
	Perc. Rank.	0,18	1,11	0,38	**0,10**†
	Medida F1	**45,65**†	17,41	6,04	44,57
Dez.	**Precisão**	58,64	11,71	7,60	**97,71**†
	Revocação	**30,48**†	30,44	11,25	25,39
	EE_SP	13,00	12,00	6,00	**20,00**†
	EE_CP	13,00	11,00	5,00	**20,00**†
	Ganho_SP	30,78	28,48	12,84	**46,46**†
	Ganho_CP	30,56	26,08	12,14	**46,44**†
	Perc. Rank.	0,19	0,94	0,54	**0,09**†
	Medida F1	40,11	16,91	9,07	**40,31**†

inicial do *ranking*.

As Tabelas 5.4, 5.5 e 5.6 mostram os resultados das técnicas selecionadas, considerando alterações na distribuição das classes de fraude. O objetivo com esse experimento é avaliar se uma melhora na distribuição das classes de fraude permitirá uma melhora nos resultados dos classificadores. Durante a realização dos testes, ocorreu um problema com o *RIPPER*, já que o aumento na distribuição e no número de registros fez com que o consumo de memória dessa técnica aumentasse consideravelmente, impedindo que a mesma fosse executada com uma distribuição de 10% das classes de fraude. Sendo assim, não foi possível obter os resultados para esse algoritmo para a técnica de *Oversampling*. No geral, tanto para o algoritmo *J48* quanto para o *Naive Bayes* ocorreu uma melhora da cobertura de fraude, mostrando que essa alteração na distribuição contribui para uma melhora na identificação da classe minoritária. No entanto, tanto o *J48* quanto o *LAC* apresentam uma piora na precisão e isso se deve ao aumento na quantidade de registros legítimos sendo classificados como fraude. No caso do *J48* essa piora na precisão não afetou negativamente a eficiência econômica, pois na maioria dos casos ela foi superior ao resultado da técnica onde não houve mudança da distribuição. Para o *LAC* a alteração da distribuição representou uma piora na eficiência econômica

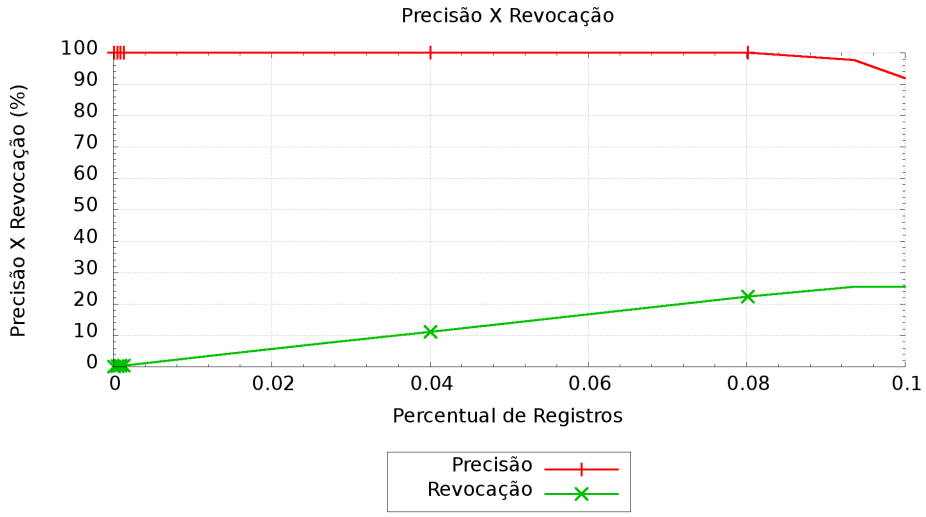

Figura 5.18. *Precisão X Revocação* dos resultados obtidos em dezembro com o *Stacking*.

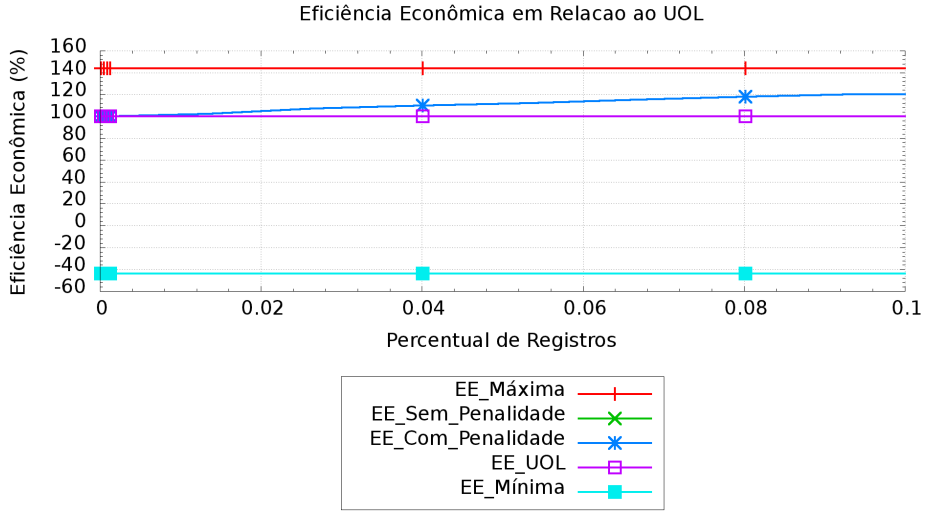

Figura 5.19. *Eficiência Econômica* dos resultados obtidos em dezembro com o *Stacking*.

uma vez que houve uma queda significativa na precisão. Nesse algoritmo, essa nova distribuição apenas aumentou as probabilidades dos registros que já são fraude, no entanto muitos registro legítimos que antes não eram classificados como fraude passaram a ser considerados como tal, o que acarretou piora na precisão e pouca ou nenhuma melhora na revocação. O *Naive Bayes* mostrou os melhores resultados apresentando melhoras significativas em todas as métricas. O mês de dezembro apresentou o melhor resultado em relação à eficiência econômica, atingindo um valor de 36,42% sem considerar a penalidade e 36,41% considerando a penalidade, mas não superior à técnica

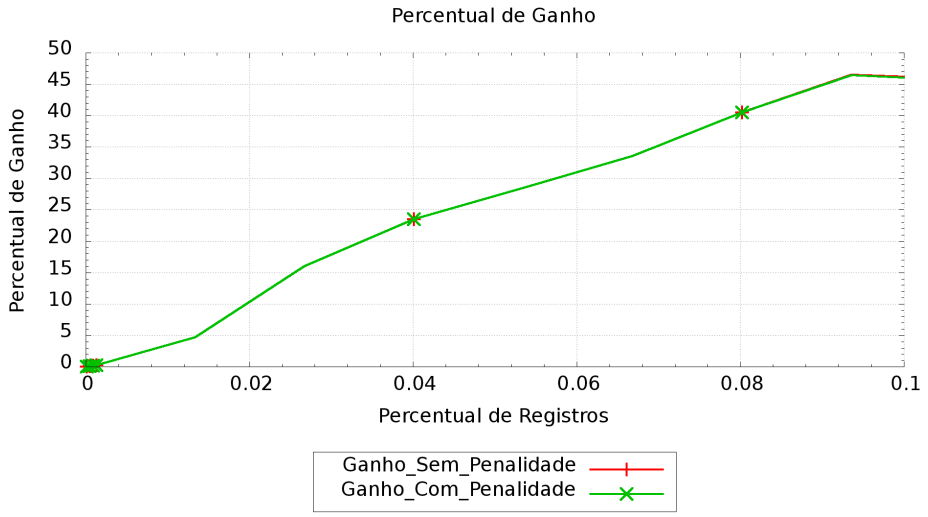

Figura 5.20. *Ganho em relação ao máximo* dos resultados obtidos em dezembro com o *Stacking*.

de *Stacking*. As Figuras 5.21, 5.22 e 5.23 mostram o comportamento dos resultados do modelo *Naive Bayes* na parte inicial do *ranking* para o mês de dezembro.

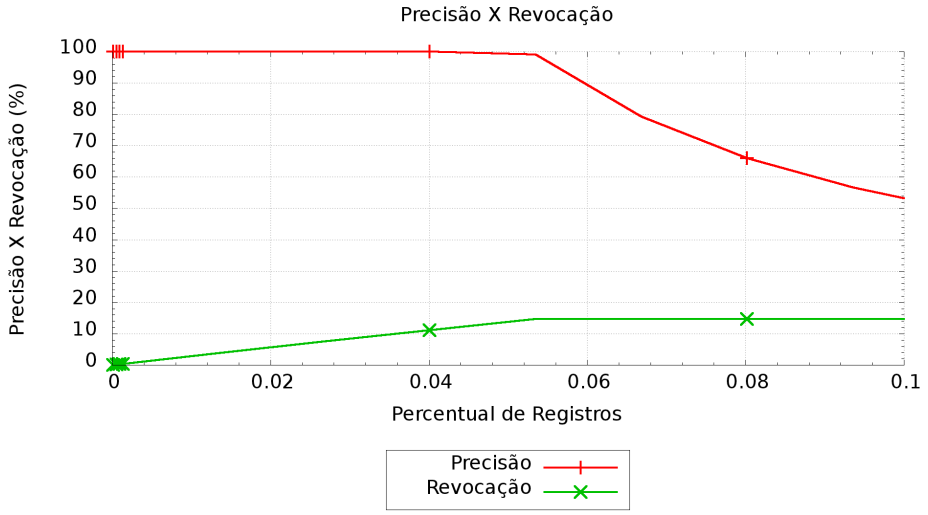

Figura 5.21. *Precisão X Revocação* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*.

Após a realização desses experimentos, pode-se avaliar algumas hipóteses realizadas em relação à metodologia aplicada e a sua eficácia na resolução do problema. Essas hipóteses serão discutidas a seguir:

1. **Os atributos selecionados são relevantes para a detecção de fraude.** Todas as tabelas de resultados apresentadas mostram ganhos em relação à efi-

Tabela 5.4. Resultado em percentual do *oversampling* para a técnica *J48*.

Mês	Medida	10% J48	20% J48	30% J48	40% J48	50% J48
Out.	**Precisão**	8,38	6,72	7,80	**11,53**†	9,30
	Revocação	17,02	**20,60**†	17,02	14,95	18,88
	EE_SP	**3,00**†	**3,00**†	**3,00**†	**3,00**†	**3,00**†
	EE_CP	2,00	2,00	2,00	2,00	**3,00**†
	Ganho_SP	8,11	9,25	9,44	8,85	**9,98**†
	Ganho_CP	6,29	6,39	7,29	7,67	**8,15**†
	Perc. Rank.	0,66	1,00	0,75	**0,42**†	0,66
	Medida F1	11,23	10,13	10,70	**13,02**†	12,46
Nov.	**Precisão**	**10,05**†	9,04	7,50	7,12	8,90
	Revocação	22,34	21,87	23,94	**25,26**†	21,17
	EE_SP	5,00	5,00	5,00	**6,00**†	5,00
	EE_CP	**5,00**†	4,00	4,00	4,00	**5,00**†
	Ganho_SP	15,28	14,44	14,23	**16,14**†	15,82
	Ganho_CP	13,33	12,12	11,05	12,99	**13,70**†
	Perc. Rank.	**0,72**†	0,78	1,03	1,15	0,77
	Medida F1	**13,86**†	12,79	11,42	11,11	12,53
Dez.	**Precisão**	**9,06**†	8,11	8,07	7,80	5,89
	Revocação	21,86	**27,09**†	22,75	20,23	25,13
	EE_SP	9,00	**10,00**†	9,00	9,00	8,00
	EE_CP	8,00	**9,00**†	8,00	8,00	7,00
	Ganho_SP	21,20	**22,68**†	20,58	19,58	18,89
	Ganho_CP	19,21	**20,09**†	18,28	17,45	15,41
	Perc. Rank.	**0,88**†	1,20	1,02	0,94	1,54
	Medida F1	**12,81**†	12,48	11,91	11,26	9,54

ciência econômica do *UOL PagSeguro*. Sendo assim, pode-se considerar que os atributos selecionados são relevantes para resolução do problema. No entanto, existem vários outros atributos que podem representar indícios de fraude, conforme mostrados nos Capítulos 2 e 4, sendo que novos *datasets* podem ser gerados considerando esses atributos e propiciar uma melhora dos resultados obtidos.

2. **As técnicas selecionadas são eficazes na detecção de fraude.** Os bons resultados apresentados com as técnicas não só mostram a relevância dos atributos selecionados como também a eficiência das técnicas na detecção de fraude. Até mesmo o pior resultado obtido mostrou ganhos de 4,46% em relação ao *UOL PagSeguro*. Sendo assim, todas elas apresentaram resultados significativos na detecção de fraude. Além do mais, de acordo com Brause et al. [1999], dado o elevado número de transações realizadas a cada dia, uma redução de 2,5% em atos fraudulentos irá proporcionar uma economia de um milhão de dólares por ano. O que destaca ainda mais a importância dos resultados alcançados.

3. **A utilização do *ranking* permite uma melhora no resultado da classificação.** Na maioria dos casos, o *ranking* potencializou os resultados dos classifi-

Tabela 5.5. Resultado em percentual do *oversampling* para a técnica *Naive Bayes*.

Mês	Medida	10% NB	20% NB	30% NB	40% NB	50% NB
Out.	**Precisão**	70,00	84,00	**91,00**†	100,00	72,67
	Revocação	6,04	7,26	7,86	8,64	**9,42**†
	EE_SP	5,00	7,00	7,00	**8,00**†	**8,00**†
	EE_CP	5,00	7,00	7,00	**8,00**†	**8,00**†
	Ganho_SP	16,39	20,19	21,33	23,31	**24,96**†
	Ganho_CP	16,29	20,10	21,28	23,31	**24,95**†
	Perc. Rank.	**0,03**†	**0,03**†	**0,03**†	**0,03**†	0,04
	Medida F1	11,12	13,36	14,47	15,91	**16,68**†
Nov.	**Precisão**	74,33	84,00	90,00	97,33	**100,00**†
	Revocação	8,69	9,82	10,52	11,38	**11,70**†
	EE_SP	9,00	10,00	10,00	**11,00**†	**11,00**†
	EE_CP	9,00	10,00	10,00	**11,00**†	**11,00**†
	Ganho_SP	26,71	29,23	29,90	31,44	**33,02**†
	Ganho_CP	26,62	29,18	29,86	31,43	**32,97**†
	Perc. Rank.	**0,03**†	**0,03**†	**0,03**†	**0,03**†	0,05
	Medida F1	15,56	17,58	18,84	20,38	**20,95**†
Dez.	**Precisão**	84,33	**100,00**†	55,70	71,40	99,00
	Revocação	9,39	11,14	12,40	13,25	**14,70**†
	EE_SP	12,00	13,00	14,00	15,00	**16,00**†
	EE_CP	12,00	13,00	14,00	15,00	**16,00**†
	Ganho_SP	26,52	30,65	31,90	33,64	**36,42**†
	Ganho_CP	26,43	30,65	31,57	33,40	**36,41**†
	Perc. Rank.	**0,04**†	**0,04**†	0,08	0,07	**0,04**†
	Medida F1	16,90	20,05	20,28	22,35	**25,60**†

cadores. No caso do *LAC* a identificação das fraudes somente foi possível com a utilização do *ranking* das probabilidades. No caso do *J48*, sem que fosse utilizado meta aprendizado ou *oversampling*, foi possível uma melhora da cobertura com o *ranking*, mas uma piora na precisão e na eficiência econômica. No entanto, a eficiência econômica foi pouco afetada. Sendo assim, nem sempre o *ranking* potencializará os resultados obtidos, mas não causará perdas muito significativas.

4. **Técnicas de meta aprendizado podem melhorar a qualidade dos resultados de um classificador.** A Tabela 5.3 mostra que na maioria dos casos as técnicas de meta aprendizagem podem representar uma melhora dos resultados do classificador, já que para o *Naive Bayes* essa técnica representou uma piora nos resultados. Vale ressaltar que o melhor resultado na detecção de fraude foi obtido com o *Stacking* que combina as características de vários classificadores e tenta extrair o melhor resultado deles.

5. **Melhora na distribuição das classes minoritárias melhoram os resultados do classificador.** As Tabelas 5.4, 5.5 e 5.6 mostram que os resultados dos

Tabela 5.6. Resultado em percentual do *oversampling* para a técnica *LAC*.

Mês	Medida	10% LAC	20% LAC	30% LAC	40% LAC	50% LAC
Out.	**Precisão**	6,64	6,47	5,92	6,04	**8,48**†
	Revocação	18,49	17,75	**19,70**†	16,97	11,36
	EE_SP	**2,00**†	**2,00**†	**2,00**†	1,00	1,00
	EE_CP	**1,00**†	**1,00**†	**1,00**†	0,00	**1,00**†
	Ganho_SP	**5,39**†	4,87	4,85	3,94	3,66
	Ganho_CP	**2,35**†	1,86	0,67	0,53	2,12
	Perc. Rank.	0,79	0,77	1,08	0,91	**0,44**†
	Medida F1	**9,77**†	9,48	9,10	8,91	9,71
Nov.	**Precisão**	**7,63**†	3,47	2,34	6,46	5,90
	Revocação	**18,79**†	17,80	15,00	7,00	7,13
	EE_SP	**2,00**†	1,00	1,00	1,00	1,00
	EE_CP	**1,00**†	**1,00**†	**1,00**†	0,00	0,00
	Ganho_SP	**5,41**†	3,92	2,22	3,50	3,22
	Ganho_CP	2,29	**3,28**†	1,69	0,96	0,69
	Perc. Rank.	0,80	0,06	**0,05**†	0,35	0,39
	Medida F1	**10,85**†	5,81	4,05	6,72	6,46
Dez.	**Precisão**	**6,18**†	5,26	5,42	4,69	4,30
	Revocação	24,35	**24,61**†	21,16	22,83	22,01
	EE_SP	**8,00**†	**8,00**†	7,00	7,00	7,00
	EE_CP	**7,00**†	6,00	6,00	5,00	5,00
	Ganho_SP	**19,09**†	17,88	16,83	15,72	14,89
	Ganho_CP	**15,12**†	13,46	13,12	11,57	10,70
	Perc. Rank.	1,42	1,68	1,47	1,75	**1,84**†
	Medida F1	**9,86**†	8,67	8,63	7,78	7,19

classificadores tendem a melhorar com uma melhora da distribuição das classes minoritárias. Para o *LAC* o *ranking* potencializou seus resultados, mas, caso o *ranking* não tivesse sido utilizado, apenas com uma distribuição superior a 30% da classe minoritária é que seria possível a detecção de algum registro de fraude. No geral, é possível uma melhora na qualidade do classificador com uma distribuição da classe minoritária de até 50%. Acima disso, os resultados tendem a piorar significativamente, uma vez que o classificador irá considerar um maior número de registros legítimos como fraude, representando um queda expressiva na precisão.

6. **A eficiência econômica permite avaliar a qualidade dos resultados.** Geralmente as melhores eficiências econômicas estarão associadas aos melhores modelos. No entanto, ela pode tender para modelos que identificam casos de fraude de maiores valores. Sendo assim, é interessante que ela seja utilizada em conjunto com outras medidas de avaliação do modelo como a precisão e a revocação. Isso irá garantir que os melhores modelos com os melhores ganhos financeiros sejam selecionados.

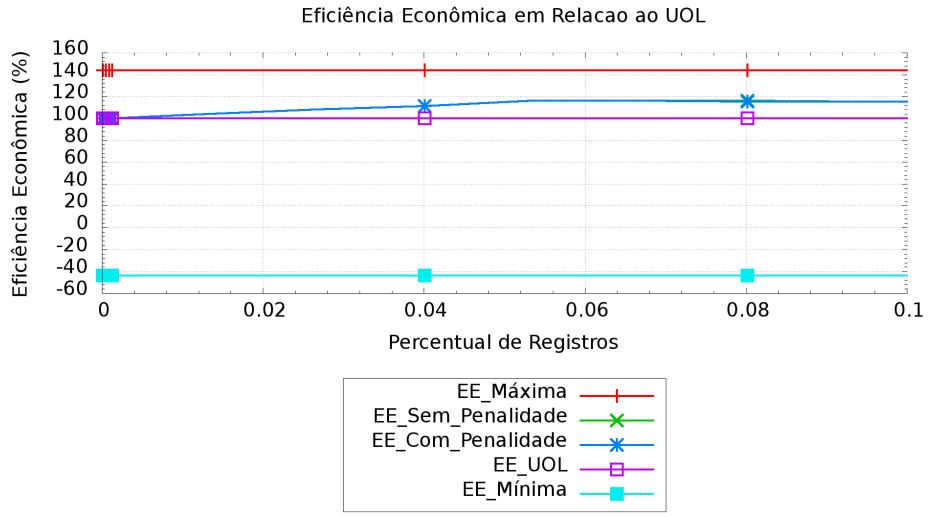

Figura 5.22. *Eficiência Econômica* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*.

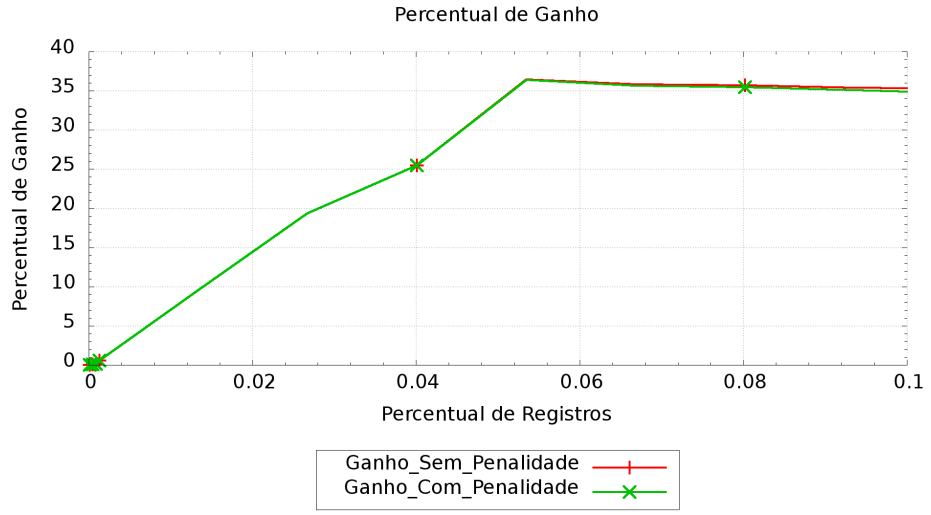

Figura 5.23. *Ganho em relação ao máximo* dos resultados obtidos em dezembro com o *oversampling* de 50% para o *Naive Bayes*.

7. **Qual técnica apresentou o melhor resultado?** Conforme apresentado na Tabela 5.3 a técnica *Stacking* apresentou o melhor resultado, atingindo ganhos de eficiência de até 46,46%. Uma das vantagens dessa técnica é que ela combina vários classificadores extraindo a melhor solução deles. Sendo assim, o resultado obtido pode ser melhorado combinando outras técnicas que se mostrem mais eficazes na detecção de fraude.

5.7 Considerações Finais

Este capítulo realizou um estudo de caso onde foi instanciada a metodologia proposta no Capítulo 4. Para a realização do processo de detecção de fraude foi utilizado um conjunto de dados real de uma das maiores empresas no Brasil de serviços de pagamentos eletrônicos, o *UOL PagSeguro*. Os passos definidos na metodologia foram seguidos permitindo a extração de um conjunto de dados do banco de dados Oracle com atributos relevantes para detecção de fraude. Este conjunto passou por um pré-processamento onde atributos pouco significativos foram retirados. Foram identificados padrões nos dados que representavam bons indícios da relevância dos atributos selecionados. Ao final, os dados foram transformados e novas características foram criadas, formando o *dataset* que foi utilizado para execução das técnicas de mineração. Os resultados obtidos com as técnicas mostram ganhos em relação ao máximo possível de até 46,46%. Isso mostra a relevância do *dataset* extraído e a eficácia das técnicas empregadas. Além disso, os resultados mostram que a metodologia sugerida é eficiente para o processo de detecção de fraude. O próximo capítulo irá apresentar uma visão geral sobre o trabalho realizado, além das conclusões e trabalhos futuros.

Capítulo 6

Conclusão e Trabalhos Futuros

Devido aos prejuízos financeiros causados pelas fraudes em pagamentos *online* com cartão de crédito, este trabalho propôs uma metodologia para sua detecção considerando o cenário de comércio eletrônico. Para que seja possível a criação ou utilização de métodos e técnicas eficazes para o combate à fraude, é necessário o entendimento de como essas fraudes ocorrem e as formas de prevenção utilizadas atualmente pelos serviços *online*. Sendo assim, foi apresentada uma ampla discussão sobre as fraudes de cartão de crédito em compras feitas pela *Web*. Foram abordados assuntos relacionados ao funcionamento do cartão de crédito, identificado as entidades envolvidas, apresentado os tipos de fraudes existentes e os impactos associados. Foi descrito também o processo de gestão de risco utilizado por médias e grandes empresas que tem como objetivo tornar as compras na *Web* mais seguras.

Além disso, a popularização do comércio eletrônico faz com que as vendas pela Internet envolvam milhares de transações ao longo do dia, o que torna inviável a análise manual de cada uma delas para que seja possível decidir em um curto espaço de tempo, se uma transação é ou não fraudulenta. Outro ponto a se considerar é que este também é um problema de solução não trivial, já que além do grande volume de dados envolvidos, as transações de fraude apresentam uma frequência muito baixa em relação às transações legítimas.

A metodologia proposta para detecção de fraude fornece teorias e ferramentas que auxiliam na resolução dos problemas associados a essa complexa tarefa de classificação. Ela é baseada no processo de Descoberta do Conhecimento em Banco de Dados, sendo composta por cinco passos com o objetivo de auxiliar o processo de detecção de fraude desde a obtenção dos dados para a formação de um *dataset*, passando pela escolha de técnicas mais promissoras, até a avaliação dos resultados obtidos. No passo de seleção dos dados uma grande contribuição foi a criação de um método para

extração de dados de um banco relacional. Foi proposto também um processo para auditoria da integridade dos dados à medida que o método de extração é realizado. No passo de pré-processamento, além da apresentação de técnicas para limpeza dos dados, foram apresentadas formas de visualização dos dados que permitam identificar padrões que podem representar o sucesso no processo de detecção de fraude. No passo de transformação foi apresentado um conjunto de variáveis que podem ser derivadas do *dataset* original e que são possíveis preditoras de fraude. No passo de mineração de dados foram apresentadas as várias tarefas possíveis para essa etapa e como elas podem auxiliar no combate a fraude. Optou-se pela tarefa de classificação, pois envolve a construção de modelos para detecção de fraude, sendo que essa escolha também foi necessária uma vez que cada uma das tarefas envolvem uma ampla pesquisa associada. Decidiu-se por utilizar principalmente técnicas em que o modelo gerado também seja descritivo para que seja possível um melhor entendimento dos padrões de fraude. As outras técnicas utilizadas como meta aprendizado e *oversampling* foram escolhidas no sentido de melhorar os resultados das técnicas escolhidas inicialmente. No último passo, interpretação e avaliação dos resultados, são apresentadas medidas para se verificar a qualidade dos resultados obtidos com as técnicas de mineração de dados, além de se verificar se o *dataset* extraído é relevante para a detecção de fraude. Uma grande contribuição nessa etapa foi a criação de um método para avaliação dos resultados que permite, na maioria dos casos, potencializar os resultados obtidos com o classificador por meio da utilização de um *ranking* das probabilidades de fraude. Foi criada também uma medida para o cálculo da eficiência econômica que indica o retorno financeiro obtido por cada modelo de classificação.

Para verificação da eficácia da metodologia proposta, foi feito um estudo de caso em um conjunto de dados real de uma das maiores empresas no Brasil de serviços de pagamentos eletrônicos, o *UOL PagSeguro*. Os passos da metodologia foram seguidos, em que foram extraídos da base de dados um conjunto de dados contendo atributos considerados relevantes para detecção de fraude. Este conjunto passou por um pré-processamento onde informações irrelevantes foram retiradas, bem como foram identificados padrões nos dados que mostravam bons indícios da relevância dos atributos selecionados. Por fim, os dados foram transformados e novas características foram criadas formando o *dataset* final que foi utilizado para execução das técnicas de mineração. Todas as técnicas sugeridas pela metodologia foram testadas baseadas no método de avaliação proposto. Os resultados alcançados nessa avaliação foram bons, onde todas as técnicas mostraram ganhos em eficiência econômica superiores aos alcançados pelo *UOL PagSeguro* no modelo atual de funcionamento. Dentre os melhores resultados pode-se citar o ganho de 46,46% obtido pela técnica de *Stacking*, o ganho

de 36,42% obtido com a técnica de *oversampling* para o *Naive Bayes* e o ganho de 18,08% obtido com a técnica *Naive Bayes*. Esses valores representam a porcentagem em relação ao ganho máximo possível de ser obtido, sendo o máximo representado por 100%. É possível perceber que são valores expressivos, que podem representar uma economia significativa para o *UOL PagSeguro*. Um ponto importante de se destacar é a ótima precisão obtida com a técnica *Stacking* com um valor de 97,71%, mostrando que essa combinação de classificadores é uma boa opção para o processo de detecção de fraude.

Os resultados obtidos mostram que a metodologia proposta é bem aplicável para o processo de detecção de fraude, mostrando também que ela é útil para extração de conjuntos de dados contendo atributos relevantes para essa tarefa. Como trabalhos futuros, objetiva-se uma nova execução dessa metodologia para a seleção de mais atributos que sejam significativos para o processo. Objetiva-se também a definição de novos métodos e técnicas que possam aprimorar esse processo de seleção. Além disso, pretende-se avaliar novas técnicas de mineração de dados com o objetivo de se identificar aquelas mais promissoras para a detecção de fraude. Por conseguinte, as técnicas que se mostrarem mais eficientes poderão ser utilizadas na técnica de *Stacking* para uma melhora ainda maior dos resultados obtidos. Com o conhecimento adquirido durante todo esse processo, pretende-se então definir novas abordagens em mineração de dados que sejam mais eficientes para detecção de fraude.

Referências Bibliográficas

Abbott, D. W.; Matkovsky, I. P.; Elder, J. F. & IV (1998). An evaluation of high-end data mining tools for fraud detection. Em *IEEE International Conference on Systems, Man, and Cybernetics*, pp. 12--14. IEEE.

Adriaans, P. & Zantinge, D. (1996). *Data Mining*. Addison-Wesley.

Aleskerov, E.; Freisleben, B. & Rao, B. (1997). CARDWATCH: A neural network based database mining system for credit card fraud detection. Em *Computational Intelligence for Financial Engineering, Proceedings of the IEEE/IAFE*, pp. 220--226.

Alvarez, G. & Petrovic, S. (2003). A new taxonomy of web attacks suitable for efficient encoding. *Computers & Security*, 22(5):435–449.

Amo, S. (2004). Técnicas de mineração de dados. *XXIV Congresso da Sociedade Brasileira de Computação.*

Barse, E. L.; Kvarnström, H. & Jonsson, E. (2003). Synthesizing test data for fraud detection systems. Em *Proceedings of the 19th Annual Computer Security Applications Conference*, ACSAC '03, pp. 384--, Washington, DC, USA. IEEE Computer Society.

Bhatla, T. P.; Prabhu, V. & Dua, A. (2003). *Understanding Credit Card Frauds*. USA.

Boente, A. N. P.; Goldschmidt, R. R. & Estrela, V. V. (2007). Uma metodologia para apoio à realização do processo de descoberta de conhecimento em bases de dados. *RevISTa do Instituto Superior de Tecnologia em Ciência da Computação IST-Rio*, 1:9.

Bolton, R. J. & Hand, D. J. (2002). Unsupervised Profiling Methods for Fraud Detection. *Statistical Science*, 17(3):235--255.

Brause, R.; Langsdorf, T. & Hepp, M. (1999). Neural data mining for credit card fraud detection. Em *Proceedings of the 11th IEEE International Conference on Tools with*

Artificial Intelligence, ICTAI '99, pp. 103--, Washington, DC, USA. IEEE Computer Society.

Cabena, P.; Hadjinian, P.; Stadler, R.; Verhees, J. & Zanasi, A. (1998). *Discovering Data Mining: From Concepts to Implementation.* Prentice Hall Saddle River, New Jersey, USA.

Chang, C.-C. & Lin, C.-J. (2011). Libsvm: A library for support vector machines. *ACM Trans. Intell. Syst. Technol.*, 2(3):27:1--27:27. ISSN 2157-6904.

Chau, D. H.; P, S. & Faloutsos, C. (2006). Detecting fraudulent personalities in networks of online auctioneers. Em *In Proc. ECML/PKDD*, pp. 103--114.

Cios, K. J. & Kurgan, L. A. (2005). Trends in data mining and knowledge discovery. pp. 1–26. Springer.

Cios, K. J.; Pedrycz, W.; Swiniarski, R. W. & Kurgan, L. A. (2007). *Data Mining A Knowledge Discovery Approach.* Springer.

Cohen, W. W. (1995). Fast effective rule induction. Em *In Proceedings of the Twelfth International Conference on Machine Learning*, pp. 115--123. Morgan Kaufmann.

Domingos, P. & Pazzani, M. (1996). Beyond independence: Conditions for the optimality of the simple bayesian classifier. Em *Machine Learning*, pp. 105--112. Morgan Kaufmann.

Elkan, C. (1997). Naive bayesian learning, technical report cs97-557. Department of Computer Science and Engineering.

Elkan, C. (2001). Magical thinking in data mining: lessons from coil challenge 2000. Em *Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '01, pp. 426--431, New York, NY, USA. ACM.

Fawcett, T. (2003). "in vivo"spam filtering: a challenge problem for kdd. *SIGKDD Explor. Newsl.*, 5:140--148. ISSN 1931-0145.

Fawcett, T. & Provost, F. (1997). Adaptive fraud detection. data mining and knowledge discovery.

Fayyad, U. M. (1996). Data mining and knowledge discovery: Making sense out of data. *IEEE Expert: Intelligent Systems and Their Applications*, 11:20–25. ISSN 0885-9000.

Fayyad, U. M.; Piatetsky-Shapiro, G. & Smyth, P. (1996a). Advances in knowledge discovery and data mining. capítulo From data mining to knowledge discovery: an overview, pp. 1–34. American Association for Artificial Intelligence, Menlo Park, CA, USA.

Fayyad, U. M.; Piatetsky-Shapiro, G. & Smyth, P. (1996b). From data mining to knowledge discovery in databases. *AI Magazine*, 17(3):37–54.

Fayyad, U. M.; Piatetsky-Shapiro, G. & Smyth, P. (1996c). The kdd process for extracting useful knowledge from volumes of data. *Commun. ACM*, 39:27–34. ISSN 0001-0782.

Frank, A. & Asuncion, A. (2010). Uci machine learning repository.

Freund, Y. & Schapire, R. E. (1995). A decision-theoretic generalization of on-line learning and an application to boosting. Em *Proceedings of the Second European Conference on Computational Learning Theory*, EuroCOLT '95, pp. 23--37, London, UK, UK. Springer-Verlag.

Gadi, M. F. A. (2008). *Uma Comparação de Métodos de Classificação Aplicados à Detecção de Fraude em Cartões de Crédito*. São Paulo.

Grazioli, S. & Jarvenpaa, S. L. (2000). Perils of Internet fraud: an empirical investigation of deception and trust with experienced Internet consumers. *Systems, Man and Cybernetics, Part A, IEEE Transactions on*, 30(4):395--410.

Hall, M.; Frank, E.; Holmes, G.; Pfahringer, B.; Reutemann, P. & Witten, I. H. (2009). The weka data mining software: an update. *SIGKDD Explor. Newsl.*, 11(1):10--18. ISSN 1931-0145.

Han, J. & Kamber, M. (2005). *Data Mining: Concepts and Techniques*. Morgan Kaufman.

Japkowicz, N. & Stephen, S. (2002). The class imbalance problem: A systematic study. *Intelligent Data Analysis*, 6(5):429–449.

Júnior, J. F.; Veloso, A.; Jr., W. M. & Pereira, A. (2011). Fraud detection in electronic transactions. *Iadis International Conference WWW/Internet*, 1:333--339. ISSN 978-989-8533-01-2.

Kantardzic, M. (2002). *Data Mining: Concepts, Models, Methods, and Algorithms*. IEEE Press.

Kou, Y.; Lu, C.-T.; Sirwongwattana, S. & Huang, Y.-P. (2004). Survey of fraud detection techniques. Em *Networking, Sensing and Control, 2004 IEEE International Conference on*, volume 2, pp. 749--754 Vol.2.

Kumar, A. & Nagadevara, V. (2006). Development of hybrid classification methodology for mining skewed data sets - a case study of indian customs data. *Computer Systems and Applications, ACS/IEEE International Conference on*, 0:584–591.

Langley, P.; Iba, W. & Thompson, K. (1992). An analysis of bayesian classifiers. Em *In Proceedings of The National Conference on Arti Cial Intelligence*, pp. 223--228. MIT Press.

Larivière, B. & Van den Poel, D. (2005). Predicting customer retention and profitability by using random forests and regression forests techniques. *Expert Syst. Appl.*, 29(2):472--484. ISSN 0957-4174.

Lindqvist, U. & Jonsson, E. (1997). How to systematically classify computer security intrusions. *Security and Privacy, IEEE Symposium on*, 0:0154. ISSN 1540-7993.

Lundin, E.; Kvarnström, H. & Jonsson, E. (2002). A synthetic fraud data generation methodology. Em *Proceedings of the 4th International Conference on Information and Communications Security*, ICICS '02, pp. 265--277, London, UK. Springer-Verlag.

Maes, S.; Tuyls, K.; Vanschoenwinkel, B. & Manderick, B. (1993). Credit card fraud detection using bayesian and neural networks. Em *In: Maciunas RJ, editor. Interactive image-guided neurosurgery. American Association Neurological Surgeons*, pp. 261--270.

Magalhães, M. N. & de Lima, A. C. P. (2002). *Noções de Probabilidade e Estatística.* EdUSP.

Mannila, H. (1997). Methods and problems in data mining. Em *Proceedings of the 6th biennial International Conference on Database Theory (ICDT'97)*, Lecture Notes in Computer Science, Vol. 1186, pp. 41–55. Springer-Verlag.

Maranzato, R.; Pereira, A.; Neubert, M. & do Lago, A. P. (2010). Fraud detection in reputation systems in e-markets using logistic regression and stepwise optimization. *SIGAPP Appl. Comput. Rev.*, 11:14--26. ISSN 1559-6915.

Maria Izabela R. Caffé, P. S. P. e. J. A. B. (2011). Avaliação do algoritmo de stacking em dados biomédicos. *XXXI Congresso da Sociedade Brasileira de Computação (ISSN 2175-2761), XI Workshop de Informática Médica*, p. 10.

Metwally, A.; Agrawal, D. & Abbadi, A. E. (2005). Using association rules for fraud detection in web advertising networks. Em *Proceedings of the 31st international conference on Very large data bases*, VLDB '05, pp. 169–180. VLDB Endowment.

Mindware Research Group, C. (2011). *2011 Online Fraud Report*. California, USA, 12th annual edition edição.

Netmap (2004). Fraud and crime example brochure.

Ngai, E. W. T.; Hu, Y.; Wong, Y. H.; Chen, Y. & Sun, X. (2011). The application of data mining techniques in financial fraud detection: A classification framework and an academic review of literature. *Decis. Support Syst.*, 50(3):559--569. ISSN 0167-9236.

Pal, N. R. & Jain, L. (2005). *Data Mining A Knowledge Discovery Approach*. Springer.

Philip K. Chan, Wei Fan, A. P. & Stolfo, S. J. (1999). Distributed data mining in credit card fraud detection. *IEEE Intelligent Systems*, 14:67--74.

Phua, C.; Alahakoon, D. & Lee, V. (2004). Minority report in fraud detection: classification of skewed data. *SIGKDD Explor. Newsl.*, 6(1):50--59.

Phua, C.; Lee, V.; Smith-Miles, K. & Gayler, R. (2005). A comprehensive survey of data mining-based fraud detection research.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA. ISBN 1-55860-238-0.

S., A. & A., B. (1998). *Advanced Techniques in Knowledge Discoveryand Data Mining*. Springer.

Shearer, C. (2000). The CRISP-DM model: The new blueprint for data mining. *Journal of Data Warehousing*, 5:13–22.

Shetty, J. & Adibi, J. (2009). Enron dataset. Disponível em: http://www.isi.edu/~adibi/Enron/Enron.htm.

Simoudis, E. (1996). Reality check for data mining. *IEEE Expert: Intelligent Systems and Their Applications*, 11:26–33. ISSN 0885-9000.

Statnikov, A. R.; Wang, L. & Aliferis, C. F. (2008). A comprehensive comparison of random forests and support vector machines for microarray-based cancer classification. *BMC Bioinformatics*, 9.

Tan, P.-N.; Steinbach, M. & Kumar, V. (2009). *Introdução ao Data Mining: Mineração de Dados*. Ciência Moderna.

Thomas, B. (2009). *OCA - Oracle Database 11g Administrator - Study Guide*. Wiley.

Thomas, B.; Clergue, J.; Schaad, A. & Dacier, M. (2004). A comparison of conventional and online fraud. Em *CRIS'04, 2nd International Conference on Critical Infrastructures, October 25-27, 2004 - Grenoble, France*.

Torgo, L. (2010). *Data Mining with R: Learning with Case Studies*. CRC Press.

Vasiu, L. & Vasiu, I. (2004). Dissecting computer fraud: From definitional issues to a taxonomy. Em *Proceedings of the Proceedings of the 37th Annual Hawaii International Conference on System Sciences (HICSS'04) - Track 7 - Volume 7*, HICSS '04, pp. 70170.3--, Washington, DC, USA. IEEE Computer Society.

Veloso, A.; Meira, W. & Zaki, M. J. (2006). Lazy associative classification. Em *Int. Conf. on Data Mining*, pp. 645--654.

Whitrow, C.; Hand, D. J.; Juszczak, P.; Weston, D. & Adams, N. M. (2009). Transaction aggregation as a strategy for credit card fraud detection. *Data Min. Knowl. Discov.*, 18(1):30--55. ISSN 1384-5810.

Wikipedia (2010). *Cartão de crédito*. Wikipedia.

Williams, G. J. & Huang, Z. (1997). Mining the knowledge mine: The hot spots methodology for mining large real world databases.

Wong, N.; Ray, P.; Stephens, G. & Lewis, L. (2012a). Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results. *Information Systems Journal*, 22(1):53--76. ISSN 1365-2575.

Wong, N.; Ray, P.; Stephens, G. & Lewis, L. (2012b). Artificial immune systems for the detection of credit card fraud: an architecture, prototype and preliminary results. *Inf. Syst. J*, 22(1):53--76.

Written, I. W. & Frank, E. (2005). *Data Mining: Practical Machine Learning Tools and Techniques*. Elsevier.