UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA DE PRODUÇÃO
MESTRADO EM ENGENHARIA DE PRODUÇÃO

DAYANA CARLA DE MACEDO

COMPARAÇÃO DA REDUÇÃO DE DIMENSIONALIDADE DE DADOS USANDO SELEÇÃO DE ATRIBUTOS E CONCEITO DE FRAMEWORK: UM EXPERIMENTO NO DOMÍNIO DE CLIENTES

DISSERTAÇÃO

PONTA GROSSA
2012

DAYANA CARLA DE MACEDO

COMPARAÇÃO DA REDUÇÃO DE DIMENSIONALIDADE DE DADOS USANDO SELEÇÃO DE ATRIBUTOS E CONCEITO DE FRAMEWORK: UM EXPERIMENTO NO DOMÍNIO DE CLIENTES

Dissertação apresentada como requisito parcial à obtenção do título de Mestre em Engenharia de Produção, no Programa de Pós-Graduação em Engenharia de Produção, Universidade Tecnológica Federal do Paraná.

Orientadora: Profa. Dra. Simone Nasser Matos

Co-orientadora: Profa. Ms. Helyane Bronoski Borges

PONTA GROSSA

2012

Ficha catalográfica elaborada pelo
Departamento de Biblioteca da UTFPR. Campus Ponta Grossa
n.18/12

M141 Macedo, Dayana Carla de

Comparação da redução de dimensionalidade de dados usando seleção de atributos e conceito de *framework*: um experimento no domínio de clientes. / Dayana Carla de Macedo. Ponta Grossa, 2012.
135 f.: il.: 30 cm

Orientadora: Profa. Dra. Simone Nasser Matos
Co-orientadora: Profa. Ms. Helyane Bronoski Borges

Dissertação (Mestrado em Engenharia de Produção) - Programa de Pós-Graduação em Engenharia de Produção. Universidade Tecnológica Federal do Paraná. Campus Ponta Grossa.

1. Dimensionalidade - Redução. 2. Atributos - Seleção. 3. Framework. 4. Domínio Cliente. I. Matos, Simone Nasser. II. Borges, Helyane Bronoski. III. Título.

CDD 670.42

Ministério da Educação
Universidade Tecnológica Federal do Paraná
Campus Ponta Grossa

Diretoria de Pesquisa e Pós-Graduação
Programa de Pós-Graduação em Engenharia de Produção
Mestrado em Engenharia de Produção

TERMO DE APROVAÇÃO

COMPARAÇÃO DA REDUÇÃO DE DIMENSIONALIDADE DE DADOS USANDO SELEÇÃO DE ATRIBUTOS E CONCEITO DE FRAMEWORK: UM EXPERIMENTO NO DOMÍNIO DE CLIENTES

por

DAYANA CARLA DE MACEDO

Esta dissertação foi apresentada em 5 de março de 2012 como requisito parcial para a obtenção do título de Mestre em Engenharia de Produção. A candidata foi arguida pela Banca Examinadora composta pelos professores abaixo assinados. Após deliberação, a Banca Examinadora considerou o trabalho aprovado.

_____
Profª. Dra. Simone Nasser Matos
(Orientadora)

_____
Profª. Msc. Helyane Bronoski Borges
(Co-orientadora)

_____
Prof. Dr. João Carlos Colmenero
(Membro titular)

_____
Prof. Dr. Pedro Paulo de Andrade Júnior
(Membro titular)

_____
Prof. Dr. Pedro José Steiner Neto
(Membro externo)

- O Termo de Aprovação assinado encontra-se na Coordenação do Curso -

Dedico esse trabalho a mim, por toda disciplina e foco necessários para sua execução, os quais considero alicerces da vitória.

AGRADECIMENTOS

Confesso que o mestrado foi uma das etapas mais gratificantes e árduas da minha vida, pois cresci profissionalmente, pessoalmente e espiritualmente.

Descobri que pude chegar muito mais longe do que podia imaginar antes de findar essa etapa, pois aceitei a trabalhar com uma área no mestrado a qual nunca havia tido contato antes. Mas, por meio de dedicação, disciplina, foco e muita força de vontade consegui concluir essa pesquisa.

Dessa forma, utilizo esse espaço para agradecer a todas as pessoas que contribuíram diretamente ou indiretamente para a conclusão deste trabalho.

Primeiramente agradeço à Deus, pois guiou meus passos até aqui, sendo fonte e base de inspiração e sustentação.

Agradeço a minha orientadora Profª. Simone Nasser Matos, pois soube entender as minhas dificuldades e dúvidas durante a pesquisa, ainda mais se tratando de um assunto extremamente complexo para mim. Sempre esteve disponível a me orientar e esclarecer essas dúvidas por meio do seu conhecimento. Obrigada por toda orientação, contribuição e paciência.

A Profª. Helyane Bronoski Borges, pois orientou muitas das etapas da execução desse trabalho, esclarecendo dúvidas ao decorrer da elaboração e análise do experimento.

Os agradecimentos aos meus amigos e colegas de turma e a turma posterior do mestrado, pois durante esses três anos de pesquisa muitos contribuíram com sugestões, ideias e questionamentos.

Agradeço também aos professores e funcionários da pós-graduação, pois no decorrer desses anos estavam sempre dispostos a sanar e esclarecer eventuais dúvidas.

Foram muitas pessoas que juntamente comigo fizeram parte das tristezas, alegrias e prazeres que uma vida estudantil trás, seja por meio de orientações, conversas, conselhos, incentivos, estudos, entre outros.

Saindo do âmbito do mestrado, agradeço as pessoas que fazem parte da minha vida pessoal e que contribuíram para a realização deste.

Ao meu namorado, Ricardo Lirani, por estar sempre ao meu lado e entender a minha ausência que muitas vezes foi necessária para a execução dessa pesquisa.

A você agradeço ao apoio, amor, companheirismo constante e por estar sempre ao meu lado nos momentos felizes e difíceis.

A minha irmã, Daniele Cristina Chagas, por sempre acreditar e torcer por mim. O meu muito obrigado a você por sempre estar disposta a me ouvir e apoiar.

Enfim, agradeço a todos que me ajudaram e torceram para que eu chegasse ao fim de mais uma etapa na carreira acadêmica.

É com tristeza e já com saudade do mestrado que encerro esse espaço de agradecimentos, mas com a certeza que está foi uma das fases mais desafiantes da minha vida. Contudo, por outro lado tenho como sentimento novos desafios e etapas que irão surgir no decorrer da minha vida para o meu crescimento profissional e pessoal.

RESUMO

MACEDO, Dayana Carla de. **Comparação da redução de dimensionalidade de dados usando seleção de atributos e conceito de framework:** um experimento no domínio cliente. 2012. 136 f. Dissertação (Mestrado em Engenharia de Produção) – Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2012.

Os dados de clientes nas empresas são coletados e armazenados em um Banco de Dados e sua administração requer o uso de uma ferramenta computacional. A construção de um modelo de Perfil de Cliente a partir de um banco de dados requer o processo descoberta de conhecimento em uma base de dados. Essa busca de conhecimento e extração de padrões das bases de dados demanda a utilização de um aplicativo com capacidade analítica para extrair informações que estão implícitas e desconhecidas, porém, potencialmente úteis. Um Banco de Dados por meio do processo de recuperação é capaz de obter informações dos clientes, mas a dificuldade é de que esses sistemas não geram padrões. Estes Bancos de dados contêm uma quantidade expressiva de atributos, os quais podem prejudicar o processo de extração de padrões. Assim, métodos de redução de dimensionalidade são empregados para eliminar atributos redundantes e melhorar o desempenho do processo de aprendizagem tanto na velocidade quanto na taxa de acerto. Também identificam um subconjunto de atributos relevantes e ideal para uma determinada base de dados. Os dois métodos de redução utilizados nesta pesquisa foram: Seleção de Atributos e Conceitos de *Framework*, até então não aplicados no domínio de Clientes. O Método de Seleção de Atributos tem o intuito de identificar os atributos relevantes para uma tarefa alvo na Mineração de Dados, levando em conta os atributos originais. Já os Conceitos de *Framework* promovem sucessivos refinamentos nos atributos que podem levar a construção de um modelo mais consistente em um domínio de aplicação. A presente pesquisa aplicou esses dois métodos para comparação destes no domínio Clientes, usando três bases de dados chamadas: *Stalog*, *Customer* e *Insurance*. Identificaram-se cinco etapas principais para a comparação dos dois métodos de redução: Preparação das Bases de Dados, Escolha das Bases de Dados, Aplicação dos Métodos de Seleção de Atributos e dos Conceitos de *Framework*, Execução dos Algoritmos de Classificação e Avaliação dos Resultados. Com a operacionalização das cinco etapas, compostas por vários processos, foi possível comparar os dois métodos e identificar os melhores algoritmos que aumentam a taxa de acerto dos algoritmos classificadores e consequentemente gerar os atributos mais relevantes para uma base de dados, aumentando o desempenho do processo de aprendizagem. Desta forma, com os melhores subconjuntos identificados é possível submetê-los a aplicação de tarefas da Mineração de Dados as quais permitem a construção de regras que ajudam na Gestão do Conhecimento do Perfil do Cliente.

Palavras-chave: Redução de Dimensionalidade. Seleção de Atributos. Framework. Domínio Cliente. Atributos.

ABSTRACT

MACEDO, Dayana Carla de. **Comparison of the dimensionality reduction of data using attribute selection and framework concept.** 2012. 136 p. Dissertation (Master Degree in Production Engineering) – Post Graduation Program in Production Engineering, Federal Technology University - Paraná. Ponta Grossa, 2012.

Information related to the Customers at companies are collected and stored in databases. The administration of these data often requires the use of a computational tool. The building of a Customer Profile model from the database requires the process of knowledge discovery in databases. This search of knowledge and extraction patterns of the databases demands the use of a tool with analytics capability to extract information that are implicit, and are previously unknown, but, potentially useful. A data base through of the recovery of date, obtain information of the Customers, but the difficulty is in the fact of these systems do not generate patterns. However, these databases have an expressive amount of data, where redundant information it prejudices this process of patterns extraction. Thus, dimensionality reduction methods are employed to remove redundant information and improve the performance of the learning process as the speed as in the performance of classifier. Furthermore, it identifies a subset of relevant and ideal attributes for a determinate database. The two methods of dimensionality reduction used in this search were: Attribute Selection and Framework Concepts which theretofore were not applied in Customer domain. The Attribute Selection Method has as goal to identify the relevant attributes for a target task, taking into account the original attributes. Considering the Framework Concepts it promotes successive refinements on the attributes where can tale he building of a model more consistent application domain. The present search applied these two methods in order to comparison of these in the Customer domain, using three databases called: Stalog, Customer e Insurance. This paper identified five main steps in order to comparison of the two methods: Preparation of Database, Choice of Database, Application of the Attributes Selection and Framework Concepts Methods, Execution of the Algorithms of the Classification and Evaluation of the Results. With the implementation of theses five steps composed of several processes, it was possible to compare the two methods and identify the best classifiers algorithms and consequently to create the attributes more relevant for a database, increasing the performance of the learning process. Of this way, with the best subset identified is possible submit them to the application of the Data Mining Tasks which allow the building of rules that help the Knowledge Management of Customer Profile.

Keywords: Dimensionality Reduction. Attribute Selection. Framework. Customer Domain. Attributes

LISTA DE FIGURAS

LISTA DE GRÁFICOS

LISTA DE QUADROS

LISTA DE TABELAS

LISTA DE SIGLAS E ACRÔNIMOS

BI	*Business Intelligence*
BW	*Business Warehouse*
CFS	*Correlation-basead Feature Selection*
CRM	*Customer Relationship Management*
CSE	*Consistency Subset Eval*
CSV	*Comma-separated values*
DHP	*Direct Hashing and Pruning*
GSP	*Generalized Sequential Pattern*
KDD	*Knowledge Discovery in Databases*
LVF	*Las Vegas Filter*
OLAP	*Online Analytical Processing*
LVI	*Las Vegas Incremental Filter*
ORB	*Object Request Broker*
SAP	*Systeme Anwendungen Produkte in der Datenverarbeitung*
SBG	*Sequential Backward Generation*
SFG	*Sequential Forward Generation*
SVM	*Support Vector Machines*
UML	*Unified Modeling Language*
WEKA	*Waikato Environment for Knowledge Analysis*

SUMÁRIO

1 INTRODUÇÃO

A Engenharia de Produção trata da engenharia especializada no desenvolvimento, aperfeiçoamento e implementação de projetos e ações que têm por objetivo fundamental a integração e a formação de inter-relação entre pessoas, informações, materiais, energia e equipamentos de maneira que respeite os pressupostos éticos e culturais da sociedade para a produção de bens e serviços da forma mais econômica possível (FLEURY, 2008).

O processo de gestão da informação pertence à área do conhecimento, que também é formada pela ciência da informação. Das subáreas da Engenharia de Produção, este trabalho encontra-se na Gestão do Conhecimento.

Esta pesquisa aborda a gestão da informação contida nas bases de dados das organizações sobre seus clientes, para o auxílio no processo de construção de base no domínio de Cliente.

No entanto administrar e aproveitar as informações coletadas sobre os clientes é um dos maiores desafios da organização, onde a adesão de uma ferramenta tecnologia da informação contribui para este processo de gestão e a obtenção de novos conhecimentos.

Fato este é importante, visto que o novo panorama competitivo exige a análise contínua de dados em busca de *insights*, para reconhecer e interpretar as tendências emergentes de mercado. Cabe aos gestores prover mecanismos para o processo de compreensão dessas informações e delas então extrair novas ideias (PRAHALAD; KRISHNAN, 2008).

Este processo de extração de *insights* fornece o entendimento do cliente para que a organização possa disponibilizar os produtos certos com o uso dos meios das maneiras certas (KOTLER; KELLER, 2006).

Nas organizações as informações coletadas sobre os clientes estão dispostas comumente em um Banco de Dados. Atualmente as empresas usam o modelo relacional e objeto relacional para a construção do Banco de Dados. O modelo relacional é fundamentado na representação dos dados em formato de tabelas. Por sua vez, o modelo objeto relacional permite relações fora da primeira forma normal e outras características de um modelo orientado a objetos (SILBERSCHATZ, KORTH, SUDARSHAN, 2006).

No entanto para extrair conhecimento e padrões de banco de dados no formato desses modelos atuais ocorre por meio de um sistema recuperação. Porém, a dificuldade é de que esses sistemas não geram padrões capazes de se obter conhecimento desconhecido e informação potencialmente útil (GALVÃO, 2009).

Nesse contexto, surge a questão da identificação de quais são os atributos necessários em relação a seus clientes que uma empresa ou organização necessita possuir em seu Banco de Dados? Ou como determinar quais os atributos mais relevantes de um banco de dados referente aos seus consumidores?

No entanto há a necessidade de selecionar os atributos mais relevantes (ROMDHANE; FADHEL; AYEB, 2010). Desta forma métodos de redução de dimensionalidade são usados para a identificação de atributos mais relevantes em uma base de dados.

Uma quantidade excessiva de atributos pode prejudicar a busca por padrões e extração de conhecimento útil. Segundo Kira e Rendell (1992), os atributos redundantes prejudicam o desempenho dos algoritmos de aprendizagem tanto na velocidade quanto na taxa de acerto.

Na Mineração de Dados algumas técnicas são usadas para a redução de dimensionalidade de atributos em bases de dados, dentre essas se destaca a Seleção de Atributos.

A redução de dimensionalidade também pode ser realizada por meio da aplicação dos conceitos de *Framework*, pois este promove sucessivos refinamentos nos atributos que levam a construção de um modelo mais consistente em um domínio de aplicação.

Nesse contexto, a presente pesquisa usou a Seleção de Atributos e Conceitos de *Framework* em bases de dados no domínio de Cliente, com o intuito de compará-los. Esses métodos foram aplicados em três bases de dados denominadas de: *Stalog*, *Customer* e *Insurance*.

A aplicação do Método de Seleção de Atributos foi feita por meio da abordagem Filtro e *Wrapper* no ambiente de Mineração de Dados, a saber, *Waikato Environment for Knowledge Analysis* (WEKA, 2011).

Na abordagem Filtro foram utilizados dois algoritmos: *Correlation-basead Feature Selection* (*CFS*) e *Consistency Subset Eval* (*CSE*). Em relação à *Wrapper* usaram-se os algoritmos classificadores: *Naive Bayes*, *J48* e *SVM*.

A aplicação dos Conceitos de *Framework* se deu por meio da adaptação da metodologia de Ben-Abdallah et al. (2004). Esse método foi escolhido porque estabelece um conjunto de relações e regras que facilitam a análise dos atributos.

O referido processo de aplicação destes métodos identificou cinco etapas que corroboraram para o estudo comparativo. Foram desenvolvidos algoritmos contendo: entradas, processos e saída a partir da segunda etapa.

Utilizaram-se como critérios de avaliação a Validação Cruzada por meio da média e desvio-padrão de taxa de acerto, média de atributos selecionados, com a finalidade de verificar se o uso da Seleção de Atributos e *Framework* contribuía para a melhora significativa dos valores de taxa de acerto quando comparado com as bases possuindo todos os atributos.

Dessa forma, com o intuito de comparar os Métodos de Seleção de Atributos e Conceitos de *Framework*, a caracterização do problema foi baseada na seguinte pergunta de partida: É possível reduzir a dimensionalidade dos dados e mesmo assim continuar a ter uma taxa de acerto desejável que varia de 70% à 100%?

A partir da problemática a hipótese básica e secundária caracterizou-se como: A hipótese básica: “O uso da Seleção de Atributos e Aplicação dos Conceitos de *Framework* para redução de dimensionalidade aumentam a taxa de acerto da base.” e como hipótese secundária “Os métodos de redução de dimensionalidade contribuem para a identificação dos atributos mais relevantes.”

1.1 JUSTIFICATIVA

Considerando as áreas da Engenharia da Produção a presente pesquisa está inserida na Gestão do Conhecimento, que é formada pela administração da informação envolvendo a Tecnologia e a Ciência da Informação.

Este processo de administração de informações tem como intuito a construção da base de conhecimento codificado e a gestão de pessoas, formado pelas áreas de filosofia, psicologia, sociologia e administração, para o processo de entendimento e dinâmica do processo de criação e difusão de conhecimento tácito.

A extração de conhecimento e de padrões de um Banco de Dados muitas vezes exige o uso de ferramentas computacionais. As aplicações destas

ferramentas contribuem para a identificação de compreensão do comportamento dos clientes.

As organizações quando possuem o foco no cliente estabelecem relacionamentos baseados no aprendizado de suas necessidades e desejos, oferecendo produtos adequados e assim mantendo relações de longo prazo com estes.

As informações referentes aos clientes em um Banco de Dados são representadas através de um conjunto de tabelas. Estas tabelas são constituídas de atributos e seus respectivos valores.

Porém as ferramentas comumente utilizadas na empresas empregadas ao gerenciamento dos dados não possuem como função a identificação dos atributos relevantes que levam extração de conhecimento para a geração de padrões.

Atualmente essas ferramentas disponíveis no mercado não têm como objetivo principal o processo de extração de conhecimento. Estes padrões são usados para verificação de tendências em relação ao mercado. Assim, evidencia a necessidade de se usar modelos quantitativos de análises de dados, de forma a transformá-los em conhecimento útil para a tomada de decisão (PRAHALAD, KRISHNAN, 2008).

O excesso de atributos e dados prejudica o processo de descoberta de conhecimento. Por isto, o uso de método de redução de dimensionalidade contribui para uma melhora dos dados, além de identificar os atributos mais relevantes de um determinado domínio. Os atributos considerados redundantes prejudicam o desempenho de algoritmos de aprendizagem no que se refere à velocidade e taxa de acerto (KIRA, RENDELL, 1992).

1.2 OBJETIVOS

O objetivo geral e os específicos deste trabalho estão descritos a seguir.

1.2.1 Objetivo Geral

Comparar as taxas de acertos dos Métodos de Redução de Dimensionalidade usando Seleção de Atributos e Aplicação dos Conceitos de *Framework* em bases de dados no domínio Cliente.

1.2.2 Objetivos Específicos

Os objetivos específicos estão relacionados a seguir:

- Levantar etapas para o desenvolvimento do experimento.
- Aplicar e Analisar as taxas de acerto do Método de Seleção de Atributos e Conceitos de *Framework* em cada uma das bases de dados.
- Usar a tarefa de classificação da Mineração de Dados para avaliar o Método de Seleção de Atributos e do Conceito de *Framework*.
- Identificar critérios para a realização da avaliação entre os métodos de seleção e os Conceitos de *Framework.*

1.3 ORGANIZAÇÃO DO TRABALHO

O presente trabalho está organizado em cinco capítulos. O Capítulo 1 apresenta a introdução, bem como a definição do objeto de pesquisa, justificativa, objetivos geral e os específicos.

O Capítulo 2 aborda a revisão da literatura, sendo dividida em três seções. A seção 2.1 refere-se à Mineração de dados, conceituando e detalhando as fases de sua operacionalização. Além disto, relata o tema Seleção de Atributos, estendendo-o para as abordagens existentes na literatura, tais como: o processo de busca por subconjuntos de atributos, as abordagens para avaliação de subconjunto de atributos e o processo de geração de um subconjunto. Esta seção também descreve a tarefa de Mineração de dados chamada classificação, bem como os algoritmos classificadores. A seção 2.2 trata a temática referente o *Framework*, apresentado os principais conceitos existentes na literatura, assim como o detalhamento de suas

classificações e alguns pontos relevantes com relação ao seu uso. A seção 2.3 aborda os assuntos referentes ao experimento, Perfil de Clientes, bem como sua relação na área de Engenharia de Produção.

O Capítulo 3 apresenta a metodologia utilizada para operacionalizar essa pesquisa, bem como o detalhamento de todas as etapas e processos aplicados.

O Capítulo 4 aborda os resultados obtidos pelo presente trabalho, tanto para o método de Seleção de Atributos como para *Framework*. Ainda nesse capítulo é efetuada uma comparação geral entre os resultados obtidos por meio da aplicação destes dois conceitos.

O Capítulo 5 relata as considerações finais dessa pesquisa e os trabalhos futuros que podem ser realizados a partir dos resultados obtidos.

2 REVISÃO DE LITERATURA

Este capítulo apresenta os conteúdos principais que nortearam o desenvolvimento da pesquisa, evidenciando os principais pesquisadores, trabalhos e conceitos publicados. A seção 2.1 descreve o processo de descoberta do conhecimento, bem como o detalhamento das suas etapas. Retrata também o processo de Mineração de Dados e a redução de dimensionalidade usando Seleção de Atributos. A seção 2.2 descreve o uso de *Framework*, descrevendo sua classificação, seus benefícios e métodos de criação. A seção 2.3 apresenta o tema do experimento, na área de Perfil de Cliente, assim como trabalhos relacionados.

2.1 PROCESSO DE DESCOBERTA DO CONHECIMENTO

O termo *KDD* (*Knowledge Discovery Database*) foi formalizado em 1989 em referência ao amplo conceito de procurar conhecimento a partir de base de dados.

Segundo Fayyad, Piatetsky-Shapiro e Smyth (1996), "*KDD* é um processo, de várias etapas, não trivial, interativo e iterativo, para identificação de padrões compreensíveis, válidos, novos e potencialmente úteis a partir de grandes conjuntos de dados".

O termo iterativo sugere a possibilidade de repetições integrais ou parciais do processo de *KDD* e a expressão não trivial alerta para a complexidade normalmente presente na execução de seus processos (GOLDSCHIMIDT; PASSOS, 2005).

À expressão "padrão válido" indica que o conhecimento deve ser verdadeiro e adequado ao contexto da aplicação. O termo "padrão novo" se refere ao acréscimo de novos conhecimentos aos existentes, gerando conhecimento útil que pode ser aplicado de forma a proporcionar benefícios ao contexto de aplicação (GOLDSCHMIDT; PASSOS, 2005).

A idéia geral de descobrir conhecimento em uma grande base de dados é atraente e intuitiva, porém isso tecnicamente dizendo é significamente difícil e desafiante (BRACHMAN; ANAND, 1996).

O conhecimento descoberto de base dados é a extração de informação não-triviais, implícita, previamente desconhecida e potencialmente útil de uma base de dados (FRAWLEY, PIATETSKY-SHAPIRO; MATHEUS, 1992).

Com o rápido crescimento de base de dados em muitos empreendimentos, a Mineração de dados tem se tornado uma abordagem importante para a análise de dados (OLAFSSON; LI; WU, 2008).

Por meio das tarefas de Mineração de Dados é possível extrair conhecimento de um grande volume de dados, descobrindo novas correlações, padrões e tendências entre as informações de uma empresa.

O processo de descoberta de conhecimento possui várias etapas operacionais, composto pelas etapas de: Pré-processamento, Mineração de Dados e Pós-processamento, as quais são descritas a seguir.

2.1.1 Pré-Processamento

Nesta etapa é necessário efetuar a seleção de dados, ou seja, selecionar identificar um conjunto de dados pertencentes a um domínio para que, a partir de um critério definido pelo especialista do domínio, possa ser analisado.

A seguir, encontram-se descritas as principais funções de pré-processamento dos dados segundo Boente, Goldschmidt e Estrela (2008):

- Seleção de dados: nesta função é necessário efetuar a identificação de quais informações da base de dados existentes devem ser efetivamente consideradas durante o processo de *KDD*.
- Limpeza de dados: Com o intuito de assegurar a qualidade relacionada a completude, veracidade e integridade é realizado o tratamento sobre os dados, ou seja, informações errôneas ou inconsistentes nas bases de dados devem ser corrigidas de forma a não comprometer o conhecimento a ser extraído no final do processo. Por exemplo, se um atributo apresentar um caractere não válido, quantidade de espaços a mais, para seu valor, o mesmo deve ser retirado ou substituído. As seguintes funções que podem ser aplicadas na limpeza de dados (GOLDSCHMIDT; PASSOS, 2005): limpeza de informações ausentes que compreende a

eliminação de valores faltantes em conjunto de dados, a limpeza de inconsistências a qual abrange a identificação e a eliminação de valores inconsistentes em conjunto de dados e por fim a limpeza de valores não pertencentes ao domínio que compreende a identificação e a eliminação de valores que não pertençam ao domínio dos atributos do problema. Atributo é um dado que é associado a cada ocorrência de 1 (uma) entidade de relacionamento.

- Codificação dos Dados: Para a utilização dos dados como entrada dos algoritmos de Mineração de Dados na forma correta, estes devem ser codificados, podendo ser: Numérica – Categórica, que transforma valores reais em categorias ou intervalos; ou Categórica – Numérica, que representa numericamente valores de atributos categóricos. Como exemplo, em uma base de dados um atributo sexo, pode assumir valores categóricos, sendo “masculino” ou “feminino” e para entrada deve ser codificado, como um inteiro, recebendo o valor 1 (feminino) e 2 (masculino).
- Enriquecimento dos dados: Têm como objetivo agregar mais informações aos instâncias existentes, enriquecendo os dados, para que estes forneçam mais informações ao processo de *KDD*. O analista pode definir quais atributos que deseja adicionar as informações já existentes para melhorar os resultados. Por exemplo, considerando uma base de clientes em que não se tenha como meta classificar cliente por renda, o atributo renda não existe neste momento. Neste caso, se o objetivo futuro for classificar os clientes por renda, à base será enriquecida com o atributo renda.

Durante o pré-processamento as restrições de espaço em memória ou tempo de processamento com relação ao número de atributos disponíveis para análise pode inviabilizar a utilização de algoritmos de extração de padrões. Nesse caso, é necessário aplicar métodos de redução de dados antes de iniciar a busca pelos padrões (REZENDE, 2003). Os métodos de redução são chamados redução de dimensionalidade e resulta da remoção de dados considerados redundantes e irrelevantes, permitindo uma melhor compreensão dos resultados gerados (BORGES, 2006).

Quando há criação de modelos de Mineração de dados no pré-processamento, é necessário efetuar a validação desses modelos. A validação é o processo de avaliação de como seus modelos são executados nos dados reais. Dessa forma, é relevante a validação dos modelos de mineração para o entendimento de suas qualidades e características antes da implantação desses em um ambiente de produção (MICROSOFT, 2011).

Há várias abordagens para avaliar a qualidade e características de um modelo de Mineração de dados. A primeira inclui o uso de várias medidas que possuem validade estatística para a determinação de problemas nos dados ou modelo gerado. A próxima etapa é a criação de conjuntos de treinamentos e de testes para o teste de exatidão das previsões, com intuito de revisar os resultados encontrados a fim de determinar se os padrões encontrados são significativos no cenário de negócios de destino (MICROSOFT, 2011).

Seguem alguns métodos indicados para a partição do conjunto de dados, para avaliação dos modelos de conhecimentos gerados segundo Goldschmidt e Passos (2005):

- *Hold-out:* Método que consiste em dividir de forma aleatória as instâncias em uma percentagem fixa denominada *p* treinamento e (1- *p*) denominada teste, onde é considerada normalmente $p > 0,5$. Porém, não há fundamentação teórica para essa porcentagem, usando-se comumente *p*: 2/3 e $(1 - p) = 1/3$ (REZENDE, 2003). Esta abordagem é muito utilizada quando se objetiva produzir um único modelo de conhecimento a ser aplicado posteriormente em algum sistema de apoio à decisão.
- Validação Cruzada com *K* Conjuntos (*K-fold CrossValidation*): Divide aleatoriamente o conjunto de dados com *N* elementos em *K* subconjuntos disjuntos (*folds*), com aproximadamente o mesmo número de elementos (*N*/*K*). Assim, cada um dos *K* subconjuntos é utilizado como conjunto de teste e os $(K - 1)$ demais subconjuntos são reunidos em um conjunto de treinamento. Dessa forma, ocorre repetição desse processo *K* vezes, sendo gerados e avaliados *K* modelos de conhecimento. Esse tipo de método é utilizado quando se objetiva avaliar a tecnologia utilizada na formulação do algoritmo de Mineração de Dados.

- Validação Cruzada com *K* Conjuntos Estratificada (*Stratifield K-fold CrossValidation*): Esse tipo de método se aplica em classificação, sendo muito similar a *Validação Cruzada com K conjunto,* em que a geração dos subconjuntos são mutuamente exclusivos. A proporção de exemplos em cada uma das classes é considerada durante a amostragem. Portanto, exemplificando, se o conjunto de dados original possui duas classes com distribuição de 20% e 80%, cada subconjunto também deverá conter aproximadamente esta mesma proporção de classes.
- *Leave-One-Out:* Conhecido como um caso de Validação Cruzada com *K* conjuntos, em que cada um dos *K* subconjuntos possui uma única instância. Computacionalmente é dispendioso e usado frequentemente em amostras pequenas.
- *Bootstrap:* O conjunto de treinamento é criado de *N* sorteios de forma aleatória e com reposição a partir do conjunto original de dados (contendo *N* instâncias). O conjunto teste é composto pelas instâncias originais que são sorteados com o conjunto de treinamento. Tem como intuito identificar uma média de desempenho do algoritmo de mineração.

A seguir se detalha a etapa de mineração de dados do processo de descoberta de conhecimento em base de dados, bem como as principais suas tarefas.

2.1.2 Mineração de Dados

Usa-se o termo Mineração de Dados para descrever a todos os aspectos de um processo automático ou semi-automático de extração do conhecimento potencialmente útil e desconhecido em uma grande base de dados.

Esse processo consiste de numerosas etapas, tais como: integração de dados de numerosas bases de dados, pré-processamento de dados e indução de um modelo com um algoritmo de aprendizagem. O modelo é usado para identificar e implementar ações para um determinado empreendimento (OLAFSSON; LI; WU, 2008).

Os dados contidos nas bases são usados para aprender um determinado conceito alvo ou padrão. Dependendo da natureza desse conceito, diferentes algoritmos indutivos de aprendizagem são aplicados. Os conceitos aplicados, denominado também de tarefas de mineração, são: associação, classificação, clusterização de dados, descoberta de regras de associação, entre outros (OLAFSSON; LI; WU, 2008). A seguir se descreve brevemente alguns deles:

- Classificação: Tem como objetivo encontrar uma função para mapear um conjunto de instâncias em conjunto de rótulos categóricos pré-definidos, denominados classes. Após descoberta essa função, esta pode ser aplicada a novas instâncias de forma a prever a classe que tais instâncias se enquadram. Exemplos de tecnologias que podem ser aplicados na tarefa de classificação são: Redes Neurais (DU; LAM, 2009), Algoritmos Genéticos (DIAS; PACHECO, 2005) e Lógica Indutiva (DUARTE, 2001).
- Descoberta de Associação: Também chamada de Regras de Associação permite o relacionamento da ocorrência de um determinado conjunto de itens com a ocorrência de outros conjuntos (YAMANOTO, 2009), ou seja, é a busca por itens que ocorram de forma simultânea frequentemente em transações do banco de dados. Algoritmos tais como: Apriori (AGRAWAL; SRIKANT, 1995) *Generalized Sequential Pattern (GSP*), *Direct Hashing and Pruning (DHP*), entre outros, são exemplos de ferramentas que programam a tarefa de descoberta de associações (GOLDSCHMIDT; PASSOS, 2005).
- Regressão: Semelhante à tarefa de classificação, analisam valores numéricos e possuem como principal objetivo apresentar uma previsão a partir de dados históricos contidos em uma base de dados. Compreende também a busca por uma função que mapeie as instâncias de um banco de dados em valores reais. Estatística, Redes Neurais, dentre outras áreas, oferecem ferramentas para implementação da tarefa de regressão (MICHIE; SPIEGELHALTER; TAYLOR, 1994).
- Clusterização ou análise de agrupamento: Por meio do agrupamento de dados os *clusters* são definidos baseados em medidas de similaridade ou modelos probabilísticos. Promove a associação de um item a uma ou várias classes categóricas em que as classes são determinadas pelos

dados. A análise de *cluster* tem como objetivo verificar a existência de diferentes grupos dentro de um determinado conjunto de dados, e em caso de sua existência, determinar quais são eles (FACELI, 2007). Existe um grande número de algoritmos descrito na literatura (ESTIVILL-CASTRO, 2002, XU; WUNSCH, 2005; FACELI, 2007).

- Sumarização: Consiste em identificar e apontar características comuns entre conjuntos de dados (GOLDSCHMIDT; PASSOS, 2005). Exemplos de tecnologias que podem ser aplicadas para implantar a tarefa de sumarização são: Lógica Indutiva e Algoritmos Genéticos.
- Detecção de Desvios: Foca a identificação de instâncias que não são considerados normais, ou seja, que não atendem ao padrão na totalidade do contexto da base de dados (BOENTE; GOLDSCHMIDT; ESTRELA, 2008).
- Descoberta de Sequências: É usada para encontrar padrões de dados escondidos numa sequência de estados temporais (GOLDSCHMIDT; PASSOS, 2005). É uma extensão da tarefa de descoberta de associações em que se buscam itens frequentes, considerando várias transações ocorridas ao longo de um período**.**
- Seleção de atributos: Segundo Kira e Rendell (1992) a seleção de atributos tem como objetivo descobrir um subconjunto de atributos relevantes para uma tarefa alvo, em que se consideram os atributos originais. É considerada importante por tornar o processo de aprendizagem mais eficiente. Na Mineração de dados são definidas as técnicas e os algoritmos a serem utilizados de acordo com o problema em questão. A seleção de atributos pertence a uma das técnicas da redução de dimensionalidade de base de dados, sendo um dos focos deste trabalho em que se pretende determinar os atributos mais relevantes por meio dessa tarefa.

Desta forma, detalhe-se a Seleção de Atributos na próxima subseção.

2.1.3 Seleção de Atributos

Atualmente se associa a Mineração de Dados à busca do conhecimento compreensível, útil e surpreendente em base de dados, e a aplicação dispensa a presença de um número significativo de atributos ou mesmo instâncias presentes nas bases de dados originais, em que certos casos se não forem removidos, podem até “atrapalhar” o processo de aprendizagem (BORGES, 2006).

Com relação à dimensionalidade, teoricamente, é intuitivo pensar que quanto maior a quantidade de atributos, mais informações estariam disponíveis supostamente para o algoritmo de mineração de dados. Porém, com o crescimento de atributos nos dados, esses tendem a ficar mais esparsos (DY, 2007). Essas dificuldades que são encontradas em espaços de muitas dimensões são denominadas no termo “maldição da dimensionalidade”.

Quando se trata de redução de dimensionalidade duas abordagens são comumente utilizadas: Transformação de Atributos e a Seleção de Atributos. Segundo Guyon e Elisseeff (2003) a Seleção de Atributos tem como objetivo a eliminação de atributos redundantes e não-informativos.

O processo de Seleção de Atributos permite a ordenação dos atributos de acordo com algum critério de importância, ou seja, a redução de dimensionalidade do espaço de busca de atributos e a remoção de dados contendo ruídos, entre outros (LEE, 2005). Já a Transformação de Atributos pode trazer diversos benefícios tais como: facilitação do entendimento, visualização dos dados e a redução do custo computacional do algoritmo de mineração de dados aplicado.

Com o intuito de solucionar ou amenizar tais problemas, utiliza-se métodos de seleção de atributos (CAVÕES, 2010).

O processo de seleção de atributos garante a qualidade com que os dados chegam à fase de mineração (LIU; SETIONO, 1996). Por meio da Seleção de Atributos se melhora a questão da qualidade dos dados e modelos que são construídos durante este processo, em que estes podem ser mais compreensíveis.

O desempenho do algoritmo de aprendizagem[1] é prejudicado tanto na velocidade (devido à dimensionalidade dos dados) quanto na taxa de retorno (devido

[1] O termo algoritmo de aprendizagem é comumente usado para representarmos os algoritmos da seleção de atributos.

às informações redundantes que podem confundir o algoritmo, não o auxiliando na busca de um modelo correto para o conhecimento) (KIRA; RENDELL, 1992).

Nesse contexto, faz-se necessário entender que para cada base de dados existe um subconjunto *M** de seus atributos que melhor caracteriza a base. Porém, para uma base de dados com *N* atributos existem $O(2^N)$ subconjuntos de atributos possíveis (CAVÕES, 2010).

A Figura 1 exemplifica os subconjuntos possíveis de atributos, quando $M = 3$.

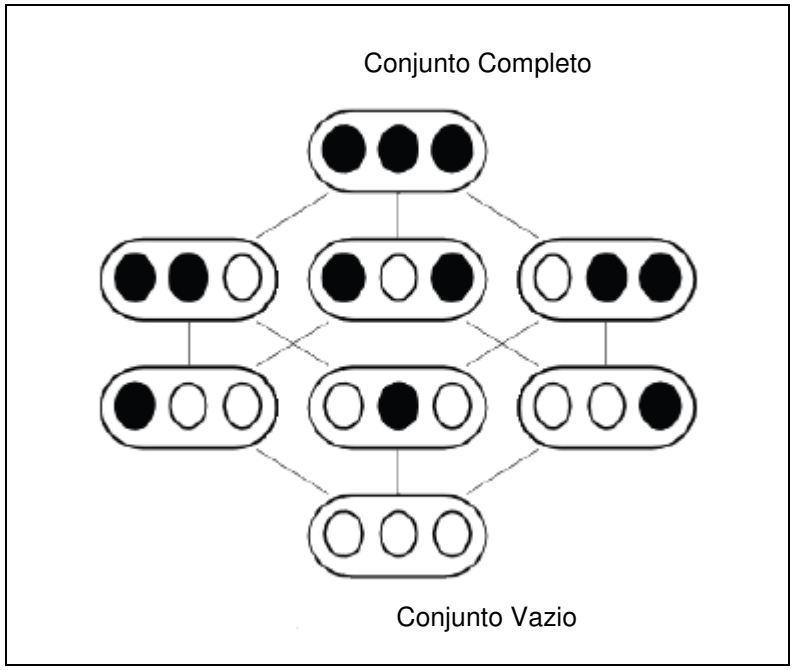


Figura 1 - Subconjuntos de Atributos Possíveis Considerando N = 3
Fonte: Cavões (2010, p. 10)

No topo se encontra o subconjunto com todos os atributos (todos os círculos preenchidos) e na base tem-se o subconjunto vazio (nenhum círculo preenchido).

Se a tarefa alvo for à classificação, a seleção de atributos buscará minimizar a taxa de erro do classificador, a complexidade do conhecimento a ser gerado por ele, e o número de atributos selecionados para compor a "nova" base (BORGES, 2006).

De acordo com Dash e Liu (1997) o método de Seleção de Atributos consiste de 4 (quatro) passos principais como mostra a Figura 2.

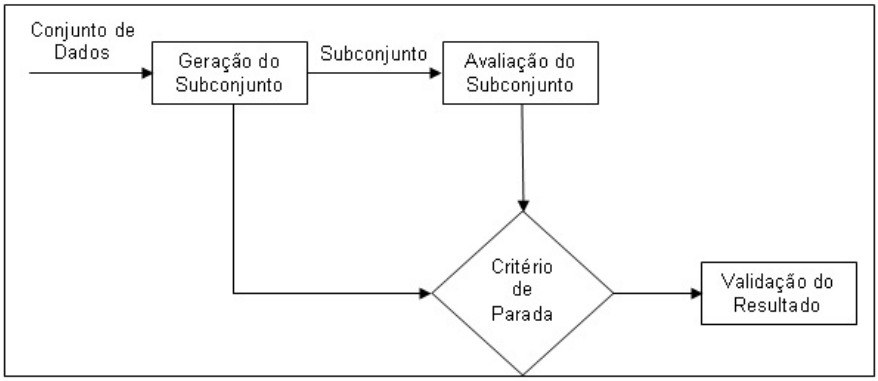


Figura 2 - Passos básicos do processo de seleção de atributos
Fonte: Dash; Liu (1997, p.133)

A seguir será abordado o método de busca por subconjuntos na seleção de atributos, conhecido também como Geração do Subconjunto. Além disto, descrevem-se, o processo de Avaliação do subconjunto, Critério de parada e a Validação do Resultado.

Geração do Subconjunto

No processo de geração do subconjunto há dois tópicos básicos (BORGES, 2006). O primeiro efetua a decisão do ponto (ou pontos) da busca que influenciaram na direção de busca. Com relação ao início da busca, essa pode começar com um conjunto completo e remover sucessivamente as características (para frente), ou iniciar com um conjunto completo e remover sucessivamente as características (para trás), ou ainda o início ocorre com ambos e adiciona e remove características simultaneamente (bidirecional).

Com a definição do ponto inicial da busca e direção, há a necessidade de determinar um procedimento para efetuar a busca. Segundo Hall (1999) a busca pode ser realizada por:

- *Greedy Hill climbing*: De acordo com algum critério, a cada iteração é determinado o atributo que melhora o subconjunto, ou seja, algum critério denominado de taxa do ganho de informação.
- *Best first*: Há similaridade ao *Greedy Hill climbing*, em que a avaliação do próximo conjunto é gerada de acordo com uma mudança local no subconjunto atual.

- *Algoritmos genéticos*: Aleatoriamente os subconjuntos são gerados baseando-se na teoria de evolução de Darwin (RUSE, 2009). Porém, os subconjuntos sofrem modificações de acordo com a distribuição de probabilidades. Uma competição entre os subconjuntos é a idéia básica, de tal forma que os melhores subconjuntos tenham chances de serem escolhidos.

Posteriormente a esse tópico, define-se a estratégia de busca, onde para um determinado conjunto de dados com N atributos, há 2^N subconjuntos candidatos. Mas, esse espaço de busca é exponencialmente proibitivo para a busca exaustiva mesmo com um moderado número de atributos. Por isto, faz-se necessário abordar as estratégias diferentes de busca que têm sido abordadas e exploradas, tais como (LIU; YU, 2005):

- Busca exponencial: Segundo o critério de avaliação é utilizada para garantir encontrar o resultado ótimo. Pela busca exaustiva, consegue-se realizar todas as combinações possíveis antes de retornar um subconjunto de características.
- Busca sequencial: Os algoritmos sequenciais, como a seleção sequencial para frente e para trás, são eficientes, porém tem a desvantagem de não levar em conta a interação entre atributos. Na seleção sequencial para frente, o processo se inicia pela busca do subconjunto considerado melhor de atributos com um conjunto vazio de atributos. Os subconjuntos de atributos com apenas um atributo são avaliados, e o melhor atributo A^* é selecionado. O atributo selecionado é então combinado com todos os demais atributos disponíveis (em pares), e posteriormente o melhor subconjunto de atributos é escolhido. A busca é efetuada continuamente, sempre adicionando um atributo por vez ao melhor subconjunto de atributos anteriormente selecionado até que não ocorra possibilidade de melhorar a qualidade do subconjunto de atributos selecionados. Na seleção de sequencial para trás, efetua-se a busca por um subconjunto de atributos ótimos com uma solução representativa para todos os atributos, e para cada iteração um atributo é removido da solução atual

até que não seja possível efetuar a melhora da qualidade na solução encontrada (BORGES, 2006).

- Busca aleatória: O processo de busca aleatória a partir de uma amostra inicia o processo de busca controlada com o intuito de melhorar esta amostra até que seja atingido o critério de parada (MOINO, 2006). A outra forma é obter o próximo subconjunto completamente aleatório, também conhecida como algoritmo simulação de Monte Carlo, proposto por Metropolis et al. (1953).

Ainda de acordo com Hall (1999), os algoritmos necessitam de um critério de avaliação. Critério de avaliação é usado para avaliar o subconjunto através de comparação, assim os subconjuntos de atributos gerados são analisados e comparados, e consequentemente um subconjunto é considerado ótimo. Aponta-se que existem diferentes critérios de avaliação, que podem levar a diferentes ótimos subconjuntos.

Como o presente trabalho efetua a Seleção de Atributos, faz-se necessário explicitar os principais algoritmos utilizados. Atualmente existe uma quantidade expressiva de algoritmos de seleção de atributos de características.

O Quadro 1 mostra as características comuns e as principais diferenças baseada nas estratégias de busca e critérios de avaliação (LIU; YU, 2005; LIU; SETIONO, 1996; HALL, 1999; KIRA; RENDELL, 1992; DEVIJVER; KITTLER, 1982; ALMUALLIM; DIETTERICH, 1991).

Algoritmo	Geração do Subconjunto	Geração de Sucessores	Medida de Avaliação
LVF	Aleatória	Aleatória	Consistência
LVI	Aleatória	Aleatória	Consistência
CFS	Sequencial	Para frente	Dependência
Relief	Aleatória	Ponderada	Distância
SBG	Sequencial	Para Trás	Qualquer
SFG	Sequencial	Para frente	Qualquer
Focus	Exponencial	Para Frente	Consistência

Quadro 1 - Característica dos Algoritmos de Seleção de Atributos
Fonte: Borges (2006, p. 31)

A escolha do algoritmo *CFS* foi baseada na diferença deste frente aos demais de abordagem filtro, pois fornece uma recompensa heurística do

subconjunto gerado, já os demais algoritmos desse tipo de abordagem somente resultam em um escore para cada atributo de forma independente. Para não se usar somente um algoritmo na abordagem filtro, aplicou-se também o algoritmo *CSE* para comparação dos resultados. A equação de mérito do algoritmo *CFS* é dada pela equação (HALL, 1999):

(1)

$$Merit_S = \frac{k\,\bar{r}_{cf}}{\sqrt{k + k\,(k-1)\bar{r}_{ff}}}$$

Onde:

$Merit_S$: é o mérito do subconjunto *S* de atributos contendo *K* atributos

$\bar{r}_{cf}$: é a média de atributos para a correlação da classe ($f \in S$)

$\bar{r}_{ff}$: é a média de atributos para a correlação do atributo.

Em relação ao algoritmo *CSE*, segundo Tan, Lim e Lai (2008), tem sido usado como uma métrica por várias abordagens para avaliação do subconjunto de atributo. Esta avaliação é feita para procurar combinações de atributos cujos valores dividem os dados em subconjuntos contendo a classe maior forte (KONONENKO, 1994).

O algoritmo usa a consistência métrica de acordo com os autores Liu e Setiono. (1996), como mostra na equação (2):

(2)

$$Consistency_S = \frac{1 - \sum_{i=0}^{J} |D_i| - |M_i|}{N}$$

Onde:

s: é um subconjunto de atributos

J: é o número de combinações distintas de valores de atributos para *s*

$|D_i|$: é o número de ocorrências da i-ésima combinação de atributos de valores

$|M_i|$: é a cardinalidade da classe majoritária para a combinação de valores de atributos

N: é o total de número de instâncias de um conjunto de dados.

Além dos dois algoritmos da abordagem Filtro explicados anteriormente, foram utilizados também os algoritmos na abordagem *Wrapper (Naive Bayes, J48 e SVM),* os quais serão detalhados na seção 2.1.2.2, pois além de serem usados na seleção de atributos também são aplicados na tarefa de classificação.

A seguir detalha-se o processo de avaliação de subconjuntos, o qual constitui a segunda etapa da Seleção de Atributos.

Abordagens para a Avaliação de Subconjuntos de Atributos

Segundo Liu e Yu (2005) a geração de subconjunto é realizada por meio de um procedimento de busca que produz subconjuntos de atributos candidatos para a avaliação, calcada em uma estratégia de busca. Cada subconjunto de atributos candidatos é comparado com um atributo anterior é avaliado. De acordo com um critério de avaliação, em que se o novo conjunto se tornar melhor, o mesmo será substituído pelo anterior.

Esse processo de geração e avaliação de subconjuntos é realizado repetidamente até satisfazer um determinado critério de parada. Nesse contexto, o melhor subconjunto selecionado deve ser avaliado por um conhecimento a *priori* ou a partir de testes diferentes em conjuntos de dados reais ou sintéticos (LIU; YU, 2005).

Os algoritmos de Seleção de Atributos foram desenvolvidos com diferentes critérios de avaliação, em que as abordagens são: Filtro e a *Wrapper*. As abordagens para avaliar os subconjuntos de atributos são categorizadas, de acordo com a participação do algoritmo de aprendizado na avaliação (GUYON; ELISSEEFF, 2003).

Segundo Borges (2006), as duas principais abordagens são: Filtro e a *Wrapper*. O modelo Filtro é dependente de características dos dados gerais para efetuar a avaliação e selecionar os subconjuntos de características sem envolvimento de um algoritmo de mineração. Já o modelo *Wrapper* necessita pré-determinar um algoritmo de mineração utilizando seu desempenho como um critério de avaliação. Com o intuito de melhorar o desempenho do algoritmo de mineração, o modelo *Wrapper* procura por um melhor conjunto de características, porém se comparado ao modelo de filtro tende ser computacionalmente mais custoso.

Já para Cavões (2010), *Wrappers* são algoritmos de seleção de atributos que no seu processo de avaliar o subconjunto de atributos "empacotam"' o algoritmo de aprendizado. A avaliação dos subconjuntos de atributos ocorre diretamente pelo algoritmo de aprendizado em questão. Os resultados obtidos com o modelo de *Wrappers* são bons, mas custosos computacionalmente.

Segundo Kohavi e John (1997) na abordagem filtro quando não é utilizado o algoritmo de aprendizado na modelagem para a avaliação de subconjunto de atributos, são usados algoritmos que possuem propriedades intrínsecas dos dados, e usualmente são mais indicados para grandes bases de dados e mais eficientes computacionalmente do que *Wrappers*.

Quando há a combinação de abordagens *Wrapper* e filtro, resulta-se em abordagens híbridas Hruschka et al. (2005).

Com relação à abordagem filtro, dentre as técnicas para avaliar os subconjuntos gerados se destaca a medida de distância, de informação, de dependência e de consistência. Segundo Liu e Motoda (1998), um atributo é considerado importante ou dito importante quando se efetua a remoção da medida de importância em relação aos atributos restantes. Por isto, faz-se necessário o entendimento de algumas medidas de importância de atributos, em que são utilizadas para avaliar os atributos e determinar em relação a que esses são importantes, como segue (LEE, 2005):

- Medidas de Informação: Determinação do ganho da informação a partir de um atributo. A definição do ganho da informação de um atributo X_i é dada pela diferença entre a incerteza *a priori* e a incerteza a *posteriori* se considerando X_i. Dessa forma, o atributo X_i é dito preferido ao atributo X_j, se o ganho de informação a partir do atributo X_i é maior que a partir do atributo X_j. Exemplo de medida de informação é a entropia, que é uma medida da incerteza associada com a variável aleatória.
- Medidas de Distância: Chamada também de medidas de separabilidade, divergência ou discriminação. Para duas classes, um atributo X_i é preferido ao atributo X_j se X_i provê uma diferença maior que X_j entre as probabilidades condicionais das duas classes. Como exemplo para esse tipo de medida se tem a Euclidiana, ou seja, ocorre à comparação entre

cada valor de cada linha por meio da distância euclidiana que mede o quão longe uma ocorrência está da outra.

- Medidas de Dependência: Segundo Borges (2006), são conhecidas como medidas de correlação ou medidas de similaridade, em que essa medida possui a habilidade de predizer o valor de uma variável de um valor para outro. Lee (2005) qualifica a habilidade de predizer o valor de uma variável (atributo) a partir do valor de outra, ou seja, qualificam o quanto duas variáveis estão correlacionadas uma com a outra. A medida clássica de dependência é o coeficiente de correlação, onde pode ser utilizada para encontrar a correlação entre um atributo e a classe. Se a correlação de um atributo X_i com a classe C é maior que a correlação do atributo X_j com C, então X_i pode ser considerado mais importante que X_i. Segundo Hall (1999) um dos algoritmos que usa esse tipo de medida é *Correlation-basead Feature Selection* (*CFS*), em que se avalia a importância de um subconjunto de atributos, baseado na habilidade individual preditiva de cada atributo e o grau de correlação entre eles.
- Medidas de Consistência: Para Liu e Setiono (1996), a medida de consistência é diferente da medida anterior devido à forte dependência da informação da classe e o uso do *bias*[2] para efetuar a seleção de um subconjunto com poucos atributos. As medidas de consistências são fortemente dependentes do conjunto de treinamento e preferem hipóteses consistentes para que possam ser definidas a partir do menor número possível de atributos. Essas medidas encontram o subconjunto mínimo de atributos que satisfazem a proporção de inconsistência aceita, onde usualmente é definida pelo usuário. Um problema em relação a esse tipo de medida é que não consegue distinguir entre dois atributos que são igualmente bons e não detectam atributos redundantes. A inconsistência é definida quando duas instâncias são iguais, possuem os mesmos atributos, porém com rótulos diferentes para cada classe (BORGES, 2006).

[2] *Bias* vêm do termo tendências *biases*, onde para Mitchell (1997), "é qualquer critério, explícito ou implícito, diferente de rigorosas consistências com os dados, usado para favorecer uma hipótese sobre outra".

- Medidas de Precisão: Trata-se das tarefas de predição, no qual um determinado algoritmo de aprendizado e os diversos subconjuntos de atributos, aquele que maior precisão proporcionar ao modelo gerado, será selecionado. Essa medida é considerada normal a utilização do mesmo algoritmo que processará o conjunto de exemplos com os atributos selecionados para realizar a tarefa de selecionar atributos.

O critério de parada pode ocorrer quando o objetivo foi alcançado, quando a busca termina ou no momento em que um subconjunto é considerado suficientemente bom. Para caracterizar esse subconjunto a razão do erro da classificação deve ser menor que a razão de erro permitida para uma dada tarefa (BORGES, 2006). Descreve-se a seguir a última etapa do processo de Seleção de Atributos, a saber, Validação dos Resultados.

Validação dos Resultados

Para validação dos resultados, pode-se utilizar a razão do erro classificador como um indicador de desempenho, para um subconjunto de atributos selecionado, através do comparativo da razão do erro classificador treinado no conjunto completo de atributos e no subconjunto de atributos selecionado (BORGES, 2006).

Outra forma para validar os resultados é a medição direta do resultado usando um conhecimento *a priori* dos dados. Se os atributos relevantes são conhecidos previamente há a possibilidade de comparação do atributo conhecido com os atributos selecionados.

A seguir a próxima subseção irá abordar os principais conceitos e algoritmos da tarefa de classificação na mineração de dados, pois esta será usada na realização do experimento.

2.1.3.1 **Classificação**

A classificação tem como objetivo geral a construção de um modelo conciso de distribuição de um atributo, denominado classe, em função dos demais atributos.

De um conjunto de instâncias denominado de treinamento este modelo é construído e o resultado é usado para designar valores ao atributo classe de instâncias onde há somente conhecimento dos atributos preditivos (BASGALUPP, 2010).

Para Borges (2006) um dos principais objetivos na tarefa de classificação é maximizar a taxa de classificações corretas nos dados de teste, correspondendo à razão entre o número de exemplos corretamente classificados e o número total de exemplos disponíveis no conjunto de teste.

O modelo de classificação é conhecido como um classificador, sendo útil para os propósitos de análise descritiva e preditiva (BASGALUPP, 2010). Na análise descritiva o classificador tem o intuito de fornecer uma explicação das características dos objetos de diferentes classes. Como exemplo, esse tipo de classificador pode ser utilizado para passar a informação se clientes são bons pagadores. Já na análise preditiva o classificador é usado para a classificação de objetos que foram usados para predizer se um novo cliente será um bom ou mau pagador.

Para Goldschmidt e Passos (2005) a classificação é uma das tarefas mais importantes da Mineração de dados e populares. Essa tarefa pode ser compreendida como a busca por uma função que permite a associação correta de cada instância X_i de um banco de dados a um único rótulo categórico, Y_i, que é chamado de classe. Sendo uma vez identificada, essa função pode ser aplicada a novos instância de forma a prever a classe que se enquadram esses instância.

A Figura 3 mostra as associações entre instâncias de dados e classes.

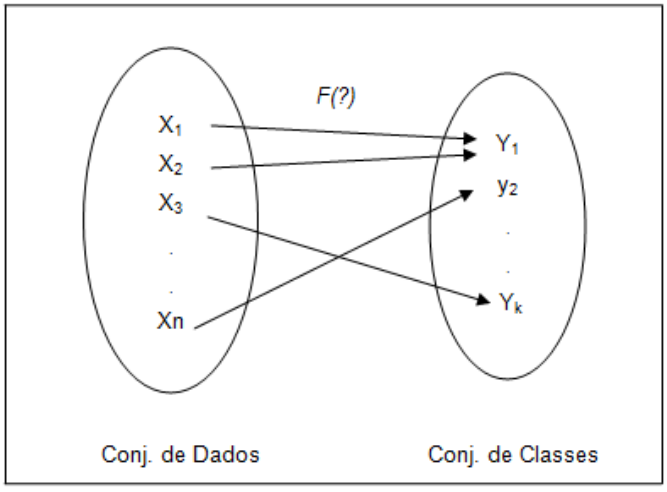


Figura 3 - Associações entre Instâncias de Dados e Classes
Fonte: Goldschmidt; Passos (2005, p. 67)

Há diversas técnicas de representar um classificador, tais como: árvores de decisão, redes neurais artificiais, algoritmos genéticos, redes bayesianas, fórmulas matemáticas, dentre outras. Porém, cada técnica usa um algoritmo de aprendizagem para a geração de um modelo que melhor se ajuste aos dados que são da composição do conjunto de treinamento, explicada na seção 2.1.1 (BASGALUPP, 2010).

O algoritmo de aprendizagem tem como principal objetivo a construção de um modelo que seja capaz de generalizar, predizer, com uma alta taxa de acerto, as classes dos objetos que não foram usados na construção do modelo (BASGALUPP, 2010).

Para Borges 2006 há vários algoritmos que podem ser empregados em bases de dados para a tarefa de classificação, porém vários são os paradigmas de aprendizagem, tais como: classificação baseada no Teorema de *Bayes*, em árvores de decisão, teoria estatística de aprendizagem, em instâncias. A seguir se reporta os principais algoritmos de classificação existentes na literatura.

Algoritmo de Classificação *Naive Bayes*

A classificação Bayesiana está entre as metodologias mais usadas e está calcada na teoria da probabilidade bayesiana (DUDA; HART; STORK, 2000).

Para Russel e Norvig (2004), as redes bayesianas são estruturas que representam as dependências entre as variáveis e fornece uma especificação concisa de qualquer probabilidade conjunta total, em que uma rede é um grafo orientado onde cada nó é identificado com informações de probabilidade quantitativa. Para esses autores a especificação completa de uma rede bayesiana é dada por:

> a) Um conjunto de variáveis aleatórias constitui os nós da rede e podem ser discretas ou contínuas;
>
> b) Um conjunto de vínculos orientados ou seta conecta pares de nós. Se houver uma seta do nó X até o nó Y, X será denominado pai de Y.
>
> c) Cada nó Xi tem uma distribuição de probabilidade condicional P que quantifica o efeito dos pais sobre o nó.

Os classificadores Bayesianos têm a capacidade de encontrar regras que respondem as perguntas do tipo (DUDA; HART; STORK, 2000):

- Qual a probabilidade de se jogar tênis dado que o dia está ensolarado, com temperatura quente, umidade alta e vento fraco? Levando em consideração termos probabilísticos essa pergunta equivale a *P* (JOGAR TÊNIS = Sim I [Ensolarado, Quente, Alta, Fraco]).
- Qual a probabilidade de NÃO se jogar tênis sendo que o dia está ensolarado, com a temperatura quente, umidade alta e vento fraco? Em termos probabilísticos essa pergunta equivale a *P* (JOGAR TÊNIS = Não I [Ensolarado, Quente, Alta, Fraco]).

Usa-se este método devido sua facilidade de interpretação e simplicidade, servindo de base de comparação para métodos mais complexos. Detalhes sobre esses métodos são apresentados em Duda, Hart e Stork (2000).

Os classificadores *bayesianos* para modelar o relacionamento entre atributos preditivos e a classe alvo um uma abordagem probabilística a qual é dividida em duas partes (HECKERMAN, 1997):

- Qualitativa: Representa o relacionamento de dependência entre um conjunto de variáveis, sendo um grafo acíclico dirigido.
- Quantitativa: Tabelas de probabilidades condicionais, onde há uma associação a cada nodo e seu relacionamento com outros.

A equação (3) traduz o teorema de *Bayes (*TAN; LIM; LAI, 2008*)*:

(3)

$$P(A|B) = \frac{P(B)}{P(A)}$$

onde:

$P(A)$: probabilidade do atributo A,

$P(B)$: probabilidade do atributo B, onde tudo que se tem conhecimento é o atributo B. Sendo que por meio do uso de probabilidades é possível o conhecimento da influência de um atributo sobre o outro.

De acordo com Tan, Lim e Lai (2008) as principais características das Redes *Bayesianas* são a representatividade, interpretabilidade e robustez. A característica representatividade se refere à representação das dependências causais entre atributos. Já a interpretabilidade diz respeito aos modelos criados que são de fácil interpretação e a robustez tem relação à acurácia dos modelos que não são afetados por eventuais ruídos ou dados redundantes.

A Figura 4 ilustra uma representação simples de uma rede *bayesiana*, onde há uma variável *t* que é independente das demais, já as demais são variáveis articuladas, sendo que *y* e *z* são condicionalmente independentes, dada a variável *x*.

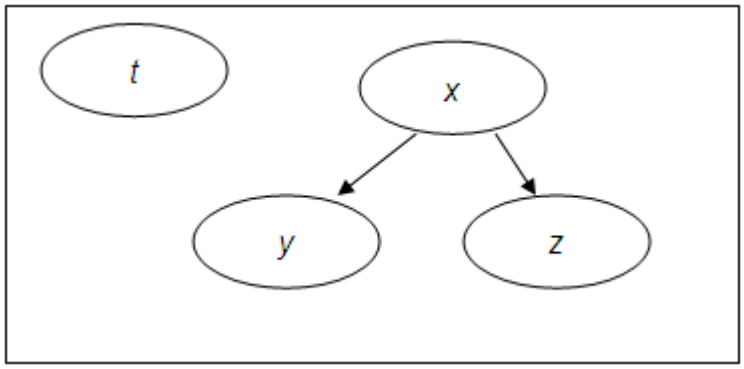


Figura 4 - Representação de uma rede bayesiana simples
Fonte: adaptado de Russel e Norvig (2004, p. 480)

Dos métodos existentes na literatura de aprendizado *bayesiano* se destaca o *Naive Bayes* também conhecido como Classificador *Bayesiano*, onde segundo Borges (2006), apresenta um bom desempenho se comparado com os demais algoritmos de aprendizagem de árvore de decisão, principalmente se combinado com o método de seleção de atributos para eliminação de informações consideradas redundantes. O algoritmo *Naive Bayes* apresenta como vantagem a simplicidade no seu cálculo, pois de maneira ingênua, admite independência entre atributos resultando na busca pela classificação que maximiza o produtório de sua equação (MITCHELL, 1997).

A equação (4) do classificador Naive Bayes é dada por (BORGES, 2006):

$$h_{NB} = \underset{h_j \in H}{\arg\max}\, P(\mathbf{h}_j) * \prod_i P(a_i | h_j) \quad (4)$$

Onde: A hipótese de *Naive Bayes* h_{NB} é a que maximiza o valor de um produto entre a probabilidade de ocorrência de uma hipótese h_j e um produtório de probabilidades das valorações dos *i*-ésimos atributos dada a hipótese h_j. Com relação ao nó do vetor na rede Bayesiana, a_i é o valor do atributo A_i. Já Π é outra unidade representativa do pi, ou seja, π.

Algoritmo de Classificação *J48*

O algoritmo J48 efetua a classificação baseada em árvore de decisão, e é um dos métodos mais usados para inferência indutiva (RUSSEL; NORVIG, 2004). A classificação baseada em árvores de decisão classifica as instâncias percorrendo uma árvore do nó da raiz até que se alcance a folha, em que cada um dos nós testa o valor de um único atributo e, para cada uma de suas valorações oferece arestas diferentes a serem percorridas na árvore a partir deste nó (MITCHELL, 1997).

Para Perissinotto (2007), o algoritmo *J48* constrói a árvore de decisão e posiciona o atributo considerado mais significativo, aquele que mais influencia a variável resposta, na raiz (início) da árvore. Posteriormente na sequência da construção, o próximo nó da árvore será o atributo considerado mais significativo, e assim de maneira sucessiva até que se gere o nó folha (final da árvore).

O conhecimento gerado é visualizado na forma de árvore, sendo representado também na forma de regras, o que torna mais compreensível e fácil a análise dos resultados. As regras originalmente definidas são generalizadas sendo representadas por *r*, onde algumas dessas são simplificadas ou até mesmo eliminadas. Dessa forma, alcança-se um conjunto ordenado de regras como resultado final (PERISSINOTTO, 2007).

Para Tavares, Bozza e Kono (2007) o algoritmo J48 constrói uma árvore de decisão, considerando o atributo mais significativo por meio da abordagem *top-down*, ou seja, o mais generalizado é considerado a raiz da árvore quando é comparado a outros atributos do conjunto. Assim, na sequência da construção, o próximo nó da árvore é o segundo atributo considerado o mais significativo. Esse processo é repetido por até que se gere o nó folha, que representa o atributo alvo da instância.

Normalmente há dois estágios de atuação do processo de geração de regras (TAVARES; BOZZA; KONO, 2007): as regras são induzidas e posteriormente

refinadas. Essa etapa é feita por meio de dois métodos: geração das árvores de decisão e o posteriormente pelo mapeamento da árvore em regras. Logo após, aplica-se o processo de refinamento pelo uso do paradigma “separar – para – conquistar”. A classificação baseada em árvore de decisão tem como vantagem a estratégia conhecida por “dividir - para - conquistar” em que se divide um problema maior em problemas menores (MITCHELL, 1997).

Algoritmo de Classificação *Support Vector Machines (SVM*)

O algoritmo chamado *Support Vector Machines* (*SVM*) foi desenvolvido a partir da teoria estatística de aprendizado de acordo com Cortes e Vapnik (1995). Tornou-se um dos mais promissores algoritmos de aprendizado no campo de gestão, incluindo classificação de texto, recuperação de informação e análise de clientes (SONG et al., 2009).

O uso do algoritmo *SVM*, comumente chamados *SVMs* é uma técnica atrativa pela habilidade de condensar informações contidas no conjunto de treinamento (SANTOS, 2002).

O objetivo de classificação usando o *SVM* é a elaboração de uma forma computacionalmente eficiente de aprendizado de “bons” hiperplanos de separação em um espaço de características de alta dimensão, onde os “bons” hiperplanos são considerados os que otimizam os limites de generalização e computacionalmente eficientes pois são capazes de tratar amostras de tamanho da ordem de 100.000 instâncias (LIMA, 2002).

Ao contrário de métodos tradicionais tal como as redes neurais em que minimizam somente o erro de treinamento empírico, o *SVM* tem como intuito minimizar um limite superior do erro de generalização, maximizando a margem entre a separação hiperplano e os dados (CORTES; VAPNIK, 1995).

Dessa forma, dado um conjunto de treinamento *E* com *n* pares (x_i, y_i), em que $x_i \in \Re^m$ e $y_i \in \{-1,+1\}$, o algoritmo *SVM* busca o classificador linear $g(x) = sgn(W * x + b)$ o qual possui a capacidade de separar dados que pertencem a *E* com erro mínimo e a margem *p* máxima de separação entre as classes presentes em *E*, conforme ilustra a Figura 5. A margem é considerada a distância de dois elementos de classes diferentes (CORTES; VAPNIK, 1995).

Para Borges (2006) dada uma função linear $f(x) = W * x + b$, com a margem $p(x_i, y_i)$, utilizada para a classificação de um padrão x_i é fornecida por $y_i f(x_i)$. Assim, ela efetua a medição de distância do padrão x_i em relação ao hiperplano separador.

A Figura 5 ilustra a margem geométrica de um ponto x_i e a margem p do hiperplano de separação ótimo.

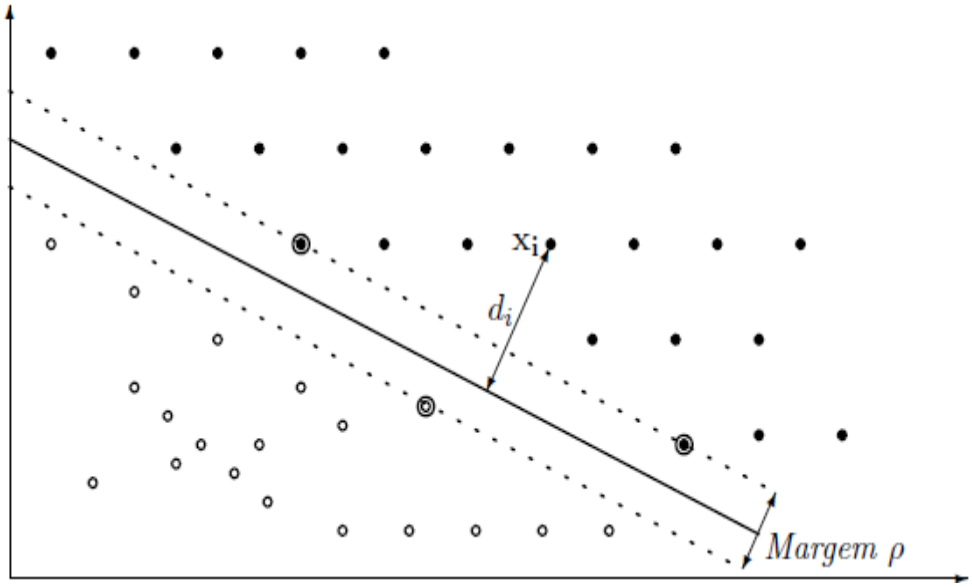


Figura 5 - Margem Geométrica de um ponto xi e a margem p do hiperplano de separação ótimo
Fonte: Borges (2006, p. 20)

Os círculos fechados são exemplos positivos e os círculos abertos são exemplos negativos. Estão sobre as margens (linhas tracejadas) são os SVMs para esse conjunto de treinamento, os quais estão realçados com um círculo mais externo (LIMA, 2002).

A seguir se detalha a etapa de pós-processamento, última etapa do processo *KDD*.

2.1.4 Pós-Processamento

A etapa de Pós-processamento compreende o processo de tratamento do conhecimento adquirido por meio da Mineração de Dados, com o intuito de facilitar a sua interpretação e avaliação.

Dentre as principais funções da etapa de Pós-processamento estão à elaboração e organização, podendo incluir a simplificação de gráficos, diagramas e

outros tipos de relatórios demonstrativos, além da conversão da forma de representação do conhecimento extraído no processo de *KDD* (BOENTE; OLIVEIRA; ROSA, 2007). A seguir encontram-se comentadas os métodos e procedimentos utilizados na etapa de pós-processamento (NICOLAIO, PELINSKI, 2006):

- Avaliação: Etapa onde o objetivo maior é a avaliação do conhecimento extraído da base de dados por meio de critérios, tais como: precisão, compreensibilidade e interessabilidade.
- Interpretação e Explanação: Consiste em tornar o conhecimento extraído compreensível ao usuário, ou seja, documentá-lo, visualizá-lo, modificá-lo e/ou compará-lo ao conhecimento pré-existente com o intuito de compreender melhor o conhecimento descoberto no processo de *KDD*.
- Filtragem: Consiste em filtrar o conhecimento extraído do conjunto de dados, realizado por meio de mecanismo que variam de acordo com a técnica utilizada, em que posteriormente na etapa de pós-processamento, com a análise do conhecimento, possa ser utilizado no processo de tomada de decisão.
- Interpretação: fase que inclui o processo de interpretação do modelo descoberto, em que se pode requerer a repetição de vários passos, porém normalmente é encarada como uma simples visualização dos dados. Os padrões identificados pelo sistema são interpretados em conhecimento, os quais podem ser utilizados como ferramenta de apoio ao processo de tomada de decisão.

Após a etapa de pós-processamento, o conhecimento extraído depois de avaliado e validado é consolidado na fase de utilização do conhecimento, sendo incorporado a um sistema inteligente, que é utilizado pelo usuário final para o apoio a algum processo de tomada de decisão, ou seja, relatado às pessoas interessadas.

2.2 *FRAMEWORK* DE APLICAÇÃO

Para Johnson e Foote (1988), um *Framework* sob o ponto de vista da estrutura é um “um conjunto de classes abstratas e concretas que formam o projeto abstrato para uma família de problemas relacionados”.

Sob o ponto de vista do propósito, define-se que *Framework* como sendo um esqueleto de uma aplicação que é instanciado pelo desenvolvedor de aplicações (JOHNSON, 1997).

Para Budd (2002) o *Framework* dá a possibilidade do reuso tanto de código, como de análise e de projeto. O reuso de análise é obtido porque descreve os objetos, seus relacionamentos e como grandes problemas são modularizados. O reuso de projeto ocorre quando o *Framework* contém algoritmos abstratos e a definição de suas interface, assim como os obstáculos que uma implementação precisa fazer. Na visão de Orfaly (1995) *Framework* é uma forma de reutilizar projetos de alto nível, que colaboram para colocar em prática um conjunto de responsabilidades.

Johnson (1997) reporta ainda que uma característica importante dos *Frameworks* é a inversão de controle. Segundo Silva et al. (2006), no desenvolvimento de um sistema tradicional baseado em componentes, o desenvolvedor toma um conjunto de componentes de uma biblioteca de reuso e o escreve em um principal programa que invoca esses componentes quando necessário. Neste caso, a responsabilidade é alicerçada no desenvolvedor pela estrutura e fluxo de controle do sistema, onde cabe a decisão de quando cada componente é invocado.

Ao desenvolver um sistema que é fundamentado em *Framework*, o principal programa é reusado. O desenvolvedor deve apenas decidir quais são os componentes que serão conectados a ele. Nesse caso, o código do desenvolvedor é invocado pelo *Framework* o qual determinará a estrutura e o fluxo de controle do sistema.

2.2.1 Classificação

De acordo com a visibilidade da sua estrutura interna um *Framework* durante a extensão ou instanciação e pode ser classificado como (JOHNSON; FOOTE 1988; YASSIN; FAYAD, 2000):

- *Black box* (caixa preta): Há a disponibilidade somente de sua interface. A forma como é implementada a funcionalidade interna não é visível ao usuário. O encapsulamento facilita seu uso, pois o usuário não necessita conhecer o funcionamento interno, porém reduz a sua flexibilidade e extensiabilidade pelo fato de não permitir ao mesmo adaptar-se às novas necessidades. Com relação ao reuso caixa-preta, também chamado de composição, o desenvolvedor não tem a necessidade de olhar o código interno para personalizá-lo, pois os componentes são selecionados e compostos. Ainda é relevante a informação de que os pontos de flexibilidade são preenchidos com a composição de objetos que os implementam. O usuário ao utilizar um reuso caixa-preta deverá ter entendimento somente da interface.
- *White Box* (caixa branca): possui estrutura interna visível ao usuário, permitindo estudar sua funcionalidade e fazer alterações implementadas externamente à classe do *Framework*. O uso desse tipo de *Framework* requer o conhecimento da funcionalidade interna. O processo de reuso é feito da criação de subclasses, onde detalhes internos referente a implementação das superclasses ficam expostos às suas subclasses.
- *Gray Box (*caixa cinza*)*: É um híbrido dos *Frameworks* caixa-preta e caixa-branca, onde o reuso é obtido por meio de herança, ligação dinâmica e pelas interfaces. Pode melhorar determinadas dificuldades, pois possuem maior flexibilidade e facilidade de extensão, onde não há necessidade de que fiquem expostas as informações que não são necessárias ao conhecimento dos desenvolvedores de aplicações.

Além da classificação em relação à visibilidade, um *Framework* pode ser classificado em relação ao propósito (FAYAD; SCHMIDT, 1997):

- *Frameworks* de infraestrutura do sistema: Usualmente são utilizados internamente em uma organização de software e não são comercializados diretamente com clientes. Como exemplos têm-se: sistemas operacionais, sistemas de comunicação, interfaces com o usuário e também ferramentas de linguagem. Esse tipo de *Framework* simplifica o desenvolvimento de sistemas portáteis e eficientes.
- *Frameworks* de integração de *middleware*: Foram projetados com o intuito de melhorar a habilidade dos desenvolvedores em modularizar, reutilizar e estender sua infraestrutura de software. Usados para promover a integração de aplicações e componentes distribuídos. Exemplo desta classe de *Framework* tem-se: o *Object Request Broker* (*ORB), middleware* voltado a orientar as mensagens e bases de dados transacionais.
- Os *Frameworks* de aplicação empresarial: Destinam-se a domínios mais amplos de aplicação e fundamentais para atividades de negócios das organizações, tais como: sistemas de telecomunicações, aviação, manufatura e engenharia financeira.

Com relação ao desenvolvimento os *Frameworks* são classificados como procedurais e orientados a objetos. Os procedurais são aqueles que não se baseiam em conceitos que são provenientes do paradigma orientado a objetos (ORFALY, 1995). Os que são orientados a objetos são desenvolvidos fundamentados no paradigma orientado a objetos (KUBO, 2006).

2.2.2 Pontos Relevantes em Relação ao Uso de *Framework*

O uso de *Frameworks* acarreta benefícios às atividades de desenvolvimento de sistemas de informação tais como (KUBO, 2006):

- Com o desenvolvimento de novas aplicações há uma rapidez e um custo menor, pois utiliza componentes pré-fabricados e pré-testados. Não há necessidade de o desenvolvedor descobrir novas classes e projetar interfaces. Há a relevância em reescrever o comportamento de métodos

específicos de determinadas classes. A estrutura do programa e o fluxo de execução já estão especificados.

- Permitem a reutilização de código e projeto por meio de polimorfismo e/ou herança.
- Há uma redução na manutenção, onde os *Frameworks* acarretam a maior parte do código que é necessário às aplicações, tornando os custos de manutenção menor. Devido ao fato da herança, quando ocorre correção de um erro em um *Framework*, ou quando há adição de uma característica, imediatamente os benefícios são estendidos às novas classes (ORFALY, 1995).
- A partir de *Frameworks* já desenvolvidos há a possibilidade de desenvolver aplicações cada vez mais complexas e poderosas.

A seguir a próxima seção retrata os Métodos para o desenvolvimento de *Framework.*

2.2.3 Métodos para Desenvolvimento de *Framework*

Na literatura especializada há algumas abordagens utilizadas no desenvolvimento de *Frameworks* de domínio.

Segundo Matos (2009), o processo de desenvolvimento de *Frameworks* de domínio consiste em uma evolução iterativa de suas estruturas de classes, envolvendo atividades de identificação de classes, modelagem de cenários e identificação de estados de objetos, entre outras.

As abordagens existentes na literatura têm o intuito de auxiliar o processo de desenvolvimento de *Frameworks* de domínio e são comparadas com o objetivo de fornecer um levantamento de seus benefícios e também de suas dificuldades.

As abordagens comumente utilizada existentes na literatura são a de Landin e Nikalasson (1995), Braga (2003) e Ben Abdallah et al. (2004), as quais são descritas no Quadro 2.

Landin e Niklasson (1995)	Braga (2003)	Ben-Abdallah et al. (2004)
Esse modelo facilita encontrar inconsistências entre os requisitos e pode ser usado como base de teste. Tem a capacidade de criação de um modelo de casos de uso para o *Framework* e para a aplicação exemplo. Há a possibilidade de verificação do que é comum entre as aplicações, *Framework* base, e o que é específico para a aplicação-exemplo.	No momento de criar o padrão na linguagem de domínio, há a procura de identificar as classes bases das que não são. É usada uma notação para diferenciação das classes base das que não são. Com essa notação há uma melhora do entendimento e compreensão do desenvolvedor para a distinção das classes que pertencem ao *Framework*, pois os relacionamentos estão pré-determinados.	Ocorre o estabelecimento de um conjunto de relações e regras que ajudam a análise das aplicações-exemplo na criação do modelo *Framework*. Há definição dos conceitos de equivalência e generalização, entre os outros. Dessa forma, facilita a compreensão, de qual relação será utilizada.

Quadro 2 - Características das abordagens para desenvolvimento de *Framework*
Fonte: Landin e Niklasson (1995); Braga (2003), Ben-Abdallah et al. (2004)

A presente pesquisa usou como base o método de Ben-Abdalhah. Maiores detalhes sobre as outras abordagens encontram-se nos trabalhos de Landin e Nikalasson (1995) e Braga (2003).

O método de Ben-Abdalhah et al. (2004) foi escolhido porque estabelece um conjunto de relações e regras que facilitam a análise das aplicações-exemplo durante a criação do modelo de *Framework.* Também tem a capacidade de definir conceitos de equivalência e generalização. Este método foi adaptado para o presente experimento. Esse tipo de abordagem permite a modelagem de *Framework* através de 3 passos, os quais são independentes sendo unificação dos diagramas de casos de uso, unificação dos diagramas de classes e unificação dos diagramas de sequência.

O método de Ben-Abdalhah et al. (2004) trata a temática de diagrama de classes, onde se utiliza como critério de comparação o nome de classes, atributos e operações. A comparação dos casos de uso, classes e objetos é feita pelo nome. Nesse sentido, sendo um dos focos do trabalho a questão relacionada a atributos, este método trata especificamente dos atributos, este método trata especificamente dos atributos e será adaptado para atender às questões relacionadas a essa pesquisa e será descrito no Capítulo 3. Maiores informações sobre o Método de Ben-Abdalhah et al. (2004) podem ser encontradas no Anexo A.

2.3 PERFIL DE CLIENTE

O conhecimento dos clientes é obtido pelo máximo de informações sobre ele, desta forma, a organização pode traçar o seu perfil e consequentemente adequar melhor os seus produtos e serviços.

Quando a questão refere-se a clientes atualmente usa-se o *Customer Relationship Management* (*CRM*), que é a gestão do relacionamento do cliente e consiste em quatro dimensões que é a identificação do cliente, atração de cliente, retenção de cliente e desenvolvimento de clientes. Essas quatro dimensões compartilham o objetivo comum de criar e aprofundar o entendimento de clientes e maximizar o valor do cliente para a organização no longo prazo (NGAI; XIU; CHAU, 2009).

A dimensão identificação de clientes inicia com o mapeamento do público alvo ou dos já existentes. Envolve a análise de clientes que estão sendo perdido para concorrentes e como ganhá-los novamente. Após identificação dos potenciais clientes é necessário definir uma estratégia de atração, sendo a dimensão atração de clientes. Nesta dimensão a organização direciona esforços e recursos para atrair estes, usando o *Marketing* Direcionado. A dimensão retenção é a preocupação central do *CRM*, pois ocorre a comparação das expectativas dos clientes com a sua percepção de estar satisfeito com o produto ou serviço oferecido, fato este essencial para a retenção de clientes (KRACKLAUER; MILLS; SEIFERT; 2004). Por fim, a dimensão do desenvolvimento do cliente trata de expandir transações de intensidade, valores e rentabilidade individual.

A aplicação das quatro dimensões do *CRM* para ganhar conhecimento do cliente tais como: identificação, atração, retenção e desenvolvimento é um conjunto de processo que apóiam a estratégia do negócio para construção de relações de longo prazo e rentável (NGAI; XIU; CHAU, 2009).

Existem ainda cinco pilares do atendimento estratégico: identificar e traçar o perfil de clientes, segmentar os clientes em agrupamentos naturais, pesquisar sobre a indústria e as preocupações dos clientes, investir em tecnologia para oferecer soluções e o gerenciamento dos clientes por meio de um tratamento consistente.

Os clientes com determinados valores, idades, estilos de vida e recursos financeiros chamam a atenção para a relevância de ações que se transformarão em ferramenta estratégica nas empresas. A segmentação tem como foco identificar

diferenças relevantes entre os consumidores e formam grupos homogêneos, acarretando o foco em esforços no segmento de interesse.

De acordo com Kotler e Keller (2006), o entendimento do cliente fornece maiores condições para que a organização possa disponibilizar os produtos certos com o uso dos meios das maneiras certas.

A identificação de quem é o consumidor, sua posição atual ou pretendida na escala profissional, seu estilo de vida, suas referências para o consumo e o que espera com a aquisição de um produto ou serviço, entre outros, fazem parte de um levantamento complexo, porém essencial.

2.3.1 Relevância de Traçar o Perfil de Cliente na Engenharia de Produção

Segundo Sveiby (1998) o campo da Gestão do Conhecimento é classificado em termos de áreas do conhecimento e a relação dos níveis de percepção que caracteriza o processo.

A área do conhecimento é formada pela gestão da informação, um das áreas tecnologia e ciência da informação, para a construção da base de conhecimento codificado e a gestão de pessoas, formada pelas áreas de filosofia, psicologia, sociologia e administração, para entendimento da dinâmica do processo de criação e difusão de conhecimento tácito.

Nesse contexto, a presente pesquisa esta trabalhando com a gestão da informação com o objetivo de auxiliar o processo de construção de uma base de dados no domínio de Cliente.

Os dados transacionais transformados podem ser uma fonte valiosa de informações sobre os hábitos de compras dos clientes (ROMDHANE; FADHEL; AYEB, 2010).

Os profissionais da produção devem priorizar a satisfação dos clientes, pois quando existem falhas provenientes dos produtos ou serviços prestados, ocorrem custos denominados de “custos de falhas externas”, sendo esses que mais prejudicam a imagem da empresa, pois os produtos já estão nas mãos dos clientes.

Como um dos maiores desafios das organizações é administrar e aproveitar as informações coletadas sobre os clientes, a adesão de uma ferramenta tecnologia

da informação contribui para a administração das informações e a obtenção de novos conhecimentos.

O novo panorama competitivo exige a análise contínua de dados em busca de *insights*, para reconhecer e interpretar as tendências emergentes de mercado. Cabe aos gestores prover mecanismos para o processo de compreensão dessas informações e delas então extrair novas ideias (PRAHALAD; KRISHNAN, 2008).

Assim, quando a questão se refere aos clientes, o foco é estabelecer relacionamentos baseados no aprendizado de suas necessidades e desejos, oferecendo produtos certos e manutenção dessa relação ao longo do tempo para a obtenção da identificação com a marca e garantia da lealdade (GAVA et al., 2005).

Dados transacionais e comportamentais em um banco, tais como: tipos, volume; histórico de compras, reclamações em *call centers*, reivindicações, dados de acesso na *WEB* atualmente são usados para efetuar o processo de segmentação de cliente (LEE; PARK, 2005).

Fan, Gordon e Pathak (2005) salientam que perfis de usuários podem ser criados através do histórico de compras de clientes, tais como as categorias de produtos que os consumidores tem adquirido recentemente.

Como exemplo, a figura 6 ilustra uma estrutura de um banco de dados, mostrando as tabelas Pedidos, Clientes, Endereço, Itens e Produtos, usando a notação *Unified Modeling Language* (*UML)*. A notação UML é um padrão desenvolvido para criar especificações de vários componentes de um sistema de software (SILBERSCHATZ, KORTH, SUDARSHAN, 2006).

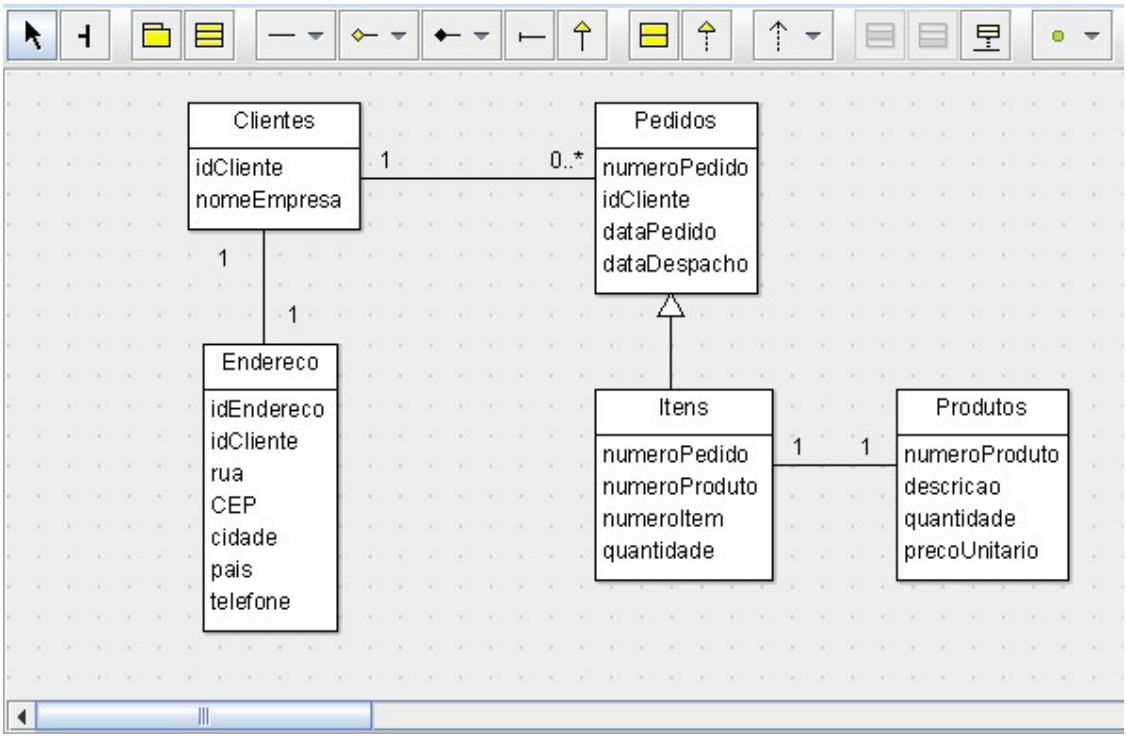


Figura 6 - Exemplo de uma Estrutura de um Banco de Dados Usando a notação UML
Fonte: adaptado de Serrato (2005)

Como é possível verificar a tabela Clientes trás informações com relação a sua identificação e nome da empresa. A tabela Pedidos contempla o número do pedido, bem como a identificação do cliente, data de aquisição e o despacho. A tabela Endereço contém informações relacionadas à localização do cliente. Por fim, as tabelas Itens e Produtos trazem informações dos itens adquiridos, como identificação do produto, preços e outros.

Um banco de dados por meio de um sistema de recuperação de dados consegue extrair informações de clientes, podendo ser obtidas por meio de categorização. Exemplificando, uma empresa que possui um banco de dados pode extrair um relatório informativo de quais são os clientes mais rentáveis e quais são os inadimplentes facilmente. Porém, a dificuldade é de que esses sistemas não geram padrões, diferentemente da Mineração de Dados.

Para a geração dos padrões há a necessidade do desenvolvimento de um conjunto de combinações de atributos capazes de satisfazer às necessidades específicas de cada cliente (PRAHALAD; KRISHNAN, 2008). Esse processo de descoberta das necessidades dos clientes é possível, por exemplo, por meio da seleção dos atributos. Como um banco de dados possui um conjunto de tabelas,

como fazer essa combinação de atributos que possam satisfazer às necessidades de cada cliente?

A partir das tabelas ilustradas na Figura 7 se pode criar uma base com os atributos que respondem as indagações dos gestores. Por exemplo, Qual é a quantidade de item de produto vendida por cliente? A Figura 8 ilustra a base que responde à esta pergunta. Esta figura também está no formato da *UML*.

Esta base apresenta a junção de todos os atributos contidos em tabelas separadas, juntamente com informações a respeito de índices e relações. Esta constitui de uma base não-normalizada.

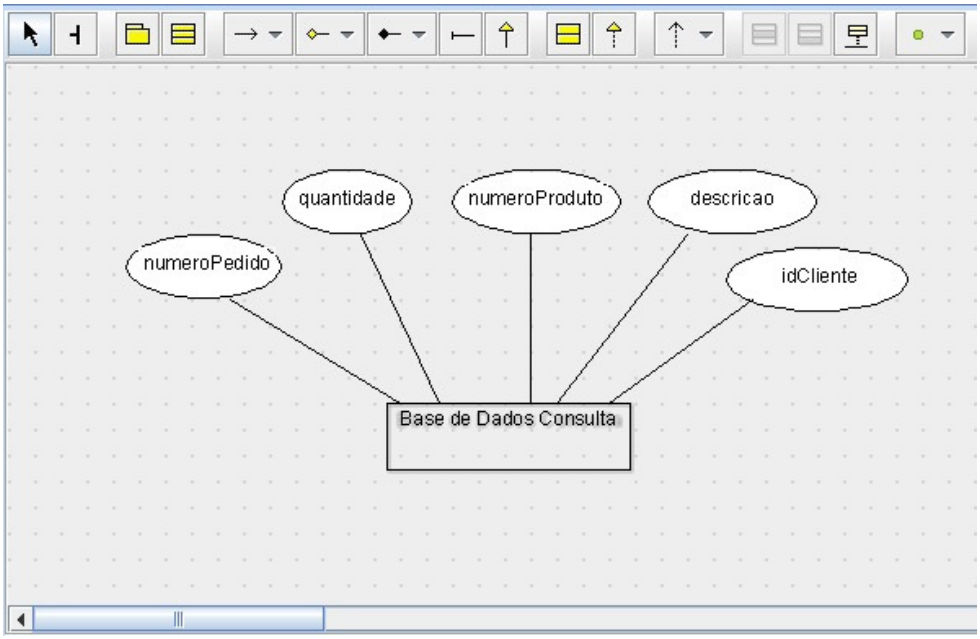


Figura 7 - Base de Dados Consulta
Fonte: Autoria própria

Nesse contexto, surge a questão da identificação de quais são os atributos necessários em relação a seus clientes que uma empresa ou organização necessita possuir? Ou como determinar quais os atributos mais relevantes de um banco de dados referente aos seus consumidores? Nesse sentido, por exemplo, a Seleção de Atributos é importante, pois os atributos considerados redundantes são eliminados porque prejudicam o desempenho de algoritmos de aprendizagem no que se refere à velocidade, taxa de acerto (KIRA, RENDELL, 1992).

A próxima seção aborda os trabalhos relacionadas com a presente pesquisa na área de mineração de dados e *Framework*, considerando a questão de redução de dimensionalidade de atributos.

2.3.2 Trabalhos Relacionados

Atualmente os bancos de dados seguem o modelo relacional e objeto relacional, em sua grande maioria. O modelo relacional é baseado na representação dos dados em forma de tabelas. Cada coluna de uma tabela possui um nome único, já uma linha do objeto representa um relacionamento entre um conjunto de valores. O modelo objeto relacional permite relações fora da primeira forma normal e outras características de um modelo orientado a objetos (SILBERSCHATZ, KORTH, SUDARSHAN, 2006).

O desenvolvimento de modelos, também chamados de perfis, é um passo importante para o *marketing* direcionado. Muitos gestores de *marketing* podem desenvolver relacionamentos de longo-prazo e agradáveis com seus consumidores se eles detectarem e predizerem mudanças em seus hábitos de consumo ou comportamento (ROMDHANE; FADHEL; AYEB, 2010).

Emprega-se o *CRM* quando a questão é o cliente e este consiste de quatro dimensões que é a identificação do cliente, atração de cliente, retenção de cliente e desenvolvimento de clientes. Muitas tarefas de mineração de dados são empregadas conjuntamente com o *CRM* visando o desenvolvimento dessas quatro dimensões (NGAI; XIU; CHAU, 2009).

Outra ferramenta utilizada é o *Business Intelligence* (BI) possui foco na tomada de decisão sintetizando métodos, técnicas e ferramentas para auxiliar a análise de dados e emitir retornos ou respostas que favorecem e contribuem de maneira inteligente e confiável as decisões da empresa (CARVALHO; FERREIRA, 2001). A tomada de decisão envolve aspectos globais de uma organização, não sendo somente os clientes desta. Ambas as ferramentas de banco de dados não têm como objetivo principal a extração de padrões.

Desta forma, muitas técnicas de Mineração de Dados têm sido empregadas, em que se usam algoritmos específicos para a extração de padrões. Estes padrões permitem obter conhecimento previamente desconhecido e informação potencialmente útil (CHEN; HEN; YU, 1996; MITRA; PAL; MITRA, 2002).

Recentemente a segmentação de cliente baseia-se em dados transacionais e comportamentais, tais como: tipos, volume; histórico de compras, reclamações em *call centers*, reinvindicações, dados de acesso na *WEB* (LEE; PARK, 2005). A partir

desses dados é possível obter os padrões que contribuem para a geração de conhecimento útil.

Fernandes (2006) sugere uma integração cada vez mais intensa entre disciplinas como o *Marketing* de Relacionamento, *Customer Relationship Management* (CRM) e Mineração de dados. Em seu estudo realizou uma pesquisa quantitativa em empresas do setor de serviços em São Paulo e do Rio de Janeiro para investigar o nível de realização das etapas do processo *KDD*, do ponto de vista de estratégias de relacionamento. Em seu trabalho foi constatado que as empresas não focam somente na retenção de clientes, mas concentram esforços nas estratégias de aquisição de novos clientes e identificação dos melhores clientes. Assim, evidencia a necessidade de modelos quantitativos de análises de informações sobre os clientes, de forma a transformá-las em conhecimento útil para a tomada de decisão.

Árvores de decisão são empregadas para extrair modelos e descrever sequencias de decisões interelacionadas ou predizer tendências futuras nos dados de clientes (BERRY; LINOFF, 2004; CHEN; HSU; CHOU, 2003; KIM et., 2005).

Também há o uso de regras de associação com a descoberta de potencias relações entre os dados, o qual permite construir um modelo para predizer o valor de uma cliente futuro (WANG et al.,2005).

Muitas técnicas de Mineração de dados são aplicadas para o desenvolvimento de cada um das dimensões do *CRM*. Na retenção de clientes são usadas técnicas como *Clustering*, Descoberta de Sequências, Classificação e Regras de Associação focando programa de lealdade e de reclamações de clientes, os quais são elementos do *CRM*.

Já com relação à dimensão desenvolvimento de clientes, empregam-se também as tarefas de classificação, *clustering*, regressão, associação e descoberta de sequencias. Estas técnicas são aplicadas nos elementos do *CRM* tais como: valor do custo de vida e *Market Basket Analysis,* a qual analisa o comportamento do cliente pelos hábitos de consumo.

Há aplicações bem sucedidas de Mineração de Dados em várias áreas incluindo *WEB, Marketing*, financeira e bancária, telecomunicação, entre outras. Romdhane, Fadhel e Ayeb (2010) desenvolveram uma abordagem para perfil de cliente composta de três etapas, onde a primeira consistiu do uso do algoritmo *FCM-*

based para agrupar os clientes, usando também a entropia da informação[3]. Na última etapa foi construído um conjunto de perfil de clientes através da rede neural chamada *backpropagation* (ROMDHANE; FADHEL; AYEB, 2010). O resultado da pesquisa desses autores foi à predição de hábitos de consumo e comportamento dos clientes, as organizações podem desenvolver relacionamentos de longo prazo e bem sucedidos com estes.

No trabalho de Park e Chang (2009) foi desenvolvido um modelo de perfil de cliente baseado na informação do comportamento individual e do grupo tal como os *clicks*, inserções em cestas de compras, compras e campos de interesse, também usando Mineração de Dados.

Jiang e Tuzhilin (2009) apontaram em seu trabalho que a quantidade de dados disponíveis para a criação de modelos é uma das maiores dificuldades enfrentadas pela Mineração de Dados, ou seja, a adequação, as irregularidades nos dados e a necessidade de capturar a natureza imprecisa do comportamento do ser humano.

Uma nova abordagem de algoritmo híbrido chamado de Algoritmo evolutivo híbrido na seleção de atributos, com o intuito de redução de dimensionalidade. Por meio dos resultados encontrados foi possível concluir que o uso desse algoritmo produziu bons resultados em relação aos classificadores e um alto nível de consistência comparada a outros algoritmos (TAN et al., 2009).

Já no trabalho de Cornelis et al., (2010) foi utilizada a seleção de atributos em conjunto com a lógica *Fuzzy* visando à questão de redução de dimensionalidade. O trabalho teve como objetivo introduzir o conceito *Fuzzy* no processo de redução para aumentar qualidade do subconjunto de dados. Pelos resultados foi possível constatar o potencial do uso do *Fuzzy* na redução de dimensionalidade em relação a atributos, levando a construção de modelos com uma melhor acuraria.

Os trabalhos na área de Seleção de Atributos visam à redução de dimensionalidade, porém não foram encontrados trabalhados utilizando os algoritmos *CFS, CSE* na abordagem Filtro e *Naive Bayes, J48* e *SVM* na abordagem *Wrapper* para o domínio de Clientes.

[3] A entropia de informação é usada para quantificar a importância de um atributo.

Não foram identificados também até a presente data pesquisas em que se apliquem conceitos de *Framework* para redução de dimensionalidade de bases no domínio de Clientes.

3 METODOLOGIA

Este capítulo apresenta a metodologia usada para o desenvolvimento deste trabalho. A seção 3.1 aborda a classificação científica da presente pesquisa. A seção 3.2 descreve o processo de desenvolvimento da pesquisa. A seção 3.3 retrata informações sobre as bases de dados, considerada a primeira etapa do experimento. A seção 3.4 descreve à segunda etapa, que consiste na aplicação do método de Seleção de Atributos e da aplicação dos conceitos de *Framework*. A seção 3.5 reporta informações referentes à classificação, chamada de terceira etapa do trabalho. A última seção aborda o processo de validação dos resultados, última etapa do experimento.

3.1 CLASSIFICAÇÃO DA PESQUISA

Essa seção tem o intuito de classificar a presente pesquisa quanto à metodologia científica usada para o seu desenvolvimento. Levaram-se em consideração à questão da problemática, natureza, objetivos e procedimentos técnicos adotados. Desta forma, segue a classificação segundo Turrioni e Mello (2011):

- Quanto à abordagem do problema: quantitativa, porque realizou a medição das relações entre as variáveis, taxas de acerto, objetivando a obtenção de informações referentes aos melhores e piores algoritmos para seleção de atributos, considerando as bases no domínio de cliente.
- Quanto à sua natureza: aplicada, pois através dos atributos mais relevantes, usou-se o algoritmo de classificação para geração de conhecimento ou padrão no domínio de cliente.
- Quanto a seus objetivos: explicativa, pois aprofundou o conhecimento dos métodos de redução existentes na literatura e os aplicou através de um experimento.
- Quanto aos procedimentos técnicos: experimental, pois se determinou o objeto de estudo, as bases de dados no domínio de cliente, identificando

as taxas de acertos que representa as variáveis de comparação entre os algoritmos para Seleção de Atributos.

A próxima seção reporta o processo de desenvolvimento da presente pesquisa.

3.2 PROCESSO DE DESENVOLVIMENTO DA PESQUISA

A presente pesquisa centraliza sua execução no método de Seleção de Atributos e aplicação dos conceitos de *Framework* com o intuito de avaliar os resultados destes conceitos na redução de dimensionalidade.

Este trabalho foi operacionalizado por meio de cinco etapas: Escolha das Bases de Dados, Preparação das Bases de Dados, Aplicação do Método de Seleção de Atributos e dos Conceitos de *Framework*, Execução dos Algoritmos de Classificação e Avaliação dos Resultados Obtidos, ilustradas na Figura 8.

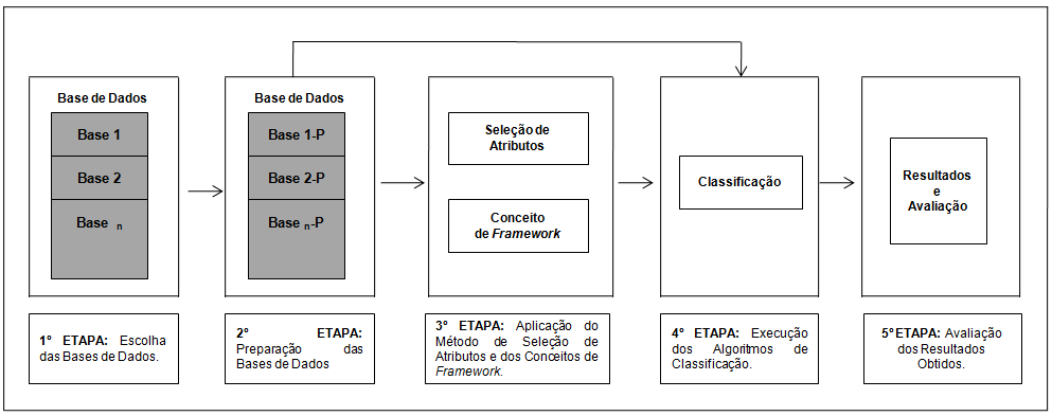


Figura 8 - Processo geral de desenvolvimento do trabalho
Fonte: Autoria própria

A primeira etapa refere-se à escolha da base de dados (Base 1, Base 2, ..., Base $_n$) que serão utilizadas para o experimento, em que o *n* representa a quantidade total de bases. Inicialmente foram selecionadas 3 (três) bases de dados para o experimento, pertencentes aos segmentos: bancário, vendas e seguros. As bases de dados são de domínio público, e estão disponíveis para *download* a partir do *site Machine Learning Repository* (UCI, 2011).

As bases poderiam ser de outros segmentos ou áreas. Nesta pesquisa optou-se pelo domínio Clientes. Lembrando que um segmento é formado por um grupo de compradores identificáveis em um mercado (KOTLER, 2000). Em relação ao domínio, este é definido como um conjunto de características que descrevem uma família de problemas onde uma determinada aplicação pretende propor uma solução (MENDES, 2011).

Na segunda etapa, Preparação das Bases de Dados foi realizada a limpeza dos dados para garantir sua integridade, resultando em: Base 1-P, ..., Base $_n$-P (onde P significa que a base foi preparada). Já na terceira etapa foi realizada a seleção dos atributos, usando o método de Seleção de Atributos e Conceitos de *Framework,* com o intuito de avaliar os resultados do seu uso.

A quarta etapa, Execução dos Algoritmos de Classificação, refere-se a aplicação da tarefa de Classificação tanto nas bases oriundas da segunda etapa quanto nas bases que foram geradas por meio da Seleção de Atributos e *Framework*. Por fim, foi realizada a análise dos resultados dos dois métodos aplicados na pesquisa. Detalha-se a seguir cada uma dessas etapas.

3.3 ESCOLHA DAS BASES DE DADOS

Nessa etapa, realiza-se a seleção da base de dados, ou seja, a amostra. A quantidade da amostra para este trabalho foi de 3 (três) bases, pois para a aplicação do conceito de *Framework* esta quantidade é a mínima. De acordo com Johnson (1993), é suficiente fazer uso de cerca de três aplicações-exemplo no processo de desenvolvimento de um *Framework* de domínio. Uma aplicação-exemplo, neste trabalho, representa um segmento de um dado domínio. Considerando a Mineração de Dados não existe um mínimo estabelecido.

Como o experimento nesta pesquisa trata de cliente, as bases selecionadas foram no domínio Cliente. Para a realização do experimento, selecionaram-se bases de dados de domínio público que contém os dados e os atributos de Clientes. A seguir segue a descrição das 3 (três) bases de dados que foram utilizadas no presente trabalho.

- Base 1: Dados de Crédito de um Banco Alemão

A base contém dados sobre pessoas que os classifica de acordo com o risco de crédito que essas podem apresentar, podendo ser classificadas como bom ou ruim. Para a presente pesquisa essa base será chamada *Stalog*. A base possui ao total 1000 instâncias, com 20 atributos, sendo que 7 são numéricos e 13 categóricos. Esta base está disponível para *download* a partir do *site Machine Learning Repository* (UCI, 2011).

- Base 2: Dados de Vendas

Essa base contém dados sobre vendas de produtos, denominada neste trabalho de *Customer*. Também está disponível no *site Machine Learning Repository* para *download*. A base possui ao total 1966 instâncias com um total de 49 atributos, sendo 46 atributos numéricos e 3 atributos categóricos.

- Base 3: Companhia de Seguros

Inicialmente com a obtenção da base foi disponibilizado um conjunto de dados, com aproximadamente 5822 instâncias, contento perfis completos dos clientes referentes informações de compra ou não da apólice de seguro. O arquivo contém 86 atributos. O nome da base é *The Insurance Company (TIC) Benchmark*, porém será chamada no trabalho como *Insurance* e está disponível para *download* no *site Machine Learning Repository* (UCI, 2011).

A Tabela 1 ilustra a quantidade total de atributos e instâncias de cada base para melhor realização e compreensão dos dados.

Tabela 1 - Quantidade Total de Atributos e Instâncias de cada Base

Bases	Atributos	Instâncias
Stalog	21	1000
Customer	49	1966
Insurance	86	5822

Fonte: Autoria própria

A próxima seção descreve a segunda etapa da pesquisa.

3.4 PREPARAÇÃO DAS BASES DE DADOS

A etapa de Preparação da Base de Dados envolve alguns processos, os quais foram descritos pelo algoritmo ilustrado na Figura 9.

Algoritmo: Preparar Base
Entrada: Base 1, Base 2,..., Base $_n$
Processo:
- Identificar o ambiente para a Mineração de Dados
- Realizar Limpeza dos Dados
- Preparar a base para o ambiente de Mineração de Dados

Saída: Base 1-P, Base 2-P,..., Base $_n$-P

Figura 9 - Algoritmo Seleção de Atributos
Fonte: Autoria própria

O algoritmo Preparar Base possui como entrada as bases que foram selecionadas na primeira etapa, a saber, Base 1, ..., Base $_n$, e depois da execução dos três processos tem-se como saída as bases preparadas, ou seja, contendo os dados consistentes. Descreve-se a seguir estes processos.

3.4.1 Identificar o Ambiente para a Mineração de Dados

Para a aplicação da Mineração de Dados é necessário definir o ambiente para executar suas tarefas.

O Quadro 3 apresenta resumidamente as principais ferramentas disponíveis no mercado utilizadas na Mineração de Dados. Para o presente experimento a ferramenta utilizada para automatização do processo foi o *Waikato Environment for Knowledge Analysis* (WEKA, 2010).

O *WEKA* é um pacote de domínio público que consiste de implementações de algoritmos e de diversas técnicas de Mineração de Dados, escrita na linguagem Java, tendo como característica principal ser portável, isto é, pode ser executável em muitas plataformas (WEKA, 2010).

Ferramentas	**Tarefas Disponíveis**	**Fabricante e *Site* de Acesso**
PolyAnalyst	Classificação, regressão, regras de associação, *clustering*, sumarização e modelagem de dependência.	*MegaputerIntelligence* www.megaputer.com
DataMite	Regras de associação.	Dr. *Phillip Vasey* através da LPA Prolog www. ipa.co.uk/dtm.htm
Microsoft Data Analyser 2002	Classificação e *clustering*.	*Microsoft Corp.* www.microsoft.com

Oracle 9i Data Mining	Classificação e regras de associação.	*Oracle Corp.* www.oracle.com
Darwin	Classificação, regressão e *clustering*.	*Oracle Corp.* www.oracle.com
MineSet	Classificação, regressão e *clustering*.	SliconGraphicsinc. www.sgi.com
WEKA	Classificação, regressão, regras de associação, *clustering*, seleção de atributos.	*UniversityWaikato* www.cs.waikato.ac.nz
Intelligent Miner	Regras de associação, padrões sequências, classificação, *clustering*, sumarização e modelagem de dependência.	*IBM Corp.* www.ibm.com

Quadro 3 - Ferramentas para a Mineração de Dados
Fonte: adaptado de Rezende (2005)

Para determinar o *WEKA* como o ambiente para a Mineração de Dados, pois é uma das ferramentas mais usadas na literatura, além de ter característica tais como: acessibilidade, ser de domínio público e executável em muitos sistemas operacionais.

Com o ambiente definido para a Mineração de Dados, há a necessidade de preparar as bases no formato em que a ferramenta escolhida, neste caso, *WEKA*, executa seus algoritmos. Assim, é necessário realizar a limpeza da base de dados. A seguir a próxima seção apresenta este processo.

3.4.2 Realizar a Limpeza dos Dados

Este processo compreende a fase de pré-processamento do processo *KDD*, que se caracteriza pela limpeza e tratamento dos dados, visando à integridade.

Ao final da primeira etapa do experimento, foi efetuado o *download* das bases selecionadas. As bases se encontravam em formato texto (*.txt*). Os arquivos em formato *.txt* foram abertos em um *software* de Planilha Eletrônica, no caso deste trabalho, o Excel e cada base foi transformada em um arquivo no formato *.xls*, para padronização da limpeza dos dados. Segue as subetapas, correspondem às etapas pertencentes ao pré-processamento do processo *Knowledge Discovery in Databases* (*KDD*), deste processo:

- Limpeza dos Dados: Os valores ausentes nos conjuntos de dados foram padronizados usando a opção “substituir”, no caso do Excel, (Ctrl + L).

Para padronizar foi realizada uma análise visual dos dados e de maneira manual foi efetuada as correções, substituições e eliminações necessárias. Valores inconsistentes que não pertenciam ao domínio dos atributos também foram identificados de maneira visual e eliminados. As subetapas podem ser realizadas de forma automatizada, porém é necessário criar um programa para este fim. Isto não foi feito neste trabalho porque não é o seu foco.

- Codificação dos dados: os dados das bases de dados já estavam codificados em valores numéricos ou categóricos.
- Enriquecimento dos dados: Não foi efetuado o enriquecimento dos dados, pois nenhuma informação aos instâncias já existentes nas bases de dados precisou ser inserida, pois as que existiam eram suficientes para o experimento.
- Todas as informações errôneas foram padronizadas e muitas eliminadas, através do uso da opção “substituir”. Para padronizar foi realizada uma análise visual dos dados e de maneira manual foi efetuada as correções, substituições e eliminações necessárias.

Após as subetapas, foi realizado a formatação das bases de dados em arquivos com a extensão .arff , pois a ferramenta escolhida WEKA trabalha com arquivos nessa extensão. A próxima seção reporta o último processo desta etapa.

3.4.3 Preparar a Base para o Ambiente de Mineração de Dados

Para o ambiente de Mineração de Dados a base de dados deve estar no formato adequado. Os arquivos das bases no ambiente *WEKA* devem conter a seguinte estrutura:

- Relação: É a primeira linha do arquivo, contendo a variável *@relation* seguida de uma palavra-chave que tem como intuito identificar a relação ou tarefa que está sendo executada.
- Atributos: é o conjunto de linhas, onde cada linha é iniciada com *@atributte:* seguindo do nome do atributo e seu tipo.

- Dados: Posteriormente a uma linha contento *@data*, no arquivo do *excel* cada linha corresponde a uma instância, devendo ter valores separados por vírgula correspondentes aos atributos da seção *@atributte*.

Para a criação dos arquivos na extensão *.arff (*formato *WEKA)*, após limpeza, codificação e enriquecimento, os dados contidos nas planilhas foram salvos inicialmente na extensão *Comma-separated values (CSV*).

Ao salvar o arquivo nesse formato, os valores pertencentes ao domínio dos atributos são separados por ponto e vírgula (;), porém o *WEKA* exige que sejam separados somente pela vírgula (,). Assim, esses arquivos em formato *CSV* foram abertos em um aplicativo capaz de ler texto, por exemplo, bloco de notas, e realizou-se a substituição do ";" por ",".

Após esse procedimento os arquivos foram salvos na extensão *.arff*. Como resultados foram obtidos as bases preparadas denominadas de: Base 1-P, Base 2-P,..., Base $_n$-P.

A seguir a próxima seção reporta a terceira etapa do experimento.

3.5 APLICAÇÃO DA SELEÇÃO DE ATRIBUTOS E CONCEITOS DE FRAMEWORK

Essa seção tem o intuito de descrever a terceira etapa da presente pesquisa na qual foram aplicados os conceitos de Seleção de Atributos e *Framework*. A subseção 3.4.1 descreve como o método de Seleção de Atributos foi usado, bem como todas as etapas do processo *KDD* que foram aplicadas. A subseção 3.4.2 relata como foi efetuada a aplicação dos conceitos de *Framework*.

3.5.1 Aplicação de Seleção de Atributos

As bases preparadas, a saber, neste experimento, *Stalog* (Base 1-P)*, Customer* (Base 2-P) e *Insurance* (Base 3-P), são submetidas ao método de Seleção de Atributos. Esse método será detalhado pelo algoritmo chamado Seleção de Atributos, ilustrado no Quadro 4.

Algoritmo: Seleção de Atributos
Entrada: Base 1-P, Base 2-P, ..., Base $_n$-P.
Processo:
- Aplicar as Abordagens Filtro e *Wrapper*

Saída: Atributos Selecionados
Subconjunto 1.1 Base 1-P, ... , Base $_n$-P
Subconjunto 1.2 Base 1-P, ... , Base $_n$-P
.
.
.
Subconjunto 1.$_a$ Base 1-P, ... , Base $_n$-P

Quadro 4 - Algoritmo Seleção de Atributos
Fonte: Autoria própria

A entrada deste algoritmo consiste nas bases de dados que foram preparadas para o ambiente *WEKA (*Base 1-P, ..., Base $_n$-P*)*. Cada uma das bases, foi aberta no ambiente *WEKA* e submetidas ao processo do algoritmo Seleção de Atributos. Os processos desse algoritmo compreendem a Aplicar as abordagens Filtro e *Wrapper* descritas a seguir.

3.5.1.1 **Aplicar abordagem filtro e *wrapper***

Na abordagem Filtro cada uma das bases foi submetida aos algoritmos *Correlation Feature Selection* (*CFS*) e *Consistency Subset Eval (CSE)*. Já na abordagem *Wrapper* foram executados os algoritmos: *Naive Bayes, J48* e *SVM (Support Vector Machines)* em cada base. A escolha desses algoritmos foi baseada em trabalhos existentes na literatura, sendo os mais comumente utilizados.

Após a escolha dos algoritmos, define-se qual método de busca e critério de avaliação que devem usados para a execução dos algoritmos de seleção. Ao utilizar esses algoritmos geram-se vários subconjuntos de atributos denominados de: Subconjunto 1.1 Base 1-P, ... , Base $_n$-P, ..., Subconjunto 1.$_a$ Base 1-P, ... , Base $_n$-P, onde "*a*" representa a quantidade de algoritmos utilizados. Um subconjunto de atributo representa os atributos mais relevantes de uma determinada base, com suas respectivas instâncias.

O Quadro 5 ilustra o critério de busca e a medida de avaliação utilizada para a geração de cada subconjunto nas bases de dados preparadas.

	Algoritmo	Critério de Busca	Medida de Avaliação	Subconjunto
Filtro	*CFS*	Sequencial	Dependência	Subconjunto 1.1
	CSE	Sequencial	Consistência	Subconjunto 1.2
Wrapper	*Naïve Bayes*	Sequencial	*Wrapper* (*Naive Bayes*)	Subconjunto 1.3
	J48	Sequencial	*Wrapper* (*J48*)	Subconjunto 1.4
	SVM	Sequencial	*Wrapper* (*SVM*)	Subconjunto 1.5

Quadro 5 - Algoritmos de Seleção de Atributos
Fonte: Autoria própria

Os subconjuntos de atributos mais relevantes gerados são caracterizados como a saída do algoritmo de Seleção de Atributos. No presente experimento foi gerado um total de 5 (cinco) subconjuntos para cada base de dados, *Stalog*, *Customer* e *Insurance*.

Posteriormente os subconjuntos foram submetidos à execução dos algoritmos classificadores, porém este procedimento pertence à quarta etapa do experimento que será detalhado na seção 3.6.

3.5.2 Aplicações dos Conceitos de *Framework*

Nessa etapa as bases de dados preparadas (Base 1-P, Base 2-P, ..., Base $_n$-P) devem ser submetidas ao algoritmo denominado Conceitos de *Framework*, para a identificação dos atributos comuns e específicos de cada uma delas. Um atributo comum (*frozen spot*) ocorre quando há existência entre as bases de dois ou mais atributos que possuem nomes idênticos ou sinônimos, porém com o mesmo conteúdo de dados. Um atributo específico (*hot spot*) entre as bases de dados consiste de nomes diferentes e de conteúdos diferentes.

Segue o Quadro 6 resumindo o procedimento do algoritmo Conceitos de *Framework*.

Algoritmo: Conceitos de *Framework*
Entrada: Base 1-P, Base 2-P, ..., Base $_n$-P.
Processo:
- Identificação de Atributos Comuns e Específicos

Saída: Atributos Selecionados
Base 1-PF,..., Base $_n$-PF
Subconjunto F 1.1 Base 1-PF, ... , Base $_n$-PF
Subconjunto F 1.2 Base 1-PF, ... , Base $_n$-PF

Quadro 6 - Algoritmo Conceitos de *Framework*
Fonte: Autoria própria

A entrada do algoritmo consiste na obtenção de cada uma das bases obtidas na segunda etapa. Com relação ao processo desse algoritmo, deve-se identificar os atributos comuns e específicos entre as bases descrito a seguir.

3.5.3 Identificação de Atributos Comuns e Específicos

No processo de identificação usa-se o método de Ben-Abdalhal et al. (2004). Nesse método existe a comparação entre as classes, composta de atributos e métodos de forma manual.

Porém, neste trabalho, o objetivo é comparar somente atributos, pois uma base de dados é composta de atributos e seus respectivos conteúdos. As relações adaptadas para comparar atributos estão descritos no Quadro 7.

Relação 1:

- Att_equiv (Base 1-P,.....Base $_n$-P): significa que as bases (Base 1-P,.....Base $_n$-P) tem nomes de atributos idênticos e com o mesmo tipo de dado. Onde: Base 1-P: primeira base, Base $_n$-P: última base.
  - ✓ Regra 1: O atributo é considerado comum e este é adicionado a base de atributos comuns entre os segmentos.

Relação 2:

- Att_int (Base 1-P,.....Base $_n$-P): significa que as bases (Base 1-P,.....Base $_n$-P) tem nomes de atributos diferentes, porém são sinônimos e com o mesmo tipo de dado.
  - ✓ Regra 2: O atributo é considerado comum e este é adicionado a base de atributos comuns entre os segmentos. Neste caso, deve-se escolher um nome padrão para o atributo.

Relação 3:

- Att_conf (Base 1-P,.....Base $_n$-P): Significa que existe pelo menos um atributo de uma das bases (Base 1-P,.....Base $_n$-P) que tem nome equivalente a algum atributo das outras bases, porém apresentam conteúdo e tipo diferentes.
  - ✓ Regra 3: Neste caso o atributo não pode ser considerado ou adicionado a base de atributos comuns, pois não há similaridade de conteúdo e tipo de dado entre as bases. Portanto, este atributo deve ser adicionado a base de atributos específicos entre os segmentos.

Relação 4:

- Att_dist (Base 1-P,.....Base $_n$-P): Significa que nenhuma relação acima foi considerada.
  - ✓ Regra 4: O atributo é considerado específico e este é adicionado a base de atributos específicos de cada segmento.

Quadro 7 - Relação de Comparação de Atributos
Fonte: Autoria própria

Exemplificando o processo de seleção de atributos, a Figura 10 ilustra como foi feito a seleção de atributos comuns e específicos. Porém, para fins de ilustração as Bases X, Y e Z foram criadas com os atributos que são fictícios. Sendo que a Base X possui xn atributos, onde tx: total de atributos da Base X e 1≤ n ≤ tx, assim sucessivamente.

Figura 10 - Demonstrativo de Seleção de atributos comuns e específicos
Fonte: Autoria própria

É possível notar que os atributos que estão em cor vermelha, são os comuns entre as três bases de dados. O primeiro atributo na Base X é o Nome, logo é analisado as outras bases restantes para verificar se há esse mesmo atributo nelas. Se houver, verifica-se qual relação esta situação pertence. Neste caso, a Relação 1 (Att_equiv) é válida e pela Regra 1 o atributo é classificado como comum.

Considerando os atributos propósito de crédito, quantidade de crédito e histórico de crédito, verifica-se que a Regra 4 será válida e portanto estes atributos são específicos somente para a Base Y, pois não houve ocorrência desses nas outras duas bases. Fato este que indica a especificidade de cada segmento ou domínio da aplicação da base.

Ainda é relevante salientar que os atributos em cor azul nas três bases de dados, Profissão, Emprego e Categoria de trabalho, são atributos escritos de maneira diferente, porém são sinônimos, porque contém o mesmo tipo de informação.

Quando ocorre essa situação, cabe ao responsável da análise, ou seja, analista do processo, verificar o conteúdo desses atributos para caracterizá-los como sinônimos. Se considerado um atributo comum, é porque são sinônimos e

segue a Regra 2 e pode ocorrer a necessidade de padronização dos nomes dos atributos. Caso contrário, não são sinônimos, segue a Regra 3, e acaba sendo classificado como um atributo específico.

Como para cada base se deve separar os atributos comuns e específicos, então a quantidade total de subconjunto neste processo é de: $n * 2$, onde n é o total de bases e o 2 representa os subconjuntos dos comuns (Subconjunto F 1.1) e específicos (Subconjunto F 1.2). Por exemplo, neste trabalho se utiliza três bases (Base 1-P, Base 2-P e Base 3-P) então se criam: $3 * 2 = 6$ subconjuntos totais. Isto está apresentado no Quadro 8.

Subconjunto	Descrição
Subconjunto F 1.1	Base 1-PF, *Stalog*: Atributos Comuns entre todas as bases, porém com os dados pertencentes à base *Stalog*. Base 2-PF *Customer*: Atributos Comuns entre todas as bases, porém com os dados pertencentes à base *Customer*. Base 3-PF *Insurance*: Atributos Comuns entre todas as bases, porém com os dados pertencentes à base *Insurance*.
Subconjunto F 1.2	Base 1-PF *Stalog*: Atributos Específicos da base *Stalog*. Base 2-PF *Customer*: Atributos Específicos da Base *Customer*. Base 3-PF *Insurance*: Atributos Específicos da Base *Insurance*.

Quadro 8 - Descrição dos Subconjuntos Gerados após aplicação conceitos de *Framework*
Fonte: Autoria própria

Após a identificação dos subconjuntos com seus respectivos atributos é necessário mover das bases preparadas às respectivas instâncias desses atributos. Esse processo pode ser realizado de forma automatizada.

Um problema que pode ocorrer durante essa remoção é que o atributo teve que passar pelo processo de categorização. Por exemplo, considerando duas bases, em que uma contém o atributo Produto, porém em outra os produtos foram categorizados em {'TV','Aparelho de som'} conforme ilustra a Figura 11.

Base X-P

Cliente	Produto
1	TV
2	Aparelho

Base Y-P

Cliente	TV	Aparelho
1	Sim	Sim
2	Sim	Não

Figura 11 - Base X-P e Y-P
Fonte: Autoria própria

Neste caso, para a criação da base utilizando os conceitos de *Framework*, além de concluir que estes atributos são Comuns, deve-se transformar os dois atributos da Base Y-P em um único atributo, recebendo as respectivas instâncias da base preparada conforme ilustra a Figura 12 para o Subconjunto *Framework* 1.1 Base Y-PF[4]. Este exemplo está sendo ilustrado para os atributos comuns, segue-se o mesmo processo para os atributos específicos.

Subconjunto Framework 1.1 Base X-PF

Cliente	Produto
1	TV
2	Aparelho

Subconjunto Framework 1.1 Base Y-PF

Cliente	Produto
1	TV
1	Aparelho
2	TV

Figura 12 - Subconjunto Framework 1.1 Base X-PF e Y-PF
Fonte: Autoria própria

Em relação à quantidade de instâncias pertencentes às bases preparadas haverá um acréscimo ao gerar as bases usando o conceito de *Framework*, conforme exemplificado anteriormente.

Geram além dos Subconjuntos F 1.1 e F 1.2, as bases com todos os atributos categorizados, a qual foi denominada de Base 1-PF, ..., Base $_n$-PF.

Os Subconjuntos *Framework* 1.1 e 1.2 e as Base 1-PF, ..., Base $_n$-PF são submetidos à execução dos algoritmos *Naive Bayes*, *J48* e *SVM* para a tarefa de classificação, a qual pertence a quarta etapa do trabalho e será descrita a seguir.

[4] A sigla PF é usada para indicar preparada usando conceito de *Framework*.

3.6 EXECUÇÃO DOS ALGORITMOS DE CLASSIFICAÇÃO

A etapa de classificação do experimento consiste na execução dos algoritmos classificadores, também utilizando a ferramenta *WEKA*. Para essa etapa segue o algoritmo chamado de Classificação, Quadro 9, resumindo o procedimento adotado nessa etapa.

Algoritmo: Classificação
Entrada: Base 1-P, 2-P e Base $_n$-P
Base 1-PF, Base 2-PF, Base $_n$-PF
Subconjunto 1.1, ..., Subconjunto 1.$_a$ Base 1-P, ... , Base $_n$-P
Subconjunto F 1.1 Base 1-PF, ... , Base $_n$-PF
Subconjunto F 1.2 Base 1-PF, ... , Base $_n$-PF
Processo: Aplicação Algoritmos Classificadores (*Naive Bayes*, *J48* e *SVM*)
Saída: Taxa de Acerto dos Classificadores

Quadro 9 - Algoritmo Classificação
Fonte: Autoria própria

A entrada do algoritmo de Classificação consiste da obtenção das bases oriundas da segunda etapa além das bases e subconjuntos originados na terceira etapa. A seguir descreve-se o processo deste algoritmo.

3.6.1 Aplicação Algoritmos Classificadores (*Naive Bayes*, *J48* e *SVM*)

Este processo ocorre para os subconjuntos de atributos gerados tanto no método de Seleção de Atributos e dos conceitos de *Framework*.

Foi utilizado o método *Validação Cruzada* para à execução dos algoritmos classificadores. Esse método consiste em subdividir a base de dados em 3 partes, onde 2 consistem em uma base de treinamento e a outra parte restante é destinada para teste. Ou seja, uma proporção de 2/3 dos dados para treinamento e o 1/3 restante para teste. Para todos os subconjuntos gerados foram criados bases de treinamento e teste.

Para a preparação das bases no método Validação Cruzada foi necessário organizar o número total de instâncias de cada base de maneira aleatória para não ocorrer indução de resultado. Isto pode ser realizado por meio de aplicativos neste trabalho, optou-se pelo *Excel* devido ao conhecimento do software. Para a geração

de números aleatórios foi necessário inserir uma nova função no *Excel*, visto que a função já existente neste aplicativo gera números aleatórios com repetição. Dessa forma, foi possível gerar os números aleatórios do total de instâncias para cada base sem repetição.

A Figura 13 ilustra um exemplo de como pode ser realizado o particionamento das bases por meio da Validação Cruzada, em que o número de bases é três.

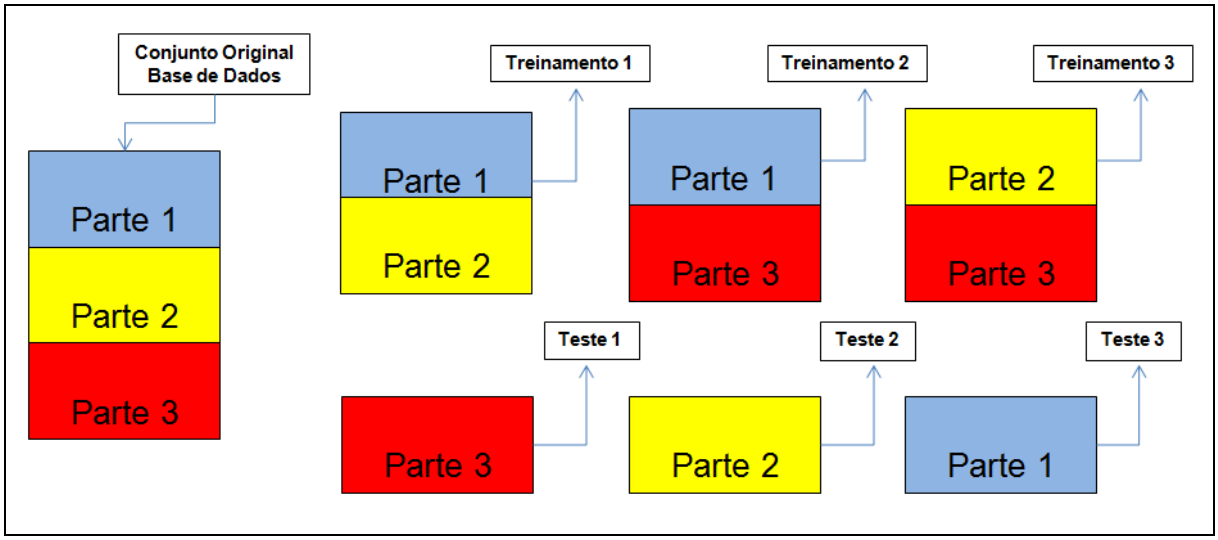


Figura 13 - Particionamento das Bases em treinamento e de testes - Método Validação Cruzada
Fonte: Autoria própria

Após essa divisão, inicialmente dois terços da base farão parte da base de Treinamento 1 (Parte 1 e Parte 2) e um terço da base será a base de Teste (Parte 3). Para as posteriores bases de Treinamento e de Testes o procedimento deve ser repetido, porém se alterna a base de Testes, se no Teste 1 foi a Parte 3 nas demais bases de Teste serão Parte 2 e Parte 1.

O próximo passo é aplicar os algoritmos classificadores: *Naive Bayes, J48* e *SVM* que já estão codificados no ambiente *WEKA*. Findando a execução dos algoritmos, anotam-se as taxas de acerto as quais são entrada para a próxima etapa.

3.7 AVALIAÇÃO DOS RESULTADOS OBTIDOS

A quinta etapa desta pesquisa é destinada a avaliar o método de Seleção de Atributos e a aplicação dos Conceitos de *Framework*. Essa etapa compreende a fase de Pós-Processamento do processo *KDD*.

O algoritmo Avaliação, exibido no Quadro 10, ilustra os passos que foram usados durante esta última etapa do desenvolvimento.

Algoritmo: Avaliação
Entrada: Taxa de Acerto dos Classificadores para cada um dos seguintes elementos:
Base 1-P, 2-P e Base $_n$-P
Base 1-PF, Base 2-PF, Base $_n$-PF
Subconjunto 1.1, ..., Subconjunto 1.$_a$ Base 1-P, ... , Base $_n$-P
Subconjunto F 1.1 Base 1-PF, ... , Base $_n$-PF
Subconjunto F 1.2 Base 1-PF, ... , Base $_n$-PF
Processo: Calcular Média e Desvio-Padrão
Saída: Valores da Média
Melhores e Piores Algoritmos
Relação de Atributos usando os conceitos de Seleção de Atributos e *Framework*

Quadro 10 - Algoritmo Avaliação
Fonte: Autoria própria

Esse algoritmo utiliza como entrada todas as taxas de acerto que foram obtidas por meio da execução dos algoritmos classificadores, considerando todas as bases e subconjuntos que foram gerados a partir da segunda etapa até a quarta.

Neste algoritmo existem dois processos principais os quais serão descritos a seguir.

3.7.1 Calcular Média e Desvio-Padrão

Como foi utilizado a Validação Cruzada de 2/3, para cada base de dados obtêm-se três valores da taxa de acerto, visto que foram geradas sempre três bases de Treinamento e três de Testes.

Neste processo existem várias avaliações que são realizadas, sendo elas com:

- Todos os atributos (Base 1-P, 2-P e Base $_n$-P): calcula-se a média aritmética e o desvio-padrão considerando as taxa de acerto que foram obtidas nas execuções dos Treinamentos e Testes para cada uma das bases em cada um dos classificadores. Logo após, compara-se os resultados para identificar o melhor e o pior algoritmo.
- Atributos Selecionados (Subconjunto 1.1, ..., Subconjunto 1.$_a$ Base 1-P, ... , Base $_n$-P): o cálculo da média e desvio-padrão é igual ao item anterior, porém nesta avaliação identifica-se qual foi o melhor e o pior desempenho para os algoritmos utilizado tanto a abordagem Filtro (*CFS* e *CSE*) quanto na *Wrapper*.
- Todos os atributos categorizados (Base 1-PF, Base 2-PF, Base $_n$-PF): segue-se a mesma ideia descrita anteriormente, porém os resultados são oriundos das bases categorizadas usando o conceito de *Framework*. Determina-se também o melhor e o pior algoritmo classificador.
- Atributos Comuns e Específicos (Subconjunto F 1.1 Base 1-PF, ... , Base $_n$-PF e Subconjunto F 1.2 Base 1-PF, ... , Base $_n$-PF): o procedimento para cálculo da média e desvio-padrão é igual ao das outras avaliações, porém os subconjuntos de entradas são os dos Comuns e os Específicos. Neste caso, determina-se entre os comuns qual o melhor e o pior algoritmo classificador, assim como para os específicos.

Após estas avaliações, realiza-se uma comparação geral entre o melhor e pior desempenho dos algoritmos classificadores para cada uma das bases considerando o método de Seleção de Atributos e os conceitos de *Framework*.

Avalia-se também o total de atributos gerados pela Seleção de Atributos para encontrar qual o melhor e o pior resultado em relação aos atributos selecionados. Além disto, compara-se esse total com os atributos selecionados por meio de *Framework*. Este processo também se deu por meio média aritmética e desvio-padrão.

Com o valor da média e do desvio-padrão é estabelecido um intervalo de valor, o qual consiste na diferença para mais ou para menos em relação à média da quantidade de atributos gerados na Seleção de Atributos.

Com o intervalo de valores estabelecido, os subconjuntos tanto da Seleção de Atributos e conceitos de *Framework* são comparados, visando encontrar uma

quantidade de atributos iguais dentro do intervalo estabelecido para cada base de dados. O procedimento de identificação de atributos iguais é efetuado por meio de uma análise manual e visual. Desta forma, identificam-se as relações dos atributos selecionados usando os conceitos de Seleção de Atributos (baseado no melhor algoritmo de seleção) e *Framework*.

O próximo capítulo descreve os resultados atingidos por meio do presente experimento.

4 ANÁLISE DOS RESULTADOS

Este capítulo reporta os resultados encontrados por meio do experimento realizado por este trabalho com uso do Método de Seleção de Atributos e da aplicação do Conceito de *Framework*. A seção 4.1 aborda os resultados dos classificadores nas bases de dados com todos os atributos. A seção 4.2 descreve os resultados da utilização do Método de Seleção de Atributos, sendo que a seção 4.2.1 relata os resultados na abordagem Filtro e a seção 4.2.2 trás os resultados referentes à abordagem *Wrapper*. Os resultados da aplicação dos conceitos de *Framework* são descritos na seção 4.3. Por fim, a última seção 4.4 apresenta a comparação geral dos conceitos utilizados no experimento.

4.1 RESULTADOS DOS CLASSIFICADORES NAS BASES DE DADOS COM TODOS OS ATRIBUTOS

Conforme mencionado na seção 3.4 as bases preparadas (Base 1-P= *Stalog*, Base 2-p= *Customer*, Base 3 P= *Insurance*) foram inicialmente submetidas ao particionamento dos dados para que ocorresse a divisão de amostra de treinamento e de teste. Dessa forma, três subconjuntos de treinamento e testes, para fins de validação dos resultados através do método Validação Cruzada foram obtidas. Criou-se um total de 18 (dezoito) bases de dados, sendo que 9 (nove) são de treinamento e 9 (nove) para teste.

Tendo como intuito de identificar o subconjunto de atributos ideal com base na taxa de acerto dos classificadores, primeiramente as 3 (três) bases de dados preparadas foram submetidas aos classificadores: *Naive Bayes, J48* e *SVM*, resultando em 3 (três) resultados, os quais foram submetidos ao cálculo da média aritmética e do desvio padrão.

A Tabela 2 apresenta o número de instâncias das bases de treinamento e de teste para a classificação das bases preparadas com todos os atributos. A quantidade de atributos das bases originais já foram ilustradas a Tabela 1 na seção 3.3.

Tabela 2 - Total de Instâncias das Bases de Treinamento e de Teste – Classificação com os Todos os Atributos

	Stalog	***Customer***	***Insurance***
Treinamento 1	667	1311	3882
Teste 1	333	654	1940
Treinamento 2	667	1311	3882
Teste 2	333	654	1940
Treinamento 3	667	1311	3882
Teste 3	333	654	1940

Fonte: Autoria própria

A Tabela 3 ilustra os resultados quando todos os atributos foram considerados. Nota-se que nos resultados dos classificadores os valores representam a média aritmética e o desvio padrão (D.P). A média aritmética[5] foi calculada através das taxas de acertos obtidas pelas amostras de treinamento e teste.

Tabela 3 - Resultados dos Classificadores nas Bases com Todos os Atributos

	Naïve Bayes		***J48***		***SVM***	
	Média	D.P.	Média	D.P.	Média	D.P.
Stalog	57,12%	0,76%	54,32%	2,37%	56,61%	1,14%
Customer	79,06%	2,43%	94,04%	0,31%	94,02%	0,32%
Insurance	75,6%	2,07%	73,90%	3,94%	74,70%	3,19%

Fonte: Autoria própria

De acordo com os resultados anteriores, para a base *Stalog* verifica-se que o algoritmo que apresentou o melhor desempenho foi o *Naive Bayes,* pois obteve o maior valor para a média dos resultados das taxas de acerto nas bases de treinamento e de teste. Porém, ainda sim o valor de 57,12% de média para a taxa de acerto[6] é um valor baixo. Com relação ao pior desempenho para esta base se destaca o algoritmo *J48*, pois apresenta a menor de taxa de acerto para as bases de treinamento e de teste.

Diferentemente dos demais resultados da base *Stalog*, a base *Customer* apresentou os melhores resultados de execução para todos os algoritmos. Para esta base nenhum valor foi inferior a 79% para a média de taxa de acerto, sendo que nas demais bases todos os valores foram inferiores a esse valor. O melhor algoritmo

[5] Média = [(Treinamento 1 e Teste 1) + (Treinamento 2 e Teste 2) + (Treinamento 3 e Teste 3)]/3
[6] Um valor ideal para a taxa de acerto deve compreender o intervalo de 70% a 100% (BORGES, 2006).

para essa base foi o *J48*, obtendo o maior valor de taxa de acerto, 94,04%. Porém, com relação ao pior desempenho destaca-se o *Naive Bayes*.

A base *Insurance* apresentou resultados melhores se comparado com a base *Stalog*, com valores maiores para a média de taxas de acerto. Com relação ao algoritmo de melhor desempenho novamente destacou-se o *Naive Bayes,* com o maior valor de média. Por sua vez, o algoritmo *J48* apresentou o pior resultado.

O Gráfico 1 ilustra um comparativo das médias de taxa de acertos para os classificadores considerando todas as bases preparadas com todos atributos.

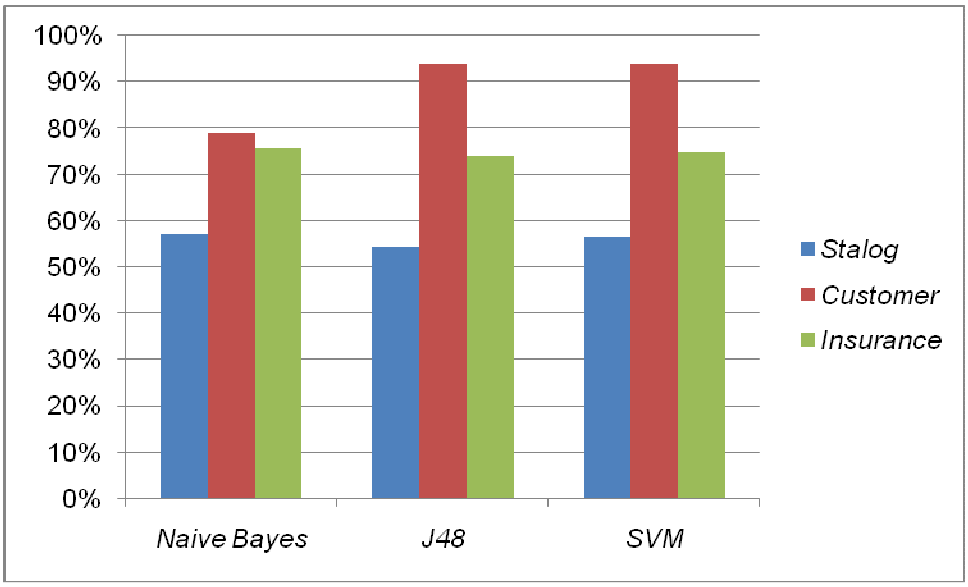


Gráfico 1 - Comparativo dos Resultados dos Classificadores com Todos os Atributos
Fonte: Autoria própria

Neste gráfico se verifica que as melhores execuções dos algoritmos classificadores ocorreram na base *Customer*, pois apresentaram as maiores taxas de acerto. O algoritmo *Naive Bayes* foi o melhor para as bases *Stalog* e *Insurance*, e ainda destaca-se como pior algoritmo o *J48*. O algoritmo que apresentou uma execução intermediária foi o *SVM* para as 3 (três) bases de dados.

A seguir a próxima seção trás os resultados obtidos na aplicação do Método de Seleção de Atributos.

4.2 RESULTADOS DA SELEÇÃO DE ATRIBUTOS SOBRE AS BASES DE DADOS

Para efetuar a seleção de atributos foram utilizadas duas abordagens: Filtro e a *Wrapper*. Utilizou-se como critério de busca o Sequencial para ambas as abordagens. Porém, na abordagem Filtro a medida de avaliação usada para o algoritmo *CFS* foi à de dependência e para o *CSE* à de consistência.

A Tabela 4 mostra o número de atributos gerados em cada subconjunto para as bases *Stalog*, *Customer* e *Insurance*, bem como para cada algoritmo. O Subconjunto 1.1 contém o subconjunto de atributos gerados em cada base para o algoritmo *CFS*, assim como para os outros subconjuntos considerando os algoritmos *CSE*, *J48*, *Naive Bayes*, *SVM*, respectivamente (Subconjunto 1.2,.., Subconjunto 1.5).

Tabela 4 - Números de Atributos selecionados para casa base

		Algoritmos	***Stalog***	***Customer***	***Insurance***
Filtro	Subconjunto 1.1	*CFS*	3	11	10
	Subconjunto 1.2	*CSE*	14	21	29
Wrapper	Subconjunto 1.3	*Naïve Bayes*	6	8	-
	Subconjunto 1.4	*J48*	6	5	-
	Subconjunto1.5	*SVM*	10	8	-

Fonte: Autoria própria

Considerando os dados da tabela, verifica-se que não foram gerados subconjuntos na abordagem *Wrapper* para a base *Insurance*. Isso ocorreu pelo fato da busca ser sequencial para frente. Nesse tipo de busca os subconjuntos de atributos para estas bases não apresentou um atributo que comparado aos pares com todos os atributos garantisse a melhor qualidade do subconjunto selecionado. O Gráfico 2 ilustra um comparativo da quantidade de atributos selecionados de cada algoritmo utilizado tanto na abordagem Filtro como *Wrapper*.

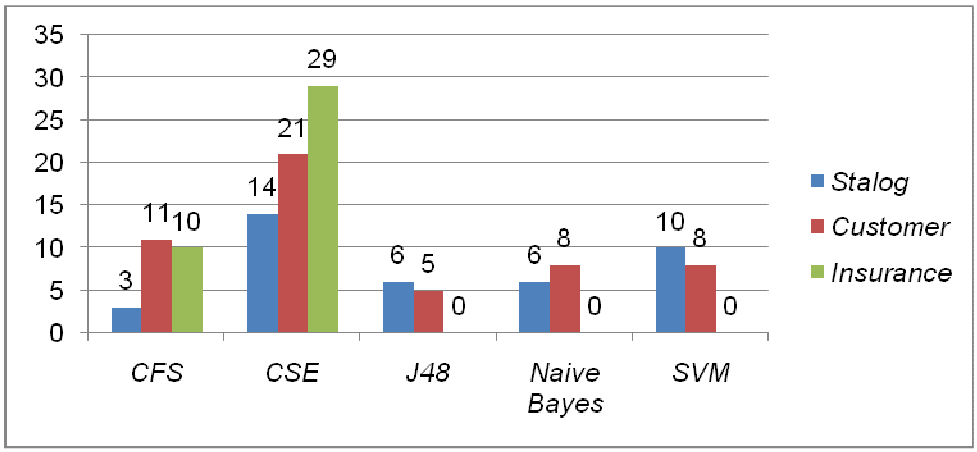


Gráfico 2 - Comparativo do número de atributos selecionados
Fonte: Autoria própria

De acordo com o gráfico, verifica-se que o maior número de atributos selecionados ocorreu para a base *Insurance*, na abordagem Filtro. Isso deve-se ao fato de que à base original possui um maior número de atributos, totalizando 86, em comparação com as demais bases. Ainda observa-se que o algoritmo que selecionou um número maior de atributos foi o *CSE* na abordagem Filtro e o que selecionou um número menor de atributos foi *J48,* na abordagem *Wrapper*.

A seguir a próxima seção trás informações sobre os resultados na abordagem Filtro.

4.2.1 Abordagem Filtro

Conforme já mencionado, as três bases *Stalog, Customer* e *Insurance* foram submetidas à abordagem Filtro, para o algoritmo *CFS* e *CSE*. Com a execução desses algoritmos foram gerados dois subconjuntos de atributos.

Para o algoritmo *CFS* foi gerado o Subconjunto 1.1 e para o algoritmo *CSE* foi gerado o Subconjunto 1.2. O Subconjunto 1.1 compreende o subconjunto de atributos gerados em cada base, *Stalog, Customer* e *Insurance*, assim como os Subconjuntos de atributos 1.2 considerando agora o algoritmo CSE.

Após a geração dos subconjuntos de atributos, a tarefa de classificação com os algoritmos *Naive Bayes, J48* e *SVM* usando o método de Validação Cruzada. Dessa forma, foi gerado um total de 36 bases de dados, sendo que 18 são de treinamento e 18 para teste. O número de instâncias para treinamento e testes já foram apresentadas na Tabela 2.

A Tabela 5 ilustra os resultados da média de taxa de acerto dos classificadores e os desvios-padrão para o subconjunto de atributos gerados com o algoritmo *CFS*.

Tabela 5 - Resultados dos Classificadores para o Algoritmo CFS

Subconjunto 1.1						
	Naïve Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Stalog	73,50%	1,39%	71,60%	1,66%	70,80%	0,32%
Customer	57,53%	1,45%	56,66%	1,08%	58,80%	1,97%
Insurance	92,61%	0,89%	94,02%	0,71%	94,02%	0,71%

Fonte: Autoria própria

Verifica-se que o algoritmo que teve o melhor desempenho em relação aos demais na base *Stalog* foi o *Naive Bayes*, com uma taxa média de acerto de 73,50%. Já o algoritmo *SVM* foi considerado o que teve o melhor desempenho, para a base *Customer*. Por fim, na base *Insurance* os melhores algoritmos foram *SVM* e *Naive Bayes*, visto que apresentaram resultados iguais, 94,02%.

Com relação aos piores desempenhos na base *Stalog*, destaca-se o *SVM com* 70,80% de média de taxa de acerto. Com relação à base *Customer* o pior desempenho do algoritmo classificador foi o *J48*, com uma taxa de acerto de 56,66% e na base *Insurance* o pior foi considerado o *Naive Bayes*, com a média de 92,61%.

Os 3 (três) algoritmos classificadores apresentaram melhor desempenho na base *Insurance*, pois nenhuma taxa de acerto foi inferior a 90%. O Gráfico 3 ilustra um comparativo das médias de taxa de acertos para os classificadores utilizados nos subconjuntos gerados para o algoritmo *CFS*.

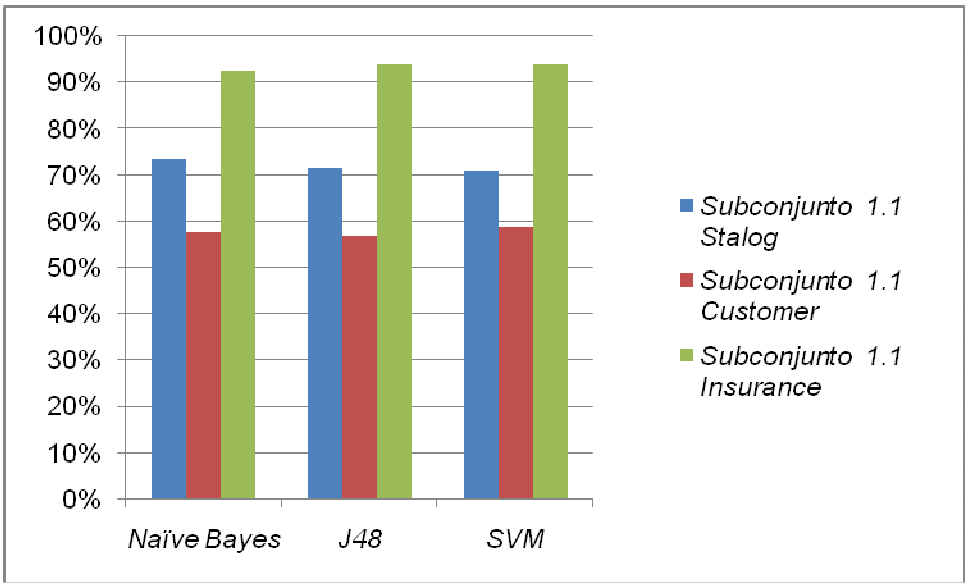


Gráfico 3 - Comparativo dos Resultados Usando os Classificadores para o Subconjunto 1.1
Fonte: Autoria própria

Novamente o melhor desempenho dos algoritmos foi observado na base *Insurance* e o pior foi na base *Customer*. Porém, cada base apresentou resultado mediano para os três algoritmos, não ocorrendo dispersões elevadas para as de taxa de acerto.

A Tabela 6 mostra à média e desvios-padrão dos resultados dos classificadores com relação ao Subconjunto 1.2 gerado com a aplicação do algoritmo *CSE*.

Tabela 6 - Resultados dos Classificadores para o Algoritmo CSE

	Subconjunto 1.2					
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Stalog	74,30%	2,55%	73,27%	2,00%	75,00%	3,74%
Customer	57,53%	1,20%	54,17%	3,33%	57,43%	1,20%
Insurance	94,02%	0,27%	94,06%	0,33%	83,00%	0,68%

Fonte: Autoria própria

Observa-se que para a base *Stalog*, o algoritmo que apresentou o melhor desempenho foi o *J48*. Na base *Customer* destaca-se o *Naive Bayes*, com o valor de média de taxa de acerto 57,52% e na *Insurance* o J48 teve o melhor desempenho com o valor de média de 94,02%.

Os piores desempenhos dos algoritmos na base *Stalog* foi verificado com o algoritmo *J48*, com o valor de média 73,26%. Esse algoritmo também teve o menor

desempenho para a base *Customer*, com 54,17%. Por fim, na base *Insurance*, o pior desempenho foi observado para o algoritmo *SVM*, apresentando o valor de média de 82,99%.

O Gráfico 4 mostra um comparativo das médias de taxa de acertos para os classificadores utilizados no Subconjunto 1.2 gerado para o algoritmo *CSE*.

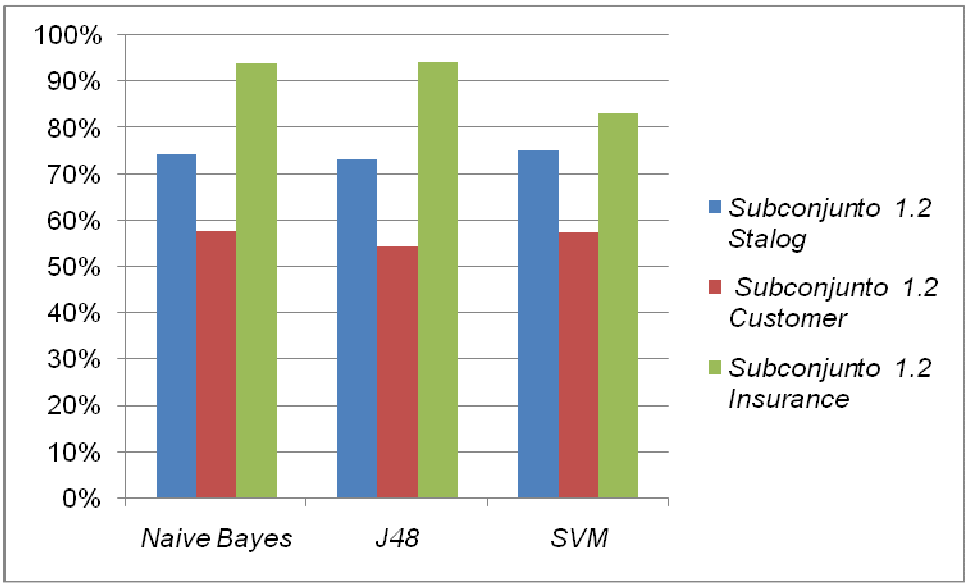


Gráfico 4 - Comparativo dos Resultados Usando os Classificadores para o Subconjunto 1.2
Fonte: Autoria própria

Novamente é possível notar que o melhor desempenho dos algoritmos classificadores foi observado na base de dados *Insurance* e o pior na base *Customer*.

O Gráfico 5 apresenta a média de todas as execuções dos algoritmos classificadores usando o resultado obtido em cada algoritmo de seleção da abordagem Filtro.

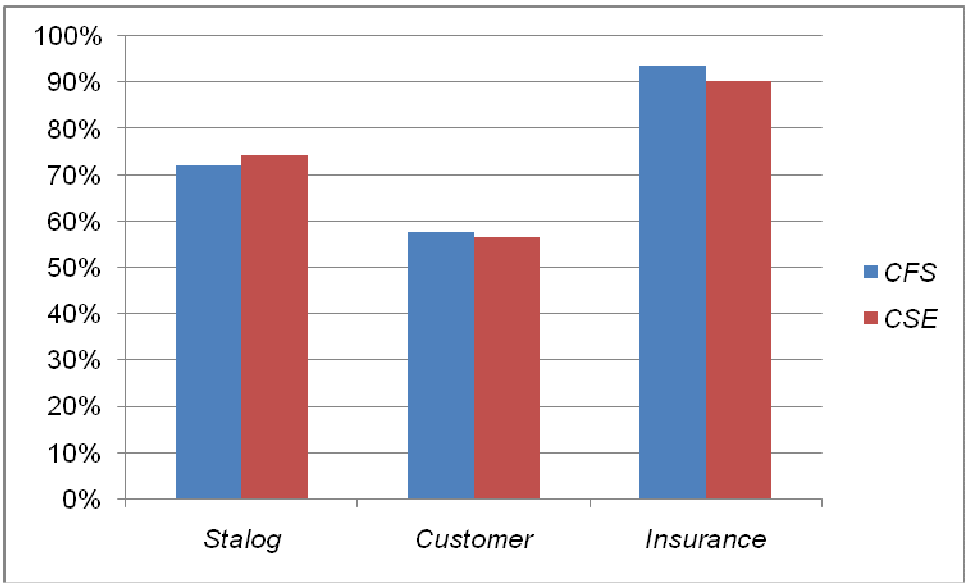


Gráfico 5 - Comparativo das Médias dos Classificadores Usando o Resultado Obtido pela Abordagem Filtro
Fonte: Autoria própria

Através das taxas de acerto, obtidas pela abordagem filtro, é possível afirmar que o melhor desempenho foi observado para o algoritmo *CFS* nas bases *Customer* e *Insurance* e já o algoritmo *CSE* apresentou um melhor desempenho somente na base *Stalog*.

A próxima seção trás os resultados obtidos para a abordagem *Wrapper*.

4.2.2 Abordagem *Wrapper*

Essa abordagem compreende a aplicação dos algoritmos *Naïve Bayes, J48* e *SVM* para a geração dos subconjuntos de atributos. Nessa etapa foram gerados 3 (três) subconjuntos por meio do método de seleção de atributos. O Subconjunto 1.3 através da aplicação do algoritmo *Naive Bayes*, Subconjunto 1.4 por meio da execução do algoritmo *SVM* e o Subconjunto 1.5 através do algoritmo *J48*. O Subconjunto 1.3 compreende o subconjunto de atributos gerados em cada base, *Stalog, Customer* e *Insurance*, assim como para os Subconjuntos de atributos 1.4 e 1.5.

Após geração dos subconjuntos, esses foram submetidos à tarefa de classificação, na qual também foi efetuado o particionamento das bases em treinamento e de testes para a aplicação através do método de Validação Cruzada. Desta forma, foi gerado um total de 36 bases de dados, sendo que 18 são de

treinamento e 18 para teste. O número de instâncias para treinamento e testes já foram apresentadas na Tabela 2.

Após particionamento das bases, o Subconjunto 1.3 gerado foi submetido à tarefa de classificação, usando apenas o algoritmo *Naive Bayes*. Já o subconjunto 1.4 foi submetido ao algoritmo *J48* e no último Subconjunto 1.5 foi executado o algoritmo *SVM*. A Tabela 7 apresenta os resultados obtidos para os classificadores *Naive Bayes, J48* e *SVM.*

Tabela 7 - Resultados dos Classificadores para a Abordagem *Wrapper*

	Subconjunto 1.3		**Subconjunto 1.4**		**Subconjunto 1.5**	
	Naive Bayes		J48		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Stalog	73,70%	3,81%	73,20%	0,33%	74,40%	1,90%
Customer	58,95%	0,82%	59,31%	1,60%	60,58%	0,20%

Fonte: Autoria própria

Para a base *Stalog*, na abordagem *Wrapper*, o melhor desempenho do algoritmo classificador foi o *SVM*, com um valor médio de taxa de acerto de 74,40%. Na base *Customer* o melhor desempenho também foi o *SVM*, com um valor de média de 60,48%.

Considerando a base *Stalog* o pior método de seleção de atributos foi o uso do algoritmo *J48*, pois apresentou o menor valor para a taxa de acerto dos algoritmos classificadores. Já na base *Customer* o pior desempenho do método de seleção de atributos foi verificado no algoritmo *Naive Bayes*, com um valor da média da taxa de 58,95%.

O Gráfico 6 trás um comparativo das médias de taxas de acertos dos algoritmos classificadores para os Subconjuntos 1.3, 1.4 e 1.5.

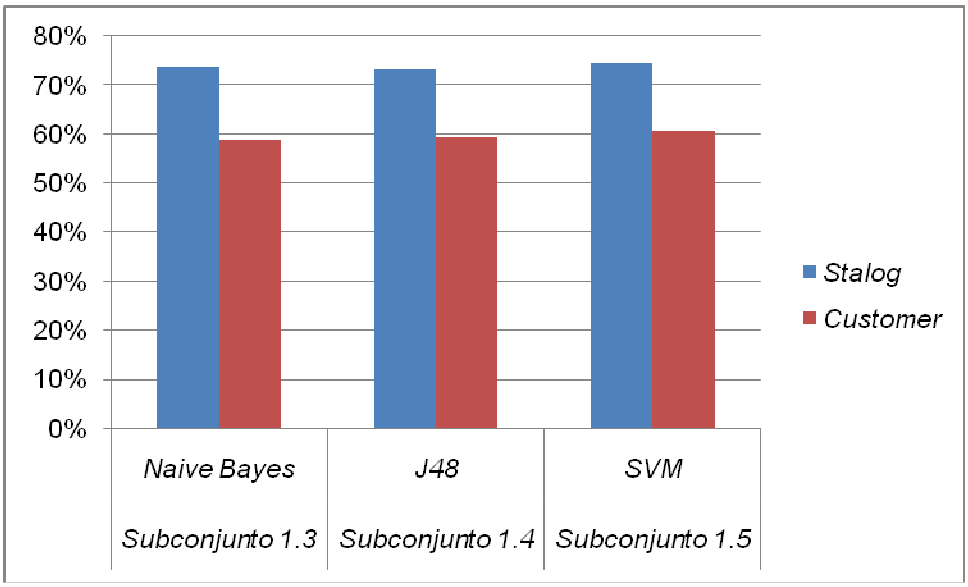


Gráfico 6 - Comparativos com Resultados Usando os Classificadores para os Subconjuntos 1.3, 1.4 e 1.5
Fonte: Autoria própria

É possível verificar que o algoritmo *SVM* apresentou o melhor desempenho entre os demais, mostrando que é o melhor método de Seleção de Atributos na abordagem *Wrapper*. Ainda com relação ao método de seleção, os piores dentre esta abordagem foram o *J48* e o *Naive Bayes*.

A próxima seção relata os resultados obtidos com a aplicação dos conceitos de *Framework*.

4.3 RESULTADOS DA APLICAÇÃO CONCEITOS DE FRAMEWORK

Nessa etapa as três bases de dados (*Stalog, Customer* e *Insurance*) foram submetidas aos conceitos de *Framework*, de acordo com o método adaptado de Ben-Abdalhal et al. (2004) para a identificação dos atributos comuns e os específicos.

Foram gerados assim dois subconjuntos de atributos para as três bases de dados: *Stalog, Customer* e *Insurance*. O Subconjunto F. 1.1 compreende os atributos comuns entre as três bases e o Subconjunto F. 1.2 dos atributos específicos de cada segmento. O Quadro 6 ilustra os subconjuntos gerados para realização do experimento usando os conceitos de *Framework*.

Abaixo a próxima seção aborda o processo efetuado para a identificação dos atributos comuns que fazem parte do Subconjunto F 1.1.

4.3.1 Geração do Subconjunto F 1.1 – Atributos Comuns

O Subconjunto F 1.1 compreende os atributos comuns entre as 3 (três bases[7]). Para obter os atributos comuns entre as bases foi efetuada uma análise manual utilizando as relações descritas no Quadro 7 e que foram adaptadas do trabalho de Ben-Abdalhal et al. (2004). Os atributos identificados como comuns entre as bases analisadas totalizaram 9 (nove), os quais estão apresentados no Quadro 11, bem como as respectivas regras aplicadas.

	Atributos Comuns Selecionados	**Regra Aplicada**
1	Cliente	Regra 2
2	Produto	Regra 2
3	Quantidade	Regra 2
4	Casa	Regra 2
5	Renda	Regra 2
6	Estado Civil	Regra 2
7	Nível de Educação	Regra 2
8	Idade	Regra 2
9	Categoria de Trabalho	Regra 2

Quadro 11 - Subconjunto F 1.1 - Atributos Comuns Selecionados
Fonte: Autoria própria

Para a identificação dos atributos comuns foi necessário efetuar uma análise dos valores atribuídos para cada um dos atributos nas suas bases preparadas. Não houve aplicação da Regra 1 porque em nenhuma das três bases esses 9 (nove) atributos tinham nomes idênticos. A Regra 2 foi aplicada em todos os casos, pois os atributos possuíam nomes diferentes, mas eram sinônimos e com o mesmo tipo de dado.

Ainda usando a Regra 2, foi necessário escolher um nome padrão para os atributos considerados comuns. No presente experimento, o Quadro 11 ilustra os nomes adotados como padrões. A escolha dos nomes fica a critério da pessoa que está realizando a análise. As Regras 3 e 4 não foram aplicadas porque tratam especificamente dos atributos específicos.

[7] Uma base de dados no contexto da definição de *Framework* é denominada de aplicação-exemplo.

Com os atributos comuns definidos, o próximo passo foi preparar as bases de dados contendo somente esses atributos. Estas bases receberam os seguintes nomes: Subconjunto F 1.1 *Stalog*, Subconjunto F 1.1 *Customer* e o Subconjunto F 1.1 *Insurance*. A quantidade de instâncias geradas para cada subconjunto está ilustrada na Tabela 8.

Tabela 8 - Quantidade Total de Atributos e Instâncias de cada Subconjunto

	Comuns	
Bases	Atributos	Instâncias
Subconjunto F 1.1 *Stalog*	9	1000
Subconjunto F 1.1 *Customer*	9	41286
Subconjunto F 1.1 *Insurance*	9	12226

Fonte: Autoria própria

Em relação à quantidade de instâncias pertencentes às bases preparadas *Stalog*, *Customer* e *Insurance* (ver Tabela 1) houve um acréscimo devido ao processo de preparação das bases, as quais deveriam conter somente os atributos selecionados. Por exemplo, na base *Customer* existiam os seguintes atributos:

{'domestic','apparel','leisure','kitchen','luxury','promo7','promo13','mensware','flatwar
e','dishes','homeacc','lamps','linens','blankets','towels','outdoor','coats','wcoat','wappar','hhapp
ar','jewelry'}

bem como suas respectivas instâncias. Neste caso, ao realizar o processo de análise de atributos verificou-se que os mesmos pertencem a uma categoria, por exemplo, Produto. Com isto a nova base (subconjunto) ao invés de conter todos esses atributos, passou a possuir um atributo com o nome Produto e suas instâncias foram obtidas transformando cada valor do atributo original em quantidade. Esse processo foi efetuado através de análise visual e também do uso de uma planilha eletrônica.

Com os Subconjuntos 1.1 *Stalog*, 1.1 *Customer* e 1.1 *Insurance* executou-se os algoritmos de classificação: *Naive Bayes*, *J48* e *SVM.* Assim como na Seleção de Atributos, também foi aplicado o método Validação Cruzada para particionamento das bases de treinamento e de teste. Dessa forma, foi gerado um total de 18 bases de dados, sendo que 9 (nove) são de treinamento e 9 (nove) para teste. A Tabela 9

apresenta o número de instâncias dos Subconjuntos F 1.1 e 1.2 de treinamento e de testes geradas após a aplicação dos conceitos de *Framework.*

Tabela 9 - Total de Instâncias para as Bases de Treinamento e de Teste para os Subconjuntos F 1.1 e 1.2

	Stalog	***Customer***	***Insurance***
Treinamento 1	666	27524	81508
Teste 1	334	13763	40754
Treinamento 2	666	27524	81508
Teste 2	334	13763	40754
Treinamento3	666	27524	81508
Teste 3	334	13763	40754

Fonte: Autoria própria

A Tabela 10 mostra os resultados das médias e desvios-padrão obtidos pelos classificadores *Naive Bayes, J48* e *SVM*.

Tabela 10 - Resultados dos Classificadores para o Subconjunto F 1.1 - Atributos Comuns

Subconjunto F 1.1 – Atributos Comuns						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Subconjunto F 1.1 *Stalog*	68,30%	2,24%	68,50%	0,55%	69,10%	2,07%
Subconjunto F 1.1 *Customer*	52,35%	0,76%	98,00%	0,96%	52,42%	0,93%
Subconjunto. F 1.1 *Insurance*	90,89%	0,12%	99,88%	0,13%	94,05%	0,08%

Fonte: Autoria própria

De acordo com os dados desta tabela, verifica-se que no Subconjunto F 1.1 *Stalog* o algoritmo de classificação que teve o melhor desempenho foi o *SVM*, com o valor médio de taxa de acerto para o classificador de 69,10%. Já para os outros dois Subconjuntos F 1.1 *Customer* e F 1.1 *Insurance* o melhor desempenho foi para o algoritmo *J48*, onde ambos apresentaram um valor superior de 98% de taxa de acerto. O pior desempenho ficou com o algoritmo *Naive Bayes* para os três subconjuntos.

O Gráfico 7 ilustra o comparativo dos resultados das médias de taxas de acerto para os algoritmos *Naive Bayes, J48* e *SVM*.

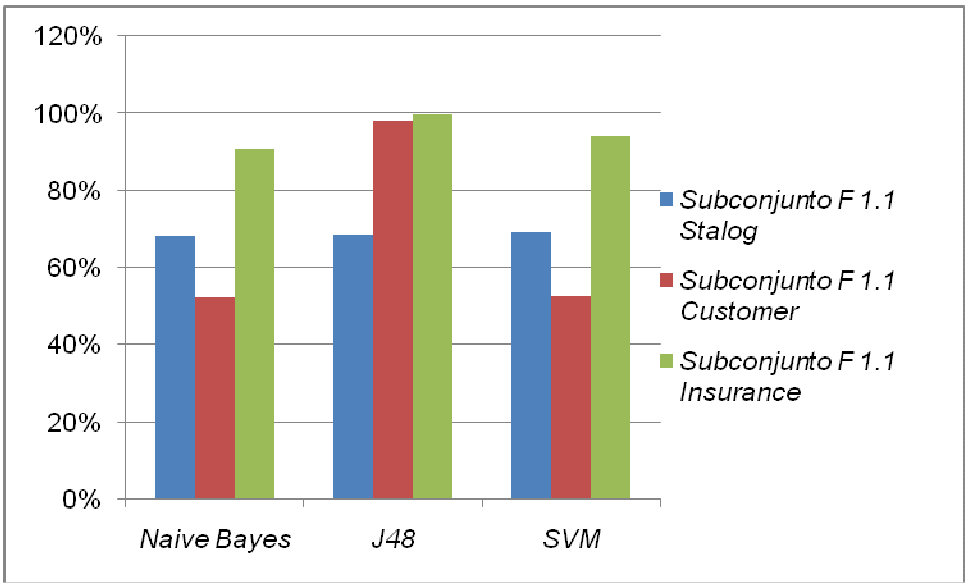


Gráfico 7 - Comparativos Resultados Classificadores para o Subconjunto F 1.1 – Atributos Comuns
Fonte: Autoria própria

De acordo com o gráfico anterior, as melhores taxas de acertos foram observadas o Subconjunto F 1.1 *Insurance.* Porém, constata-se uma grande diferença na taxa de acerto para o algoritmo *J48* no Subconjunto F 1.1 *Customer*, com 98,00%, visto que os demais resultados para os outros algoritmos apresentaram um valor inferior a 60%. Para os outros subconjuntos os valores foram medianos entre os três algoritmos para as médias de taxas de acertos.

A seguir a próxima seção contém os principais resultados para os atributos específicos.

4.3.2 Geração do Subconjunto F 1.2 – Atributos Específicos

A identificação dos atributos específicos também foi submetida à análise das relações estabelecidas no método adaptado de Ben-Abdalhal et al. (2004). As relações que consideram um atributo como específico são as Relações 3 e 4.

A Relação 3 significa que há pelo menos um atributo de umas das bases que possui nome equivalente a algum atributo das outras bases, porém apresenta conteúdo e tipo diferente. Ou seja, se o número total de bases for 3 (três) e se houve em 2 (duas) bases, 2 (dois) atributos que possuem nomes equivalentes e apresentam conteúdo equivalente, esses não são considerados comuns, pois os atributos nas 3 (três) bases não apresentam conteúdo e tipo similares. Para ser

considerado um atributo comum é necessário sua existência nas três bases, conforme explicado na seção 4.3.1.

A Relação 4 ocorre quando nenhuma das Relações (1, 2 e 3) são identificadas, isto é, os atributos são considerados específicos para cada base. De todos os atributos comparados das 3 (três) bases consideradas no experimento somente para 2 (dois) atributos foram identificadas a Regra 3. Já para o restante dos atributos a relação identificada foi a Regra 4. Devido o fato da quantidade de atributos selecionados ser elevada, segue no Apêndice A o nome dos atributos selecionados, bem como as regras aplicadas para a geração do Subconjunto F 1.2.

A Tabela 11 mostra o número de atributos selecionados no Subconjunto F 1.2.

Tabela 11 - Quantidade de Atributos Específicos e Regras Aplicadas para o Subconjunto F 1.2

Bases de dados	Quantidade de Atributos Selecionados	Regras Aplicadas
Subconjunto F 1.2 *Stalog*	15	Regra 3 e 4
Subconjunto F 1.2 *Customer*	20	Regra 3 e 4
Subconjunto F 1.2 *Insurance*	48	Regra 3 e 4

Fonte: Autoria própria

Com o subconjunto de atributos específicos de cada base identificados, esses foram submetidos aos algoritmos de classificação: *Naive Bayes, J48* e *SVM*. Os resultados desta aplicação estão apresentados na Tabela 12.

Tabela 12 - Resultados dos Classificadores para o Subconjunto F 1.2 - Atributos Específicos

	Subconjunto F 1.2 – Atributos Específicos					
	Naive Bayes		J48		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Subconjunto F 1.2 *Stalog*	74,70%	1,71%	70,20%	0,67%	74,90%	0,43%
Subconjunto F 1.2 *Customer*	60,50%	0,11%	100,00%	0,00%	60,99%	0,27%
Subconjunto F 1.2 *Insurance*	85,29%	0,45%	98,86%	0,02%	94,08%	0,11%

Fonte: Autoria própria

Na classificação desse subconjunto foi observado o valor de 100% de taxa de acerto do algoritmo classificador *J48* para a base *Customer.* Esse mesmo algoritmo também apresentou o melhor desempenho para a base *Insurance.* O algoritmo *Naive Bayes* foi o que resultou em um pior desempenho entre os demais algoritmos para os três subconjuntos. Estes resultados estão ilustrados no Gráfico 8.

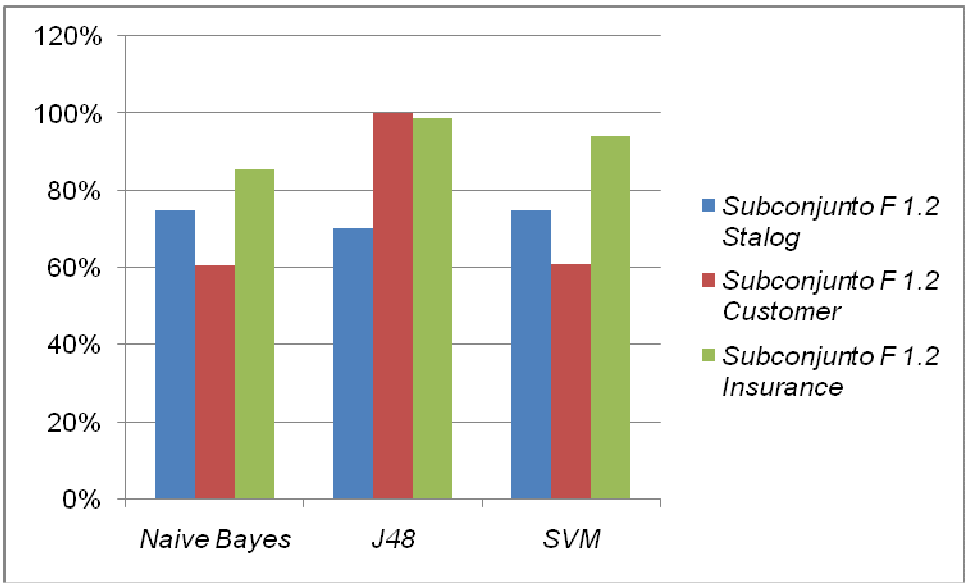


Gráfico 8 - Comparativos Resultados Classificadores para o Subconjunto F 1.2 – Atributos Específicos
Fonte: Autoria própria

A seguir a próxima seção aborda um comparativo dos dois Métodos de Redução, Seleção de Atributos e Conceitos de *Framework*.

4.4 COMPARAÇÃO GERAL

Essa seção tem o intuito de comparar os métodos realizados na presente pesquisa para verificar qual é o melhor método de redução de dimensionalidade entre as bases de dados. A seção 4.4.1 contém um comparativo entre as duas abordagens usadas no método de Seleção de Atributos, Filtro e *Wrapper*. A seção 4.4.2 aborda uma comparação da aplicação de *Framework* usando o método adaptado de Ben-Abdalhal et al. (2004) para identificação de atributos comuns e os específicos entre os segmentos. A seção 4.4.3 apresenta uma análise comparativa entre os conceitos usados para redução de dimensionalidade.

4.4.1 Método de Seleção de Atributos

Posteriormente aos resultados da aplicação do algoritmo de seleção de atributos, usando a abordagem Filtro e *Wrapper*, esses foram submetidos a uma

análise com o intuito de verificar qual foi à melhor abordagem para a geração de subconjuntos.

A Tabela 13 apresenta os valores de média dos classificadores obtidos para o Subconjunto 1.1 *Stalog* com todos os atributos e os subconjuntos submetidos à abordagem Filtro (*CFS* e *CSE*) e *Wrapper*. Lembrando na abordagem *Wrapper* os valores ilustrados correspondem os resultados da classificação de cada algoritmo. Para a análise da Abordagem Filtro é necessário avaliar os resultados obtidos nos três algoritmos classificadores utilizando a média aritmética para apontar qual dos dois algoritmos, *CFS* e *CSE,* foi o melhor. Já, na Abordagem *Wrapper* é feita a análise isolada dos resultados, sem aplicar a média aritmética obtidos de cada classificador para verificar qual dos algoritmos *Naive Bayes*, *J48* e *SVM* foi o melhor na seleção de atributos.

Tabela 13 - Resultados dos Classificadores para o Subconjunto 1.1 *Stalog* Usando o Método de Seleção de Atributos

Stalog						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos	57,12%	0,76%	54,32%	2,37%	56,61%	1,14%
CFS	73,50%	1,39%	71,60%	1,66%	70,80%	0,32%
CSE	74,30%	2,54%	73,27%	2,00%	75,00%	3,74%
Wrapper	73,70%	3,81%	73,20%	0,33%	74,40%	1,90%

Fonte: Autoria própria

Observa-se que para este subconjunto os melhores resultados foram com a utilização do método de Seleção na Abordagem *Wrapper*, juntamente com o algoritmo *SVM*. Os algoritmos *CFS* e *CSE* na média dos classificadores não foram superiores a *Wrapper.* Os piores resultados apresentados foram observados sem o uso do Método de Seleção.

A Tabela 14 mostra os valores de média dos classificadores para o Subconjunto 1.1 *Customer* com todos os atributos submetidos à abordagem Filtro (*CFS* e *CSE*) e *Wrapper*.

Tabela 14 - Resultados dos Classificadores para o Subconjunto *Customer* Usando o Método de Seleção de Atributos

Customer						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos	79,06%	2,43%	94,04%	0,31%	94,02%	0,32%
CFS	57,53%	1,45%	56,66%	1,08%	58,80%	1,97%
CSE	57,53%	1,20%	54,17%	3,33%	57,43%	1,20%
Wrapper	58,95%	0,82%	59,31%	1,60%	60,58%	0,20%

Fonte: Autoria própria

Ao contrário da *Stalog,* os melhores resultados para a *Customer* foram obtidos quando todos os atributos estavam presentes na base. Isto pode ter ocorrido porque os algoritmos de seleção usados não foram os ideais para esse tipo de base.

Os resultados referentes ao Subconjunto 1.1 *Insurance* estão ilustrados na Tabela 15. Observa-se que os valores referentes à abordagem *Wrapper* não estão sendo exibidos porque não houve a geração do subconjunto de atributos, fato esse já explicado na seção 4.2.2.

Tabela 15 - Resultados dos Classificadores para o Subconjunto *Insurance* usando o Método de Seleção de Atributos

Insurance						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos	75,60%	2,07%	73,90%	3,94%	74,70%	3,19%
CFS	92,61%	0,89%	94,02%	0,71%	94,02%	0,71%
CSE	94,02%	0,27%	94,06%	0,33%	83,00%	0,68%

Fonte: Autoria própria

Os resultados apresentados na tabela anterior mostram que novamente os melhores resultados foram conseguidos por meio dos métodos de Seleção de Atributos na Abordagem Filtro, usando o algoritmo *CFS*.

Considerando as três bases analisadas, em duas delas a Seleção de Atributos foi a que apresentou os melhores resultados de média de taxa de acerto.

4.4.2 Aplicações de Conceitos de *Framework*

Com os subconjuntos gerados após aplicação dos Conceitos de *Framework*, esses foram submetidos aos algoritmos classificadores. Assim, a Tabela 16 apresenta a média aritmética dos classificadores para o Subconjunto F 1.1 *Stalog*.

Tabela 16 - Resultados dos Classificadores para o Subconjunto 1.1 *Stalog* Usando o Conceito de *Framework*

Stalog						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos da Base 1-P	57,12%	0,76%	54,32%	2,37%	56,61%	1,14%
Todos os Atributos com a Categorização Base 1-PF	74,10%	2,67%	70,70%	2,81%	76,28%	2,75%
Comuns	68,30%	2,24%	68,50%	0,55%	69,10%	2,07%
Específicos	74,70%	1,71%	70,20%	0,67%	74,90%	0,43%

Fonte: Autoria própria

Conforme os resultados ilustrados na tabela anterior, a utilização dos subconjuntos com todos os atributos categorizados (Base 1-PF) obteve os melhores resultados, e os piores ocorreram com a base contendo somente os atributos Comuns.

A Tabela 17 apresenta os resultados para o Subconjunto F 1.1 *Customer*, no qual a utilização dos atributos Específicos obteve os melhores resultados. O pior resultado foi com o uso dos atributos Comuns. Comparando os resultados entre a base com todos os atributos (Base 2-P) e os que foram categorizados (Base 2-PF) os melhores resultados foram sem a aplicação dos conceitos de *Framework*.

Tabela 17 - Resultados dos Classificadores para o Subconjunto F 1.1 *Customer* Usando o Conceito de *Framework*

Customer						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos da Base 2-P	79,06%	2,43%	94,04%	0,31%	94,02%	0,32%
Todos os Atributos com a Categorização Base 2-PF	60,25%	0,64%	100,00%	0,00%	61,07%	0,55%
Comuns	52,35%	0,76%	98,00%	0,96%	52,42%	0,93%
Específicos	60,50%	0,11%	100,00%	0,00%	60,99%	0,27%

Fonte: Autoria própria

Os resultados para o Subconjunto F 1.1 *Insurance* estão ilustrados na Tabela 18. O uso do subconjunto com os atributos comuns obteve os melhores resultados, sendo os piores valores quando os todos os atributos categorizados foram considerados. Em relação à aplicação do Conceito de *Framework* verificou-se que os resultados foram os esperados, no qual o subconjunto contendo todos atributos categorizados (Base 3-PF) obteve o melhor resultado em relação ao subconjunto com todos os atributos (Base 3-P).

Tabela 18 - Resultados dos Classificadores para o Subconjunto F 1.1 *Insurance* Usando o Conceito de *Framework*

Insurance						
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos da Base 3-P	75,60%	2,07%	73,90%	3,94%	74,70%	3,19%
Todos os Atributos com a Categorização Base 3-PF	82,19%	0,33%	99,99%	0,01%	94,02%	0,12%
Comuns	90,89%	0,12%	99,88%	0,13%	94,05%	0,08%
Específicos	85,29%	0,45%	¨98,86%	0,02%	94,08%	0,11%

Fonte: Autoria própria

A aplicação de *Framework* além de ter apresentado os melhores resultados para as bases *Stalog* e *Insurance,* traz como benefício a separação dos atributos em dois subconjuntos: comuns e os específicos, como explicado na seção 2.2. Ambos os subconjuntos podem ser representados na notação *UML* por meio do diagrama de classe.

A Figura 14 ilustra o diagrama de classe para o subconjunto contendo os atributos comuns, denominado de Subconjunto F 1.1 – Atributos Comuns, os quais foram descritos no Quadro 11. Usa-se a mesma representação gráfica da Figura 14 para os atributos classificados como específicos.

Este diagrama corresponde a análise efetuada no domínio de Cliente considerando: *Stalog, Customer* e *Insurance.* Caso novas bases (*Base$_{a1,...,}$ Base$_{at}$*), onde *Base$_{a1}$ =* é a primeira base a ser inserida e *Base$_{at}$ =* última base a ser acrescentada, neste domínio necessitem ser analisadas, o processo ocorrerá somente entre a Base do Subconjunto F 1.1 – Atributos Comuns e as *Base$_{a1,...,}$ Base$_{at}$*, ou seja, não há necessidade de uma nova análise com as outras bases.

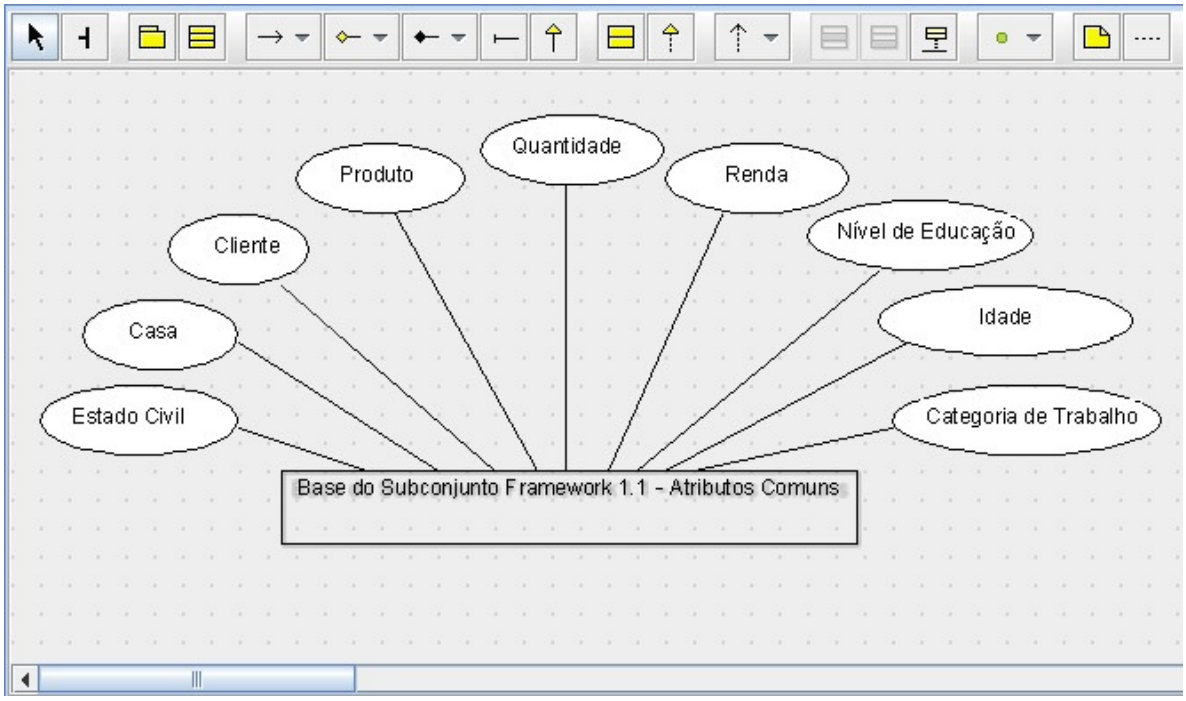


Figura 14 - Diagrama de Classe da Base dos Comuns
Fonte: Autoria própria

Desta forma, reusa-se o que já foi feito, pois o refinamento realizado em uma etapa anterior é usado como a entrada para uma etapa posterior. Esses sucessivos refinamentos podem levar a um modelo mais consistente no domínio da aplicação. Isto também pode ser realizado para os subconjuntos dos específicos.

4.4.3 Método de Seleção de Atributos e Conceitos de *Framework*

Com os atributos comuns e específicos definidos, foi efetuada uma análise comparativa destes com os melhores subconjuntos gerados no Método de Seleção de Atributos.

Para essa comparação foi necessário identificar o número total de atributos selecionados no Método de Seleção de Atributos de cada subconjunto. Com esse levantamento foi efetuado o cálculo da média da quantidade de atributos selecionados para cada subconjunto. Com os valores das médias obtidas, foi calculado o desvio-padrão para estabelecer um intervalo.

O cálculo do intervalo foi feito por meio da subtração do desvio-padrão sobre a média para obter o menor valor do intervalo. Já para estabelecer o maior valor do intervalo foi efetuada a adição do desvio-padrão sobre a média. Obtendo dessa forma dois valores, mínimo e máximo do intervalo estabelecido.

A Tabela 19 apresenta a quantidade de atributos selecionados no Método de Seleção para cada subconjunto, a média de atributos selecionados por subconjunto, desvio-padrão e o intervalo. No Método de Seleção de Atributo para a *Stalog* e *Customer* o melhor subconjunto gerado foi com a aplicação do algoritmo *SVM* na abordagem *Wrapper* e para a *Insurance* o melhor foi com o algoritmo *CFS* na abordagem Filtro.

Tabela 19 - Quantidade de atributos Selecionados, Média, Desvio-Padrão e o Intervalo estabelecido.

	CFS	***CSE***	***Naive Bayes***	***J48***	***SVM***	**Média**	**D. Padrão**	**Intervalo**
Stalog	3	14	6	6	10	7,8	4,2661	(3,5330 – 12,0661)
Insurance	10	29	-	-	-	19,5	13,4350	(6,065 – 32,935)
Customer	11	21	5	8	8	10,6	6,1886	(4,4114 – 16,7886)

Fonte: Autoria própria

A análise de quantidade de atributos também foi feita com o intuito de identificar quais atributos estavam nos subconjuntos gerados no método de Seleção de Atributos quanto na Aplicação de Conceitos de *Framework*, dentro do intervalo estabelecido. Por exemplo, para a *Stalog* foram selecionados os seguintes atributos:

{'Status of existing checking account'; 'Duration in month' ; 'Credit history' ; 'Purpose' ; 'Credit'; 'Amount', 'Savings Accounts/bonds' , 'Personal status and sex' , 'Age in years' , 'Number of existing credits at this bank', 'Number of people being liable to provide maintenance for' }

e se compará-los com os identificados como atributos comuns, obteve-se um total de 4 (quatro) atributos que foram iguais: *'Age in years'*; *Purpose* '; ' *Credit Amount*' e '*Personal status and sex*', os quais receberam o nome, respectivamente, de: Idade, Produto, Quantidade, Estado Civil, no Subconjunto F 1.1 – Atributos Comuns. Assim, foram obtidos os resultados apresentados na Tabela 20.

Tabela 20 - Números de Atributos identificados nos Subconjuntos gerados no Método de Seleção e Aplicação Conceitos de *Framework*.

	Intervalo	**Comuns**	**Específicos**
Stalog	(3,5330 – 12,0661)	4	6
Insurance	(6,065 – 32,935)	4	6
Customer	(4,4114 – 16,7886)	2	6

Fonte: Autoria própria

Conclui-se que os atributos específicos para todos os subconjuntos estão dentro do intervalo estabelecido. Verificou-se a existência de 6 (seus) atributos específicos tanto no melhor subconjunto gerado no Método de Seleção de Atributos, como na aplicação de *Framework*.

Já com relação aos atributos comuns, somente a base *Stalog*, apresentou essa igualdade de atributos, pois apresentou 4 (quatro) atributos selecionados no Método de Seleção de Atributos iguais ao método de aplicação dos conceitos de *Framework.*

As bases *Customer* e *Insurance* não apresentaram esta igualdade de atributos selecionados, pois não tinham o número mínimo de atributos entre os métodos estabelecidos no intervalo.

Um das diferenças entre os conceitos utilizados para redução de dimensionalidade usados neste trabalho, Método Seleção de Atributos e *Frameworks*, está em que no Método de Seleção a análise é realizada em uma única base de dados utilizando um algoritmo e no *Framework* o processo de redução ocorre por meio da investigação de todas as bases do segmento do domínio.

Outra diferença é que no método de Seleção de Atributos não há necessidade realizar o processo de categorização de atributos, o qual por sua vez deve ser feito ao aplicar os conceitos de *Framework*.

Na aplicação de *Frameworks* ao gerar os subconjuntos, os mesmos podem ser utilizados como entrada quando novas bases forem analisadas. Isto permite reduzir o tempo de análise.

Portanto, com os experimentos realizados na área de Perfil de Cliente foi possível verificar que na maioria das bases a utilização dos dois métodos contribui para na redução da dimensionalidade das bases de dados, nos quais os atributos mais relevantes neste domínio foram identificados. Desta forma, diminuiu os espaços de busca de atributo e da remoção de dados contendo ruídos, entre outros.

A quantidade de atributos prejudica o desempenho do algoritmo de aprendizagem quanto na velocidade (devido à dimensionalidade) como na taxa de retorno (devido informações redundantes). Isso pode ser observado nos resultados, pois a redução de atributos apresentou melhores valores de taxa de acerto se comparado com os do conjunto de base preparadas inicialmente no experimento.

Os atributos principais na área de Perfil de Cliente foram identificados e estão listados no Quadro 12 e 13. O Quadro 12 apresenta os atributos para a base *Stalog*, *Customer* e *Insurance* usando Seleção de Atributos.

Atributos Selecionados no Método de Seleção de Atributos		
Stalog	*Customer*	*Insurance*
Status of existing checking account	*Frequent*	*Customer Subtype see L0*
Duration in month	*Recency*	*Social class A*
Credit history	*Telind*	*Average income*
Purpose	*Promo13*	*Contribution car policies*
Credit Amount	*Homeacc*	*Contribution fire policies*
Savings Accounts/bonds	*Numkids*	*Contribution boat policies*
Personal status and sex	*Travtime*	*Contribution social security insurance policies*
Age in years	*Dining*	*Number of private third party insurance*
Number of existing credits at this bank	*-*	*Number of car policies*
Number of people .. maintenance for	*-*	*Number of boat policies*

Quadro 12 - Atributos Selecionados no Método de Seleção de Atributos
Fonte: Autoria própria

O Quadro 13 apresenta os atributos específicos para as bases *Stalog*, *Customer* e *Insurance* usando os conceitos de *Framework*. Os atributos comuns já foram relacionados na Figura 14.

Atributos Específicos Selecionados Usando Conceito de *Framework*		
Stalog	*Customer*	*Insurance*
Status of existing checking account	*Homeval*	*Customer Subtype see L0*
Duration in month real	*Frequent*	*Number of houses 1 - 10r*
Credit history	*Recency*	*Avg size household 1 - 6*
Credit Amount real	*Ntitle*	*Customer main type see L2*
Savings Accounts/bonds	*Telind*	*Roman catholic see L3*
Present Employment Since	*Aprtmnt*	*Protestant ...*
sex	*Mobile*	*Other religion*
Others debtors/guarantors	*County*	*No religion*
Present residence since real	*Return*	*Household without children*
Property	*Custdate*	*Household with children*
Other Installment Plans	*Tmktord*	*High status*
Number of existing credits at this bank real	*Statecod*	*Social class A*
Number of people being liable to provide maintenance for real	*Race*	*Social class B1*
Telephone	*Heat*	*Social class B2*
Foreign worker	*Numcars*	*Social class C*
-	*Numkids*	*Social class D*
-	*Travtime*	*1 car*

-	*Valratio*	*2 cars*
-	*Dining*	*No car*
-	*Sex*	*National Health Service*
-	-	*Private health insurance*
-	-	*Income < 30000*
-	-	*Income 30-45.000*
-	-	*Income 45-75.000*
-	-	*Income 75-122.000*
-	-	*Income >123.000*
-	-	*Purchasing power class*
-	-	*Contribution private third party insurance see*
-	-	*Contribution third party insurance (firms) ...*
-	-	*Contribution third party insurance (agriculture)*
-	-	*Contribution car policies*
-	-	*Contribution delivery van policies*
-	-	*Contribution motorcycle/scooter policies*
-	-	*Contribution lorry policies*
-	-	*Contribution trailer policies*
-	-	*Contribution tractor policies*
-	-	*Contribution agricultural machines policies*
-	-	*Contribution moped policies*
-	-	*Contribution life insurances*
-	-	*Contribution private accident insurance policies*
-	-	*Contribution family accidents insurance policies*
-	-	*Contribution disability insurance policies*
-	-	*Contribution fire policies*
-	-	*Contribution surfboard policies*
-	-	*Contribution boat policies*
-	-	*Contribution bicycle policies*
-	-	*Contribution property insurance policies*
-	-	*Contribution social security insurance policies*

Quadro 13 - Atributos Específicos Selecionados Utilizando o Conceito de *Framework*
Fonte: Autoria própria

Com relação à análise usando a classificação nos dois métodos de redução usados na presente pesquisa, foi efetuado um comparativo das médias obtidas dos classificadores tanto no Método de Seleção quanto no Método de Aplicação de Conceitos de *Framework*. A Tabela 21 ilustra os resultados para a base *Stalog*.

Tabela 21 - Média dos Algoritmos de Classificação do Método de Seleção de Atributos e Aplicações de Conceitos de *Framework* para a base *Stalog*

	Stalog					
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos da Base 1-P	57,12%	0,76%	54,32%	2,37%	56,61%	1,14%
Todos os Atributos com a Categorização Base 1-PF	74,10%	2,67%	70,70%	2,81%	76,28%	2,75%
CFS	73,50%	1,39%	71,60%	1,66%	70,80%	0,32%
CSE	74,30%	2,55%	73,27%	2,00%	75,00%	3,74%
Wrapper	73,70%	3,81%	73,20%	0,33%	74,40%	1,90%
Comuns	68,30%	2,24%	68,50%	0,55%	69,10%	2,07%
Específicos	74,70%	1,71%	70,20%	0,67%	74,90%	0,43%

Fonte: Autoria própria

A partir dos resultados exibidos na tabela anterior, foi possível verificar que o processo de categorização da base após aplicação dos conceitos de *Framework* apresentou melhores resultados do que as bases preparadas com todos os atributos.

Com relação à identificação do melhor subconjunto gerado entre os métodos de redução para esta base, foi verificado que a Abordagem *Wrapper*, usando o algoritmo de seleção *SVM* apresentou o maior valor de taxa de acerto em comparação com os demais. Assim, conclui-se que nesta base o melhor subconjunto de atributos gerados foi com o uso deste algoritmo.

A Tabela 22 apresenta a média dos classificadores para a base *Customer* usando os dois métodos de redução.

Tabela 22 - Média dos Algoritmos de Classificação do Método de Seleção de Atributos e Aplicações de Conceitos de *Framework* para a base *Customer*

	Customer					
	Naive Bayes		*J48*		*SVM*	
	Média	D.P.	Média	D.P.	Média	D.P.
Todos Atributos da Base 2-P	79,06%	2,43%	94,04%	0,31%	94,02%	0,32%
Todos os Atributos com a Categorização Base 2-PF	60,25%	0,64%	100,00%	0,00%	61,07%	0,55%
CFS	57,53%	1,45%	56,66%	1,08%	58,80%	1,97%
CSE	57,53%	1,20%	54,17%	3,33%	57,43%	1,20%
Wrapper	58,95%	0,82%	59,31%	1,60%	60,58%	0,20%
Comuns	52,35%	0,76%	98,00%	0,96%	52,42%	0,93%
Específicos	60,50%	0,11%	100,00%	0,00%	60,99%	0,27%

Fonte: Autoria própria

Para esta base os valores de taxa de acerto dos métodos de redução não foram superiores aos valores da base preparada com todos os atributos.

O melhor subconjunto de atributos gerados para esta base foi com o uso do método de aplicação de conceitos de *Framework, para os atributos específicos.* Ainda com relação este método foi observado que apresentou todos os valores superiores se comparado com o método de Seleção de Atributos. Assim, conclui-se que para esta base o método de Aplicação de Conceitos de *Framework* resultou um melhor desempenho do que o Método de Seleção de Atributos.

A Tabela 23 contém os valores de média dos algoritmos de classificação para a base *Insurance.*

Tabela 23 - Média dos Algoritmos de Classificação do Método de Seleção de Atributos e Aplicações de Conceitos de *Framework* para a base *Insurance*

	Insurance					
	Naive Bayes		*J48*		*SVM*	
Todos Atributos da Base 3-P	75,60%	2,07%	73,90%	3,94%	74,70%	3,19%
Todos os Atributos com a Categorização Base 3-PF	82,19%	0,33%	99,99%	0,01%	94,02%	0,12%
CFS	92,61%	0,89%	94,02%	0,71%	94,02%	0,71%
CSE	94,02%	0,27%	94,06%	0,33%	83,00%	0,68%
Comuns	90,89	0,12%	99,88%	0,13%	94,05%	0,08%
Específicos	85,29	0,45%	98,86%	0,02%	94,08%	0,11%

Fonte: Autoria própria

É possível identificar que o processo de categorização após aplicação dos Conceitos de *Framework* apresentou um desempenho melhor que a base preparada, demonstrando novamente que este processo contribui na redução de dimensionalidade.

O melhor método de redução nesta base foi com a Aplicação dos Conceitos de *Framework*, pois apresentou valores superiores que os obtidos no Método de Seleção de Atributos. Nesta base, o subconjunto de atributos selecionados que apresentou o melhor desempenho foi o de atributos comuns com o uso dos Conceitos de *Framework.*

Notou-se que o uso de *Framework* contribui para a redução de dimensionalidade, porém é necessário realizar novos experimentos para comprovar a sua aplicabilidade juntamente com os conceitos de Mineração de Dados.

4.5 APLICABILIDADE

Esta seção tem o intuito de apresentar a aplicabilidade dos resultados obtidos com os melhores subconjuntos identificados entre os dois métodos de Redução de Dimensionalidade.

Neste trabalho para a base *Stalog* o subconjunto que apresentou os melhores resultados na Seleção de Atributos foi com a aplicação do algoritmo SVM. Já com relação à base *Customer*, o melhor subconjunto gerado foi com a aplicação dos Conceitos de Framework, sendo o subconjunto de atributos específicos. Na base Insurance o subconjunto que apresentou os melhores resultados também foi com o uso dos Conceitos de Framework, mas, por meio do Subconjunto de Atributos Comuns.

Para demonstrar a aplicabilidade dos resultados obtidos neste trabalho, foi aplicado o algoritmo denominado Aplicabilidade, ilustrado na Figura 15, nos três melhores subconjuntos identificados de cada base.

Algoritmo: Aplicabilidade
Entrada: Identificar o Ambiente para a Aplicabilidade
Melhores Subconjuntos
Subconjunto 1.5 Base 1-P
Subconjunto F 1.1 Base 2-PF
Subconjunto F 1.2 Base 3-PF
Processo: Aplicar Tarefa de Classificação - Algoritmo J48
Saída: Regras Subconjunto 1.5 Base 1-P
Regras Subconjunto F 1.1 Base 2-PF
Regras Subconjunto F 1.2 Base 3-PF

Figura 15 - Algoritmo Aplicabilidade
Fonte: Autoria própria

O algoritmo Aplicabilidade possui como entrada os melhores subconjuntos identificados na análise dos resultados na presente pesquisa. Já o processo deste algoritmo compreende à execução da tarefa de Classificação usando o algoritmo J48. Como saída obtêm-se as regras, que fornecem informações de comportamento dos seus clientes.

4.5.1 Identificar o Ambiente para a Aplicabilidade

Para o processo demonstrativo da aplicabilidade dos resultados observados neste experimento, há a necessidade de definição do ambiente para que esta ocorra.

Assim, com o intuito de demonstrar a aplicabilidade dos resultados encontrados, os melhores subconjuntos encontrados em cada base foram submetidos à tarefa de Classificação. Esta tarefa com o uso do Algoritmo J48 é comumente utilizada para a extração do conhecimento de características e comportamento do cliente por meio da obtenção de regras. As regras são o resultado da Tarefa de Classificação.

Para a execução dessa tarefa há a necessidade do estabelecimento de um ambiente apropriado, assim o *WEKA* novamente foi escolhido. Para esta escolha também foi levado em consideração à existência de trabalhos que demonstram a aplicabilidade de métodos de redução.

4.5.2 Melhores Subconjuntos

De acordo com os resultados observados foi possível identificar quais foram os melhores subconjuntos de atributos gerados em cada base. Para a base *Stalog* o Subconjunto 1.5 Base 1-P foi o que apresentou os melhores resultados em relação à taxa de acerto. Já a base Customer o melhor foi o Subconjunto F 1.1 Base 2-PF e por fim, na base Insurance o subconjunto que apresentou os melhores resultados foi Subconjunto F 1.2 Base 3-PF.

Com os melhores subconjuntos identificados estes foram submetidos à Tarefa de Classificação. Porém, conforme abordado anteriormente para a execução desta tarefa as bases dos melhores subconjuntos devem estar devidamente preparadas para a execução do algoritmo *J48*. As bases já se encontram preparadas, pois estas já foram submetidas ao processo de preparação conforme demonstra a seção 3.3.3 da presente pesquisa.

4.5.3 Aplicar Tarefa de Classificação – Algoritmo J48

Cada uma das bases dos melhores subconjuntos foi submetida ao algoritmo *J48*. Lembrando que a escolha desse algoritmo foi baseada em trabalhos existentes na literatura (OMID, 2010).

Os Subconjunto 1.5 Base 1-P, Subconjunto F 1.1 Base 2-PF, Subconjunto F 1.2 Base 3-PF foram executados no ambiente *Weka*, com a aplicação do algoritmo J48 para cada um desses. Para cada subconjunto foi gerado um conjunto de regras.

Esse conjunto de regras está representado no formato de Árvores de Decisão, a qual é capaz de realizar a classificação dos dados.

4.5.4 Regras

Como resultado da aplicação do algoritmo Aplicabilidade foi criado um conjunto de regras, as quais são denominadas de Regras Subconjunto 1.5 Base 1-P, Regras Subconjunto F 1.1 Base 2-PF e Regras Subconjunto F 1.2 Base 3-PF. Essa criação ocorre após seleção do algoritmo J48, onde este constrói um modelo para classificação das instâncias.

O Quadro 14 exemplifica uma das regras geradas para o atributo denominado *Status of existing checking accou*nt, o qual indica a situação da conta corrente existente na base *Stalog*.

```
Status of existing checking account = A11
|  Credit history = A30: 2 (13.0/3.0)
|  Credit history = A31: 2 (22.0/6.0)
|  Credit history = A32
|  |  Number of existing credits at this bank <= 1
|  |  |  Purpose = A40: 2 (41.0/15.0)
|  |  |  Purpose = A41: 1 (13.0/4.0)
|  |  |  Purpose = A42
|  |  |  |  Duration in month <= 16: 1 (16.0/2.0)
|  |  |  |  Duration in month > 16
|  |  |  |  |  Credit Amount <= 3518: 2 (17.0/5.0)
|  |  |  |  |  Credit Amount > 3518: 1 (11.0/3.0)
|  |  |  Purpose = A43
|  |  |  |  Duration in month <= 33: 1 (29.0/11.0)
|  |  |  |  Duration in month > 33: 2 (5.0)
|  |  |  Purpose = A44: 2 (5.0/1.0)
|  |  |  Purpose = A45: 2 (1.0)
|  |  |  Purpose = A46: 2 (7.0/2.0)
|  |  |  Purpose = A47: 1 (0.0)
|  |  |  Purpose = A48: 2 (1.0)
|  |  |  Purpose = A49
|  |  |  |  Duration in month <= 36: 1 (2.0)
|  |  |  |  Duration in month > 36: 2 (2.0)
|  |  |  Purpose = A410: 1 (2.0)
|  |  Number of existing credits at this bank > 1: 2 (8.0/2.0)
|  Credit history = A33
|  |  Duration in month <= 18: 1 (3.0)
|  |  Duration in month > 18: 2 (9.0)
|  Credit history = A34: 1 (67.0/18.0)
```

Quadro 14 - Regras Geradas com a aplicação do algoritmo J48
Fonte: Autoria própria

Onde:

Atributo 1: (qualitativo)

Situação da conta corrente existente

A11 : ... < 0 unidade monetária

A12 : 0 <= ... < 200 unidade monetária

A13 : ... >= 200 unidade monetária

A14 : não possui conta corrente

Atributo 2: (numérico)

Duração em meses do crédito tomado

Atributo 3: (qualitativo)

Credit history

A30 : nenhum crédito tomado

A31 : todos créditos pagos devidamente

A32 : créditos existentes pagos até agora

A33: atraso no pagamento no passado
A34: relato crítico de outros créditos existentes em outros bancos

Atributo 4: (qualitativo)
Purpose
A40 : carro (novo)
A41 : carro (usado)
A42 : movies/equipamentos
A43 : radio/televisão
A44 : aparelhos domésitocs
A45 : reparos/ consertos
A46 : educação
A47 : viagens/ férias
A48 : recilcagem
A49 : negócios
A410 : outros

Atributo 16: (numérico)
Número de créditos existentes neste banco

A partir das regras geradas é possível fazer conclusões à respeito de um cliente no processo de tomada de decisão. Por exemplo, de acordo com a regra gerada no Quadro 14, o valor A11 atribuído ao atributo 1 indica os clientes que não possuem movimentação financeira neste banco atualmente. Estes clientes têm uma operação vigente de crédito, pois por meio do atributo 16 é possível identificar que estes possuem somente uma operação. Dos clientes que já realizaram empréstimos neste banco, a regra gerada utiliza os valores A30, A31 e A32 os quais indicam que estes cumpriam com suas obrigações dentro do prazo, ou seja, pagavam devidamente suas contas.

Destes clientes o propósito do crédito foi direcionado para a aquisição de carros novos e usados e para compras de mobílias. O prazo destas operações foi menor ou igual a 16 meses. Com relação aos valores emprestados foi possível concluir que correspondem a aproximadamente 3500 da unidade monetária respectiva do país corrente da Base *Stalog*. Desta maneira, por meio da regras geradas é se estabelece as relações entre os atributos para chegar à uma determinada conclusão.

O atributo classe da Base *Stalog* tem o intuito de classificar um cliente como sendo bom ou ruim. Assim, através da aplicação do algoritmo J48 podem-se adquirir regras determinantes e relações para a tomada de decisão em relação aos clientes. Além de, obter conhecimento sobre seu comportamento.

A Figura 16 ilustra graficamente no formato de uma árvore a regra gerada a partir da aplicação de um algoritmo de classificação, a qual foi ilustrada no ambiente *WEKA* no Quadro 14.

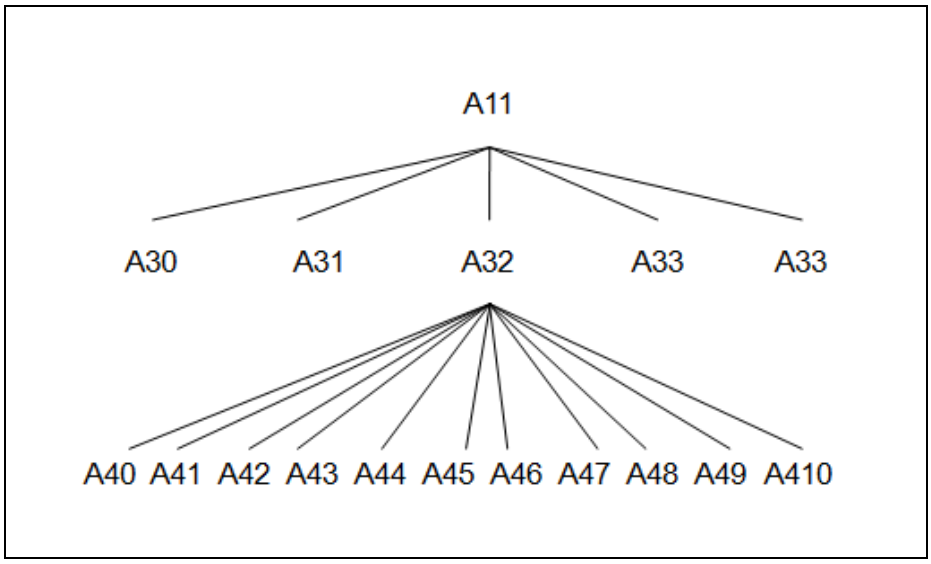


Figura 16 - Ilustração Gráfica de uma Árvore de Decisão
Fonte: Autoria própria

Neste caso as relações são estabelecidas entre os atributos e valores atribuídos à esses. Com a aplicação de um determinado algoritmo de classificação, após o procedimento de redução de dimensionalidade o gestor passa a ter o padrão e o conhecimento de seus clientes por meio da análise de seus dados. A partir disto, definem-se estratégias de atuações de mercado para manter-se competitivo. Por meio dos algoritmos de redução a quantidade de regras geradas é menor por conter somente os atributos mais relevantes o que facilita a análise dos dados.

A Gestão deste conhecimento assume um papel importante para que ocorra sua eficaz aplicação focando relacionamentos duradouros. Atualmente um fator diferencial das organizações está em oferecer valores aos seus clientes, porém esta oferta depende e demanda à busca por *insights*, para reconhecer e interpretar as tendências emergentes de mercado.

5 CONCLUSÃO

A aplicação de métodos de redução de dimensionalidade é importante, pois à busca do conhecimento útil e padrões em base de dados dispensa a presença de um número significativo de atributos.

A utilização da base de dados com todos os atributos podem prejudicar o desempenho do processo de aprendizagem dos algoritmos. Assim, há a necessidade de se aplicar métodos que garantam a qualidade dos dados que chegam à fase da Mineração de Dados. A quantidade de informações redundantes pode confundir o algoritmo e não auxiliar na busca de um modelo correto para o conhecimento.

Esse trabalho identificou cinco etapas principais para a comparação dos dois métodos de redução: o Método de Seleção de Atributos e Aplicação dos Conceitos de *Framework* para avaliação. Considerando o Método de Seleção de Atributos foram utilizadas duas abordagens: Filtro e *Wrapper*. No Método de Aplicação de Conceitos de *Framework* usou-se o Método adaptado de Ben-Abdalhal et al. (2004).

Esses dois métodos foram aplicados no domínio de Cliente, usando três bases de dados nos segmentos: Bancário, Vendas e Seguros. Neste trabalho, estas bases foram denominadas de: *Stalog, Customer* e *Insurance*, respectivamente.

Analisando os resultados obtidos, utilizando como critérios de avaliação a Validação Cruzada verificou-se que o uso dos métodos resultaram em uma melhora nos valores de taxa de acerto quando comparado com as bases possuindo todos os atributos.

No Método de Seleção de Atributos os melhores resultados foram para as bases *Stalog* e *Insurance.* Porém, para a base *Customer* a base com todos os atributos apresentou os melhores resultados.

Entre os dois métodos de abordagens: Filtro e *Wrapper* foi possível verificar que nas bases *Stalog* e *Customer* a melhor Abordagem de Seleção de Atributos foi a *Wrapper* juntamente com o algoritmo *SVM*. Essa abordagem normalmente apresenta um desempenho superior frente aos demais algoritmos de acordo com a literatura, fato também observado na presente pesquisa.

Com relação à base *Insurance* o melhor desempenho do algoritmo de seleção foi verificado para o *CFS*, na abordagem Filtro. Ressaltando que para esta

base não houve a geração de subconjuntos na abordagem *Wrapper*, pois a busca utilizada foi a sequencial. Neste tipo de busca um subconjunto vazio inicia o processo, o qual procura pelo melhor atributo, e a cada iteração compara-o com os demais. Caso não ocorra um melhor resultado entre as iterações, a busca é finalizada e nenhum subconjunto é classificado como melhor.

O algoritmo *CSE* apresentou o pior desempenho na Seleção de Atributos para as bases *Customer* e *Insurance.* Para a base *Stalog* o pior desempenho foi identificado com a aplicação do algoritmo *CFS*.

Os resultados da aplicação dos Conceitos de *Framework* foram melhores do que a base preparada com todos os atributos. Isto demonstra que o processo de categorização da base melhora o processo de Mineração de Dados.

Referente a identificação dos Subconjuntos de Atributos Comuns, somente para a base *Insurance* o resultado foi superior ao da base categorizada com todos os atributos. Já o Subconjunto de Atributos Específicos para todas as bases, os resultados da taxa de acerto foram superiores do que a base categorizada com todos os atributos. Esse resultado mostra que a Aplicação de Conceitos de *Framework* produz os melhores resultados para a taxa de acerto dos algoritmos classificadores.

Ao comparar os dois métodos de redução foi verificado que para a base *Stalog* o melhor foi com a aplicação da Seleção de Atributos, utilizando a Abordagem *Wrapper* com o algoritmo *SVM*, gerando desta forma o melhor subconjunto de atributos. Já para a base *Customer* e *Insurance* os melhores subconjuntos gerados foram com a Aplicação dos Conceitos de *Framework*.

Nesse experimento verificou-se que o processo de categorização da base de dados melhorou a qualidade dos dados, ou seja, se fosse aplicado alguma tarefa de Mineração de Dados para construção de um modelo de conhecimento esse procedimento poderia melhorar o desempenho do algoritmo na classificação.

5.1 TRABALHOS FUTUROS

Alguns trabalhos que podem ser desenvolvidos a partir desta pesquisa são os seguintes:

- Realizar novos experimentos em outros domínios para verificar a aplicação dos conceitos de *Framework.*
- Criar uma ferramenta automatizada capaz de identificar os atributos comuns e específicos usando a Linguagem Natural.
- Elaborar um algoritmo de redução de dimensionalidade usando os conceitos de *Framework* abordados nesse trabalho.
- Utilizar a base categorizada obtida por meio dos conceitos de *Framework* no processo de Mineração de Dados.
- Aplicar novos testes com a busca aleatória da Seleção de Atributos para comparação de resultados.

REFERÊNCIAS

AGRAWAL, R.; SRIKANT, R. Mining generalized association rules in large relational tables. In: CONFERENCE ON VERY LARGE DATABASES, 21st Int`L. **Proceedings…** Zurich (SUI), 1995. Disponível em: <http://rakesh.agrawal-family.com/pubs.hyml>. Acesso em: 7 jun. 2009.

ALMUALLIM, H. E.; DIETTERICH, T. G. Learning with many irrelevant features. In: NATIONAL CONFERECE ON ARTIFICIAL INTELLIGENCE. 9. **Proceedings…** Anaheim (CA), 1991, v. 2, p. 547-552.

BASGALUPP, M. P. **LEGAL-Tree:** um algoritmo genético multi-objetivo lexicográfico para indução de árvores de decisão. Tese (Doutorado) - Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo. São Carlos, 2010. Disponível em: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12052010-165344/publico/teseMarcioBasgalupp.pdf>. Acesso em: 27 jun. 2011.

BEN-ABDALLAH, H.; BOUASSIDA, N.; GARGOURI, F.; BEN-HAMADOU, A. A UML-based *Framework* design method. **Journal of Object Technology,** v. 3, n. 8, p. 98-119, sept./oct. 2004. Disponível em: <http://www.jot.fm/issues/issue_2004_09/article1.pdf>. Acesso em: 12 jan. 2012.

BERRY, M. J. A.; LINOFF, G. S. **Data mining techniques:** for marketing, sales, and customer relationship management. Indianapolis, Ind.: Wiley, 2004.

BOENTE, A. N. P.; GOLDSCHMIDT, R. R.; ESTRELA, V. V. Uma metodologia de suporte ao processo de descoberta de conhecimento em bases de dados. In: SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, 5., 2008. Resende (RJ). **Anais...** 2008. v. 1. p. 4-5. Disponível em: <http://www.boente.eti.br/publica/seget2008kdd.pdf>. Acesso em: 27 jun. 2011.

_____.; OLIVEIRA, F. S. G.; ROSA, J. L. A. Utilização de ferramenta de KDD para integração de aprendizagem e tecnologia em busca da gestão estratégica do conhecimento na empresa. In: SIMPÓSIO DE EXCELÊNCIA EM GESTÃO E TECNOLOGIA, 4., 2007. **Anais...** v. 1, p. 123-132, 2007. Disponível em: <http://www.aedb.br/seget/artigos07/1219_Artigo%20SEGET%202007.pdf>. Acesso em: 27 jun. 2011.

BORGES, H. B. **Redução de dimensionalidade em bases de dados de expressão gênica**. Dissertação (Mestrado) – Pós-Graduação em Informática. Pontifícia Universidade Católica do Paraná. Curitiba, 2006. Disponível em: <http://www.ppgia.pucpr.br/pesquisa/mining/Dissertacoes/DissertacaoHelyane2006.pdf>. Acesso em: 27 jun. 2011.

BRACHMAN, R. J.; ANAND, T. **The process of knowledge discovery in databases**. The KDD Process for Extracting Useful Knowledge from Volumes of Data, p. 37-57, 1996. Disponível em: <http://dl.acm.org/citation.cfm?id=257944> Acesso em: 27 jun. 2011.

BRAGA, R. T. V. **Um processo para construção e instanciação de *Framework*s baseados em linguagem de padrões para um domínio específico.** 232 f. 2003. Tese (Doutorado em Ciência da Computação) – Universidade de São Paulo. São Carlos, 2003.

BUDD, T. **An introduction to object oriented programming.** 3. ed. Boston: Addison Wesley, 2002.

CARVALHO, R. B.; FERREIRA, M. A. T. Using information tecnology to support knowledge conversion processes. **Information Research,** v. 7, n. 1, out. 2001. Disponível em: <http://informationr.net/ir/7-1/paper118.html>. Acesso em: 16 abr. 2011.

CAVÕES, T, F. **Seleção de atributos via agrupamento**. Dissertação (Mestrado) - Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo. São Carlos, 2010.

CHEN, M.S.; HEN, J.; YU, P. S. Data Mining: an overview from a database perspective. **IEEE Transactions on Knowledge Data Engineering,** v. 8, n. 6, p. 866-883,1996.

CHEN, Y. L., HSU, C. L., CHOU, S. C. Constructing a multi-valued and multilabeled decision tree. **Expert Systems with Applications**, v. 38, n. 2, part 1, p. 4339-4347, ago. 2003. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0957417403000472>. Acesso em: 20 de dez. 2011.

CORNELIS, C.; JENSEN, R.; HURTADO, G.; SLEZAK, D. Attribute selection with fuzzy decision reducts. **Information Sciences,** v. 180, n. 2, p. 209-224, 15 jan. 2010.

CORTES, C.; VAPNIK, V. Support vector networks. **Machine Learning,** v. 20, n 3, p. 273-297, 1995. Disponível em: <http://image.diku.dk/imagecanon/material/cortes vapnik95.pdf> Acesso em: 5 jul. 2011.

DASH, M.; LIU H. Feature selection for classification. **Intelligent Data Analysis - an International Journal,** v. 1, n. 3, p. 131-156, 1997.

DEVIJVER, P. A.; KITTLER, J. **Pattern recognition:** a statistical approach. Englewood Cliffs: Prentice Hall: 1982.

DIAS, M. M.; PACHECO, R. C. S. Uma metodologia para o desenvolvimento de sistemas de descoberta de conhecimento. **Acta Scientiarum (UEM),** Maringá (PR), v. 27, n. 1, p. 61-72, 2005.

DU, B.; LAM, J. Stability analysis of static recurrent neural networks using delay-partitioning and projection. **Neural Networks,** v. 22, n. 4, may 2009, p. 343-347.

DUARTE, D. **Utilizando técnicas de programação lógica indutiva para mineração de banco de dados relacional**. Dissertação (Mestrado) - Pós Graduação em Informática. Universidade Federal do Paraná, Curitiba, 2001.

DUDA, R. O.; HART. P.E.; STORK. D.G. **Pattern Classification**. 2. ed. New York: Wiley Interscience, 2000.

DY, J. G. Unsupervised feature selection. In: COMPUTACIONAL methods of feature selection. London: Chapman & Hall/CRC, 2007. Cap. 2, p. 19-39.

ESTIVIIL-CASTRO, V. Why so many clustering algorithms: a position paper. **SIGKDD Explorations,** v. 4, part. 1, p. 65-75, 2002.

FACELI, K. **Um framework para análise de agrupamento baseado na combinação multi-objetivo de algoritmos de agrupamento**. 161f. 2006. Tese (Doutorado) - Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo. São Carlos, 2006. Disponível em: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-12012007-082216/pt-br.php>. Acesso em: 13 maio 2011.

FAN, W., GORDON, M. D.; PATHAK, P. Effective profiling of consumer information retrieval needs: a unified framework and empirical comparison. **Decision Support Systems,** v. 40, n. 2, p. 213-233, aug. 2005. Disponível em: <http://www.sciencedirect.com/science/article/pii/S0167923604000144>. Acesso em: 5 jul. 2011.

FAYAD, M. E., SCHMIDT, D. C. Object-oriented Application frameworks. **Communications of the ACM,** v. 40, 10 p., 1997.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI Magazine,** v.17, p. 37-54, 1996.

FERNANDES, V. Visão atual da TI no chão de fábrica ERP, automação e controle. In: CONGRESS0 PROINDÚSTRIA MECÂNICA. **Anais...** São Paulo: ProIndústria, 2006.

FLEURY, A. C. C. A Engenharia de Produção nos próximos 50 anos. ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO (ENEGEP), 28., 2008. Rio de Janeiro. **Anais...** Rio de Janeiro: ABEPRO, 2008.

FRAWLEY, W.; PIATETSKY-SHAPIRO, G.; MATHEUS, C. Knowledge discovery in databases: an overview. **AI Magazine,** v. 14, n. 3, p. 57-70, 1992.

GALVÃO, N. D. **Aplicação da mineração de dados em bancos da segurança e saúde pública em acidentes de transporte.** 129 f. 2009. Tese (Doutorado) - Escola Paulista de Medicina. Universidade Federal de São Paulo. São Paulo, 2009.

GAVA, V. L.; SPINOLA, M. M.; NOMURA, L.; GONÇALVES, R. F. Comércio eletrônico: aspectos que devem ser considerados em sua análise/ implementação e avaliação no mercado brasileiro. In: ENCONTRO NACIONAL DE ENGENHARIA DE PRODUÇÃO (ENEGEP). 25., 2005. **Anais...** Porto Alegre: ENEGEP, 2005.

GOLDSCHIMIDT, R.; PASSOS, E. **Data mining:** um guia prático. Rio de Janeiro: Campus, 2005.

GUYON, I.; ELISSEEFF, A. An introduction to variable and feature selection. **Journal of Machine Learning Research,** v. 3, p. 1157-1182, 2003.

HALL, M. A. **Correlation-based feature selection for machine learning**. 198 f. 1999. Thesis (Doctor of Philosophy) - Department of Computer Science. University of Waikato. Hamilton (New Zealand), 1999. Disponível em: <http://www.cs.waikato.ac.nz/~mhall/thesis.pdf> Acesso em: 5 jul. 2011.

HECKERMAN, D. Bayesian networks for data mining. **Data Mining and knowledge Discovery Journal,** v. 1, n. 1, p. 79-119, mar. 1997.

HRUSCHKA, E. R.; COVÕES, T. F.; HRUSCHKA JR., E. R.; EBECKEN, N. F. F. Feature selection for clustering problems: a hybrid algorithm that iterates between k-means and bayesian filter. In: INTERNACIONAL CONFERENCE ON HYBRID INTELLIGENT SYSTEMS (HIS 2005), 5., 2005. **Proceedings…** 2005, p. 405-410.

JIANG, T.; TUZHILIN, A. Improving personalization solutions through optimal segmentation of customer bases. **IEEE Transactions on Knowledge and Data Engineering,** v. 21, n. 3, p. 305-320, mar.2009. Disponível em: <http://pages.stern.nyu.edu/~atuzhili/pdf/701_Jiang_Tianyi.pdf>. Acesso em: 19 jul. 2011.

JOHNSON, R. E. Frameworks, components, patterns. In: SYMPOSIUM ON SOFTWARE REUSABILITY, 1997. **Proceedings...** Massachusetts (USA): ACM Press, 1997. p. 10-17.

_____. How to design frameworks. In: CONFERENCE ON OBJECT-ORIENTED PROGRAMMING: Systems, Languages and Applications. 8., 1993. Washington. **Proceedings…** Tutorial Notes, 1993, p. 567-617.

_____. FOOTE, B. Designing reusable classes. **Journal of the Object-Oriented Programming,** v. 1, n. 2, p. 22-35, jun./jul. 1988.

KIM, J. K., SONG, H. S., KIM, T. S.; KIM, H. K. Detecting the change of customer behavior based on decision tree analysis. **Expert Systems with Applications**. v. 22, n. 4, part 1, p. 193-205, set. 2005. Disponível em: <http://onlinelibrary.wiley.com/doi/10.1111/j.1468-0394.2005.00310.x/abstract>. Acesso em: 20 de dez. 2011.

KIRA, K.; RENDELL, L. A. The feature selection problem: traditional methods and a new algorithm. In: CONFERENCE ON ARTIFICIAL INTELLIGENCE, 10., 1992, **Proceedings...** Menlo Park (CA), 1992, p. 129-136.

KOHAVI, R.; JOHN, G. H. Wrappers for feature subset selection. **Artificial Intelligence,** v. 97, p. 273-324, 1997. Disponível em: <http://ai.stanford.edu/~ronnyk/wrappersPrint.pdf>. Acesso em 11 nov. 2011.

KONONENKO, I. **Estimating attributes**: analysis and extention of relief. Amsterdam, 1994. p. 171-182.

KOTLER, P. **Administração de marketing.** 2. ed. São Paulo: Prentice Hall, 2000.

_____. KELLER, K. **Administração de marketing.** 12. ed. São Paulo: Prentice Hall, 2006.

KRACKLAUER, A. H.; MILLS, D. Q.; EIFERT, D. Customer management as the origin of collaborative customer relationship management. In: _____. **Collaborative customer relationship management:** taking CRM to the next level. Boston (MA): Springer, 2004. p. 3-6.

KUBO, M. M. **FMMG:** um *Framework* para jogos *multiplayer* móveis. Tese (Doutorado em Engenharia) - Escola Politécnica. Universidade de São Paulo. São Paulo, 2006.

LANDIN, N.; NIKLASSON, A. **Development of object-oriented *Framework*s.** 154 f. 1995. Master Thesis (Doutoral in Communication Systems) – Lund, Sweden. 1995.

LEE, H. D. **Seleção de atributos importantes para a extração de conhecimento de base de dados**. Tese (Doutorado). Universidade de São Paulo, 2005. Disponível em: <http://www.teses.usp.br/teses/disponíveis/55/55134/tde-22022006-172219> Acesso em: 17 jun. 2010.

LEE, S.; PARK, Y. Customization of technology roadmaps according to roadmapping purposes: Overall process and detailed modules. **Technological Forecasting & Social Change,** v. 72, p. 567-583, 2005. Disponível em: <http://www.maoner.com/Cited_Saritas_Oner_2004.pdf> Acesso em: 5 jul. 2011.

LIMA, A. R. G. **Máquinas de vetores suporte na classificação de impressões digitais**. 81f. 2002. Dissertação (Departamento de Computação) Universidade Federal do Ceará, Fortaleza, 2002. Disponível em: <http://www.dominiopublico.gov.br/pesquisa/DetalheObraForm.do?select_action=&co_obra=160842> Acesso em: 8 jul. 2011."

_____. MOTODA, H. **Feature selection for knowledge discovery and data mining.** Boston; Dordrecht: Kluwer Academic, 1998.

_____. SETIONO, R. A probabilistic approach to feature selection. In: INTERNATIONAL CONFERENCE ON MACHINE LEARNING. 13., 1996. **Proceedings…** 1996, p. 319-327. Disponível em: <http://wenku.baidu.com/view/462b884cfe4733687e21aa3d.html> Acesso em: 8 jul. 2011.

_____. YU, L. Toward integrating feature selection algorithms for classification and clustering. **IEEE Transactions on knowledge and Data Engineering.** v. 17, n. 4, p. p. 491-502, 2005. Disponível em: <http://www.public.asu.edu/~huanliu/papers/tkde05.pdf> Acesso em: 8 jul. 2011.

MATOS, S. N. **Um método dirigido por responsabilidade para obtenção antecipada de pontos de estabilidade e de flexibilidade no desenvolvimento de *Framework*s de domínio.** 275 f. 2009. Tese (Doutorado em Ciências) - Pós-Graduação em Engenharia Eletrônica e Computação. Instituto Tecnológico de Aeronáutica. São José dos Campos, 2009.

MENDES, M. **Modelagem de domínio para leigos.** Disponível em: <http://marcomendes.com/blog/2011/08/modelagem-de-dominio-para-leigos/>. Acesso: em 20 nov. 2011.

METROPOLIS, N.; ROSENBLUTH, A. W.; ROSENBLUTH, M. N.; TELLER, A. H. Equations of state calculations by fast computing machines. **Journal of Chemical Physics,** v. 21, n. 6, p. 1087-1092, jun. 1953.

MICHIE, D.; SPIEGELHALTER, D.; TAYLOR, C. **Machine learning, neural and statistical classifications.** Ellis Horwood, 1994. Disponível em: <http://www1.maths.leeds.ac.uk/~charles/statlog/whole.pdf> Acesso em: 17 mar. 2011.

MICROSOFT. **Particionando dados em conjuntos de treinamento e teste.** Disponível em: <http://technet.microsoft.com/pt-br/library/bb895173(SQL.100).aspx> Acesso em: 17 mar. 2011.

MITCHELL, T. M. **Machine learning.** Boston: WCB/McGraw-Hill, 1997.

MITRA, S.; PAL, S. K.; MITRA, P. Data mining in soft computing Framework: a survey. **IEEE Transactions On Neural Networks,** v. 13, n. 1, p. 3-14, 2002. Disponível em: <http://repository.ias.ac.in/26054/1/310.pdf>. Acesso em: 6 ago. 2011.

MOINO, C. A. A. **Metodologia para projeto inverso de aerofólios em grades de turbomáquinas via otimização por busca aleatória controlada.** Dissertação (Mestrado) - Instituto de Engenharia Mecânica. Universidade Federal de Itajubá, 2006. 74f. Disponível em: <http://www.portal.unifei.edu.br/files/arquivos/PRPPG/Engenharia_mecanica/Dinamica_fluidos_maquinas_fluxo_mestrado/Carlos_Alberto_Amaral_Moino.pdf>. Acesso em: 20 dez. 2011.

NGAI, E, W, T.; XIU, L.; CHAU, D. C. K. Application of data mining techniques in customer relationship management: a literature review and classification. **Expert Systems with Application,** v. 36, n. 2, part 1, p. 2592-2602, mar. 2009.

NICOLAIO, R. A.; PELINSKI, R. **Estudo e aplicação da tarefa de associação de Data Mining em uma base de dados real**. 2006. 44 f. Trabalho de Conclusão de Curso. (Graduação em Superior de Tecnologia em Sistemas de Informação) - Universidade Tecnológica Federal do Paraná. Ponta Grossa, 2006.

OLAFSSON, S.; LI, X.; WU, S. Operations and data mining. **European Journal of Operational Research,** v. 187, n. 3, p. 1429-1448, 2008. Disponível em: <http://www.comp.nus.edu.sg/~rudys/arnie/or-datamining.pdf>. Acesso em: 8 jul. 2011.

OMID, M. Design of an expert system for sorting pistachio nuts through decision tree and fuzzy logic classifier. **Expert Systems with Application,** v. 38, n. 4, part 1, p. 4339-4347, out. 2010.

ORFALY, E. **The essential distributed objects survival guide.** Object *framework*s: an overview. Boston: John Wiley & Sons, 1995. p. 221-238.

PARK, Y. J.; CHANG, K. N. Individual and group behavior-based customer profile model for personalized product recommendation. **Expert Systems with Application,** v. 36, n. 2, part 1, p. 1932-1939, mar. 2009. Disponível em: <http://linkinghub.elsevier.com/retrieve/pii/S0957417407006288> Acesso em: 8 jul. 2011.

PERISSINOTTO, M. **Sistema inteligente aplicado ao acionamento do sistema de climatização em instalações para bovinos leiteiros**. Tese (Doutorado em Agronomia). Escola Superior de Agricultura Luiz de Queiroz. Piracicaba, 2007. Disponível em: <http://www.teses.usp.br/teses/disponiveis/11/11131/tde-04032008-181236/publico/mauricioperissinotto.pdf> Acesso em: 8 jul. 2011.

PRAHALAD, C. K.; KRISHNAN, M. S. **A nova era da inovação:** a inovação focada no relacionamento com o cliente. Rio de Janeiro: Elsevier, 2008.

QUINLAN, J. R. **C4.5:** programs for machine learning. San Mateo, CA: Morgan Kaufmann Publishers, 1993.

REZENDE, S. O. **Sistemas inteligentes:** fundamentos e aplicações. Barueri (SP): Manole, 2003.

ROMDHANE, L. B.; FADHEL, N.; AYEB, B. An efficient approach for building customer profiles from business data. **Expert Systems with Applications,** v. 37, n. 2, p. 1573-1585, mar. 2010.

RUSE, M.; TRAVIS, J. **Evolution:** the first four billion years. London: The Belknap Press of Havard University Press, 2009.

RUSSEL, S.; NORVIG, P. **Inteligência artificial**. 2. ed. Rio de Janeiro. Campus, 2004.

SANTOS, E. M. **Teoria e aplicação de *support vector machines* à aprendizagem e reconhecimento de objetos baseado na aparência.** 121 f. 2002. Dissertação (Mestrado) - Universidade Federal da Paraíba. João Pessoa, 2002. Disponível em: <http://docs.computacao.ufcg.edu.br/posgraduacao/dissertacoes/2002/DissertacaoEulandaMirandadosSantos.pdf>. Acesso em: 20 dez. 2011.

SERRATO, F. Arquitetura.Net. **Camada de componentes de acesso a dados**. 2005. Disponível em: <http://imasters.uo.com.br/artigo/3763/dotnet/arquitetura_net_camanda_de_componentes_de_acesso_a_dados/>. Acesso em 12 de dezembro de 2011.

SILBERSCHATZ, A.; KORTH.; H. F.; SUDARSHAN, S. **Sistemas de banco de dados.** 5. ed. Rio de Janeiro: Elsevier, 2006.

SILVA S. M.; SILVA, W. V.; CORSO, J. M. D.; DUCLOS, L. C. Segmentação de mercado: análise do perfil sócio-econômico dos municípios do Paraná. **Informe GEPEC,** v. 10, n. 2, jul./dez., 2006.

SONG, X.-F.; CHEN, W.-M.; CHEN, Y.-P. P.; JIANG, B. Candidate working set strategy based SMO algorithm in support vector machine. **Information Processing and Management,** n. 45, p. 584-592, 2009.

SVEIBY, K. E. **A nova riqueza das organizações:** gerenciando e avaliando patrimônios de conhecimento. 3. Ed. Rio de Janeiro: Campus, 1998.

TAN, C-P.; LIM, K-S.; LAI, W. K. Multi-dimensional features reduction of consistency subset evaluator on unsupervised expectation maximization classifier for imaging surveillance application. **International Journal of Image Processing,** v. 2, n. 1, p. 18-26, 2008. Disponível em: <http://www.cscjournals.org/csc/manuscript/Journals/IJIP/Volume2/Issue1/IJIP-7.pdf>. Acesso em: 13 maio 2011.

TAN, K. C.; TEOH, E. J.; YU, Q., GOH, K. C**.** A hybrid evolutionary for attribute selection in data mining**. Expert Systems with Application,** v. 36, n. 4, part 1, p. 8616-8630, mai. 2009. Disponível em: <http://www.sciencedirect.com/science/article/pii/S095741740800729X>. Acesso em: 20 dez. 2011.

TAVARES, C.; BOZZA, D.; KONO, F. Descoberta de conhecimento aplicado a dados eleitorais. **Revista Gestão & Conhecimento,** v. 5, n. 1, p. 54-9, jan./jun. 2007: Disponível em: <http://gc.facet.br/v5n1/pdf/descoberta_de_conhecimento_aplicado_a_dados_eleitorais.pdf>. Acesso em: 13 maio 2011.

TURRIONI, J, B.; MELLO, C., H., P. **Metodologia de pesquisa em engenharia de produção:** estratégias, métodos e técnicas para condução de pesquisas quantitativas e qualitativas. Programa de Pós-Graduação em Engenharia de Produção. Universidade Federal de Itajubá , 2011. Disponível em: <http://www.carlosmello.unifei.edu.br/Disciplinas/Mestrado/PCM-10/Apostila-Mestrado/Apostila_Metodologia_Completa_2012.pdf>. Acesso em: 20 de dez. 2011.

UCI Machine Learning Repository. **Browse Through:** 5 data sets. Disponível em: <http://archive.ics.uci.edu/ml/datasets.html?format=&task=&att=&area=bus&numAtt=&numIns=&type=&sort=nameUp&view=table>http://archive.ics.uci.edu/ml/>. Acesso em: 20 dez. 2011.

WANG, K., ZHOU, S., YANG, Q.; EUNG, J. M. S. Mining customer value: from association rules to direct marketing. **Data Mining and Knowledge Discovery,** v. 11, p. 57-79, 2005.

WEKA. The University of Waikato. **Software.** Disponível em: <http://www.cs.waikato.ac.nz/ml/weka/>. Acesso em: 24 jan. 2011.

XU, R.; WUNSCH, D. Survey of clustering algorithms. **IEEE Transactions on Neural Networks,** v. 16, n. 3. p. 645-678, 2005.

YAMAMOTO, C.; H. **Visualização como suporte à extração e exploração de regras de associação.** Tese (Doutorado) - Instituto de Ciências Matemáticas e de Computação. Universidade de São Paulo. 167f. 2009. Disponível em: <http://www.teses.usp.br/teses/disponiveis/55/55134/tde-18062009-152148/en.php>. Acesso em: 13 maio 2011.

YASSIN, A.; FAYAD, M. E. Application Frameworks: a survey. In: FAYAD, M. E.; JOHNSON, R. E. **Domain-specific application Frameworks:** Frameworks experience by industry. New York: John Wiley & Sons, 2000. Cap. 29, p. 615-632.

APÊNDICE A - ATRIBUTOS SELECIONADOS PARA CADA BASE

ATRIBUTOS SELECIONADOS PARA CADA BASE

Os atributos selecionados nos Métodos de Seleção de Atributos e Aplicação dos Conceitos de *Framework* para as bases *Stalog*, *Customer* e *Insurance* serão apresentados nesse Apêndice.

Os Quadros 15, 16, 17, 18 e 19 ilustram os atributos selecionados com o uso da Seleção de Atributos para a base *Stalog.*

Algoritmo de seleção de atributos *CFS* – Abordagem Filtro
Status of existing checking account
Duration in month
Credit history

Quadro 15 - Atributos selecionados na base *Stalog* com o uso do algoritmo *CFS*
Fonte: Autoria própria

Algoritmo de seleção de atributos *CSE* – Abordagem Filtro
Status of existing checking account
Duration in month
Credit history
Purpose
Credit amount
Savings accounts/bonds
Present employment since
Personal status and sex
Others debtors/guarantors
Property
Age in years
Other installment plans
Job
Telephone

Quadro 16 - Atributos selecionados na base *Stalog com o uso do algoritmo CSE*
Fonte: Autoria própria

Algoritmo de seleção de atributos *Naive Bayes* – Abordagem *Wrapper*
Status of existing checking account
Credit history
Installment rate in percentage of disposable income
Others debtors/guarantors
Number of people being liable to provide maintenance for
Foreign worker

Quadro 17 - Atributos selecionados na base *Stalog* com o uso do algoritmo *Naive Bayes*
Fonte: Autoria própria

Algoritmo de seleção de atributos *J48* – Abordagem *Wrapper*
Status of existing checking account
Credit history
Credit amount
Installment rate in percentage of disposable income
Others debtors/guarantors
Age in years

Quadro 18 - Atributos selecionados na base *Stalog* com o uso do algoritmo *J48*
Fonte: Autoria própria

Algoritmo de seleção de atributos *SVM* – Abordagem *Wrapper*
Status of existing checking account
Duration in month
Credit history
Purpose
Credit amount
Savings accounts/bonds
Personal status and sex
Age in years
Number of existing credits at this bank
Number of people being liable to provide maintenance for

Quadro 19 - Atributos selecionados na base *Stalog* com o uso do algoritmo *SVM*
Fonte: Autoria própria

Os Quadros 20 a 24 ilustram os atributos selecionados com o uso da Seleção de Atributos para a base *Customer.*

Algoritmo de seleção de atributos *CFS* – Abordagem Filtro
Amount
Frequent
Recency
Domestic
Apparel
Kitchen
Blankets
Outdoor
Coats
Wappar
Hhappar

Quadro 20 - Atributos selecionados na base *Customer* com o uso do algoritmo *CFS*
Fonte: Autoria própria

Algoritmo de seleção de atributos *CSE* – Abordagem Filtro
Frequent
Recency
Ntitle
Domestic
Apparel
Leisure
Kitchen
Luxury
Promo13
Return
Mensware
Homeacc
Lamps
Linens
Blankets
Outdoor
Wappar
Jewelry
Statecod
Valratio
Dining

Quadro 21 - Atributos selecionados na base *Customer* com o uso do algoritmo *CSE*
Fonte: Autoria própria

Algoritmo de seleção de atributos *Naive Bayes* – Abordagem *Wrapper*
Homeval
Frequent
Recency
Telind
Return
Acctnum
Edlevel
Sex

Quadro 22 - Atributos selecionados na base *Customer* com o uso do algoritmo *Naive Bayes*
Fonte: Autoria própria

Algoritmo de seleção de atributos *J48* – Abordagem *Wrapper*
Frequent
Age
Return
Blankets
Job

Quadro 23 - Atributos selecionados na base *Customer* com o uso do algoritmo *J48*
Fonte: Autoria própria

Algoritmo de seleção de atributos *SVM* – Abordagem *Wrapper*
Frequent
Recency
Telind
Promo13
Homeacc
Numkids
Travtime
Dining

Quadro 24 - Atributos selecionados na base *Customer* com o uso do algoritmo *SVM*
Fonte: Autoria própria

Os Quadros 25 e 26 ilustram os atributos selecionados com o uso da Seleção de Atributos para a base *Insurance.*

Algoritmo de seleção de atributos *CFS* – abordagem filtro
Customer subtype see l0
Social class a
Average income
Contribution car policies
Contribution fire policies
Contribution boat policies
Contribution social security insurance policies
Number of private third party insurance
Number of car policies
Number of boat policies

Quadro 25 - Atributos selecionados na base *Insurance* com o uso do algoritmo *CFS*
Fonte: Autoria própria

Algoritmo de seleção de atributos *CSE* – Abordagem Filtro
Customer subtype see l0
Customer main type see l2
Singles
High level education
Lower level education
High status
Farmer
Middle management
Skilled labourers
Unskilled labourers
Social class c
Social class d
Rented house
1 car
No car
National health service
Income < 30000
Income 45-75.000
Income 75-122.000
Average income
Purchasing power class
Contribution private third party insurance see
Contribution car policies
Contribution moped policies
Contribution fire policies
Contribution boat policies
Contribution social security insurance policies
Number of private third party insurance
Number of car policies

Quadro 26 - Atributos selecionados na base *Insurance* com o uso do algoritmo CSE
Fonte: Autoria própria

ANEXO A - MÉTODO USADO COMO REFERÊNCIA PARA A APLICAÇÃO DOS CONCEITOS DE *FRAMEWORK*

MÉTODO DE BEN-ABDALHAH ET AL. (2004)

Para a construção de um *Framework*, Ben-Abdallah et al. (2004) define o processo utilizando três passos independentes: os diagramas de caso de uso, de classe e de sequência de cada aplicação-exemplo, em que se analisa o nome de cada um dos objetos, de forma manual, que compõem os diagramas.

Ao final da análise, o *Framework* fica composto pela identificação dos pontos de flexibilidade e estabilidade entre as aplicações-exemplos, os quais serão representados graficamente pelos diagramas de caso de uso, de classe e de sequência no formato *F-UML*.

Na figura 17 são ilustrados os três passos para a modelagem de *Framework*.

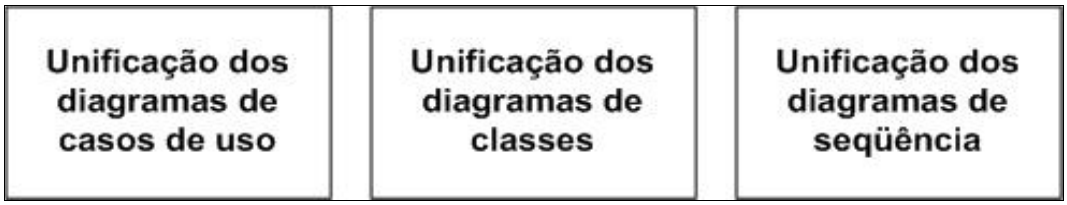


Figura 17 - Modelagem de *Framework*
Fonte: Ben-Abdallah et al. (2004)

A unificação dos diagramas de casos de uso é feita através da comparação semântica correspondente entre os nomes dos atores, casos de uso e suas relações (BEN-ABDALLAH et al., 2004).

O procedimento para unificação dos atores:

1. Consideram-se todas as aplicações-exemplo (AE_1,...,AE_{te}) sendo analisadas no domínio. Onde: *te* corresponde ao número de aplicações-exemplo.

2. Seleciona-se a aplicação AE_1 e compara cada ator de AE_1 com atores de AE_2, ..., AE_{te}. Esta comparação termina quando todos os atores da aplicação AE_1 tiverem sido comparados.

Os atores das aplicações AE_2, ..., AE_{te} que não coincidiram com nenhum ator comparado, são adicionados ao *Framework* como ponto de flexibilidade (*hot-spot*). Se o ator for encontrado em todas as aplicações-exemplo que será analisada, então este será definido como ator base. Caso contrário, será um ponto de flexibilidade.

À proporção que um ator é identificado durante o processo de comparação das aplicações-exemplo é dado pela seguinte equação:

$$R_{de} = \frac{\text{nº de ocorrência de } O_{AEi} \text{ em } AE_1, ..., AE_{te}}{te}$$

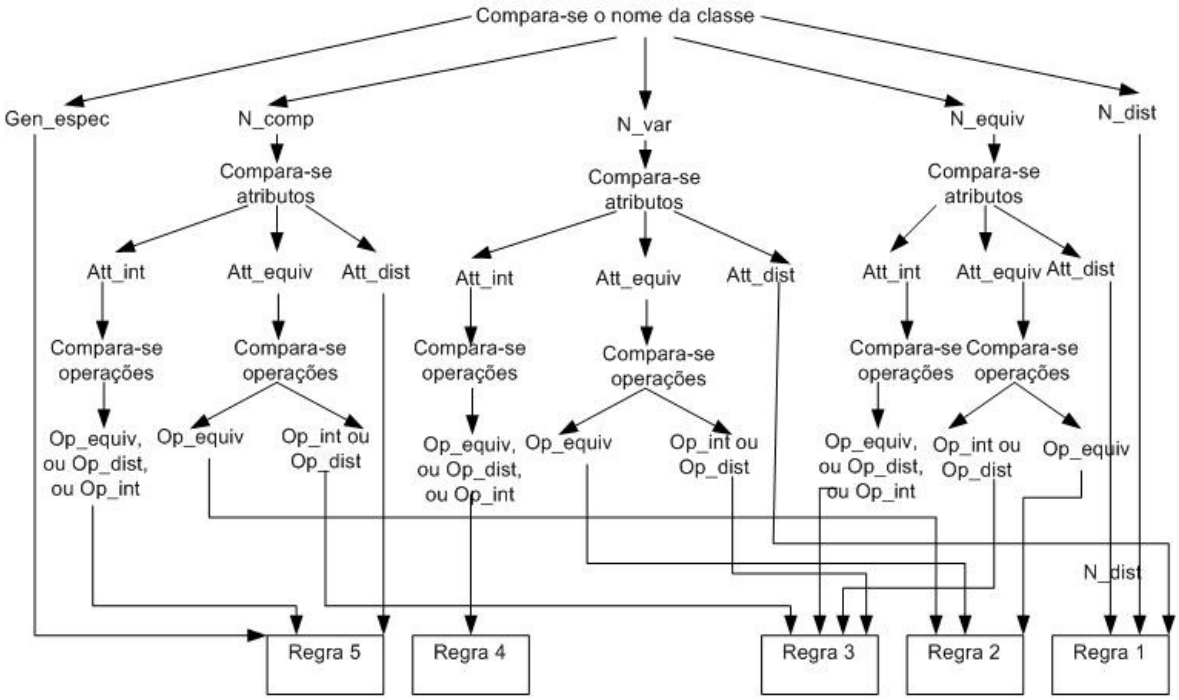
3. Transfere todos os relacionamentos entre os atores para o diagrama construído. Se a relação envolve um ator base, então o desenvolvedor decide que o ator herdado não representa todos os domínios, em consequência disso, a relação da herança é colocado um estereótipo <<incomplete>>.

Segundo Ben-Abdallah et al., (2004), a unificação dos diagramas de classe possui relações entre classes, atributos e operações que estão listadas abaixo:

- Classe: N_equiv, N_var, Gen_espec, N_comp e N_dist.
- Atributos: Att_equiv, Att_int, Att_conf, Att_dist.
- Operações: Op_equiv, Op _int, Op _conf, Op_dist.

As relações descritas anteriormente são também utilizadas para a comparação de atores pertencentes ao diagrama de caso de uso. Nesta ocorrência, compara-se o nome do ator e não o nome do caso de uso. De posse das relações inicia-se o processo de unificação tanto de ator quanto de caso de uso. O processo é o mesmo para o ator e para o caso de uso (BEN-ABDALLAH et al., 2004).

Para cada relação existe uma regra que será executada, ilustrada na Figura 18.



Regras:

Regra 1 - Adiciona-se C_{AE1} para o framework como ponto de flexibilidade se $R_{dc}(C_{AE1})=2/3$ ou se C_{AE1} está fora de uma classe que tenha sido considerada como base.

Regra 2 - Adiciona-se C_{AE1} para o framework base.

Regra 3 - Adicione a classe base contendo os atributos e métodos na interseção para o framework utilizando o esteriótipo <<extensible>>. O esteriótipo <<extensible>> mostra que a classe pode ser adaptável por adicionar ou remover atributos e operações. A hierarquia de classes adicionadas é do tipo caixa branca desde que a interface da classe possa ser mudada.

Regra 4 - As classes C_{AE1}...C_{AEe} são adicionadas ao framework como hierarquia de classes para uma nova classe base abstrata contendo os atributos e métodos da interseção utilizando o estereótipo <<extensible>>. A hierarquia de classes adicionadas é do tipo caixa branca desde que a classe de interface possa ser mudada.

Regra 5 - A classe de composição e as classes componentes são adicionadas ao framework usando composição. Os atributos e operações de interseção são colocados na classe de composição que será marcada como base.

Figura 18 - Regras para Identificação de Componentes em Uma Classe
Fonte: Ben-Abdallah et al., (2004)