Universidade Federal do Pampa

Autor: Alisson Jamie Cruz Lanot

MINERAÇÃO DE DADOS APLICADA NA IDENTIFICAÇÃO DA PROPENSÃO À EVASÃO NA UNIVERSIDADE

Trabalho de Conclusão de Curso II

BAGÉ
2012

ALISSON JAMIE CRUZ LANOT

MINERAÇÃO DE DADOS APLICADA NA IDENTIFICAÇÃO DA PROPENSÃO À EVASÃO NA UNIVERSIDADE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Computação.

Orientador: Prof. Dr. Sandro da Silva Camargo

Co-orientador: Prof. Dr. Cristian Cechinel

Bagé
2012

ALISSON JAMIE CRUZ LANOT

MINERAÇÃO DE DADOS APLICADA NA IDENTIFICAÇÃO DA PROPENSÃO À EVASÃO NA UNIVERSIDADE

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Engenharia de Computação da Universidade Federal do Pampa, como requisito parcial para obtenção do Título de Bacharel em Engenharia de Computação.

Trabalho de Conclusão de Curso defendido e aprovado em: 24 de Novembro de 2012.
Banca examinadora:

Prof. Dr. Sandro da Silva Camargo
Orientador
UNIPAMPA

Prof. Dr. Cristian Cechinel
Co-Orientador
UNIPAMPA

Profa. Dra. Ana Paula Lüdtke Ferreira
UNIPAMPA

AGRADECIMENTOS

Agradeço à minha família por sempre ter acreditado em mim, por estarem sempre ao meu lado, e por me apoiarem nos momentos difíceis.

Agradeço especialmente à minha irmã Marcele Cruz Lanot Antoniazzi, por ter permitido a minha confortável estadia em Bagé durante a realização do curso, e pela nossa amizade de longa data. Cheguei aqui graças a você!

Agradeço ao meu orientador Prof. Dr. Sandro Camargo pela proposta, e pela confiança depositada em mim para a realização deste trabalho. Agradeço também ao meu co-orientador Prof. Dr. Cristian Cechinel, pelo incentivo e auxílio na conclusão deste trabalho.

Agradeço à Secretaria Acadêmica da UNIPAMPA, por ter fornecido os conjuntos de dados essenciais para a execução deste trabalho.

RESUMO

A propensão à evasão na universidade é um problema de difícil identificação, visto que há diversos fatores que influenciam a sua ocorrência, e cada universidade pode ter motivos diferentes para a ocorrência de evasão. Identificar casos de evasão de forma manual é uma tarefa impraticável, visto que envolve a manipulação de grandes quantidades de dados. O objetivo deste trabalho é identificar dentre os dados disponíveis dos alunos, as características que contribuem para a evasão na Universidade Federal do Pampa, sendo que o estudo foi delimitado para o curso de Engenharia de Computação. Nesses conjuntos de dados são aplicados técnicas da Descoberta de Conhecimento em Banco de Dados, onde os dados serão tratados e posteriormente analisados. Esta atividade é realizada com o auxílio da ferramenta Weka, a fim de minerar os dados fornecidos pela própria Universidade. A partir da identificação dos fatores associados com a evasão, tem-se a intenção de permitir a identificação de uma situação em que o aluno tenha grandes riscos de abandonar os seus estudos, e com isto, permitir que sejam criados e otimizados os meios de prevenção da evasão na universidade. Através da aplicação de diferentes algoritmos de classificação e das técnicas de regras de associação e clusterização, foi possível verificar uma associação entre o fraco desempenho acadêmico com a evasão dos alunos.

Palavras-chave: Propensão à evasão. Mineração de Dados. Classificação. Regras de associação. Clusterização.

ABSTRACT

Dropout propensity is a problem difficult to identify in Universities, since there are several factors that may influence its occurrence, and students from different universities may have different reasons for dropping out. Identifying such cases one by one is an impractical task, since it requires dealing with large amounts of data. The main objective of this work is to identify features that may contribute students to drop out from the Federal University of Pampa (UNIPAMPA), using data collected from the university academic system. The present work was limited to the scope of students from the Computer Engineering undergraduate program of UNIPAMPA. The collected datasets were treated and analyzed in order to apply Knowledge Discovery in Databases (KDD) techniques. This task was done with the help of a tool called Weka, in which data from the university is mined. The intention here is to discover features associated with dropouts in order to identify situations of students in risk and then to find ways to create and optimize the prevention of these dropouts. Through the application of different classification algorithms, association rules and clustering techniques, it was possible to verify an association between the poor academic performance and the drop-outs.

Keywords: Drop-out propensity. Data Mining. Classification. Association Rules. Clustering.

SUMÁRIO

1 INTRODUÇÃO

1.1 Contextualização e Motivação

A Universidade Federal do Pampa (UNIPAMPA) foi criada através de um projeto do governo federal de expansão das instituições federais de ensino, com o propósito de fortalecer a economia e impulsionar o desenvolvimento tecnológico e científico das regiões oeste e sul do Rio Grande do Sul. Criada em 2006 e instituída legalmente em 2008[1], a UNIPAMPA conta com 540 professores, 564 técnicos-administrativos, cerca de 8.500 alunos regulares e 500 alunos graduados segundo dados apresentados no IV Seminário de Desenvolvimento Profissional Docente (CAP, NUDEPE E PROPLAN, 2011). Nesta apresentação também é informado que até 2011, o total de alunos deveria ser 11.520 e 1.210 alunos concluintes até 2010/2. Até 2010/2 houve 1.408 casos de evasão (JOSÉ; ANDREOLI, 2011). A desistência desses alunos é um fator preocupante, devido à grande quantidade de dinheiro público investido neles. Em consequência da desistência de tais alunos, novos editais devem ser divulgados para o preenchimento das vagas remanescentes. A divulgação de um novo processo seletivo gera mais custos para a instituição. Por ser uma universidade nova, onde gastos em infraestrutura ainda estão sendo feitos, a evasão se torna um fator crítico.

Além disso, a evasão tem por consequência uma perda de mão de obra especializada, assim como perda de competitividade nacional e menor eficiência produtiva nas empresas (SILVA FILHO et al., 2007 apud CAMPELLO; LINS, 2008).

Através do Projeto Institucional da universidade (UNIPAMPA, 2009) pode ser notado um fator interessante que ocorre na UNIPAMPA: por meio de uma pesquisa realizada em 2008 utilizando como amostra 67% dos alunos da universidade, verificou-se que 80% dos entrevistados são provenientes de escolas públicas, o que se diferencia da realidade das grandes universidades públicas brasileiras. Com a exceção da rede pública federal, as escolas públicas de ensino básico no Brasil tendem a possuir uma qualidade de ensino inferior em relação às escolas privadas (IWASSO, 2010), o que faz a UNIPAMPA ter uma grande responsabilidade para com os alunos ingressantes.

Grupos de trabalho foram criados na instituição para entender melhor o problema da evasão (JOSÉ; ANDREOLI, 2011). Tais grupos tem utilizado a análise manual para a abordagem do problema. Entretanto, ao trabalhar com grandes conjuntos de dados, a análise

[1] Lei Federal n. 11.640 de 11 de Janeiro de 2008, publicado no DOU de 23 de Janeiro de 2008.

de cada caso individual se torna inviável. Para isto, técnicas da Descoberta de Conhecimento em Banco de Dados (FAYYAD; SHAPIRO; SMYTH, 1996) podem ser aplicadas para auxiliar a manipulação da grande quantidade de dados disponíveis.

1.2 Objetivos

1.2.1 Objetivo Geral

Identificar, por meio da aplicação de técnicas de mineração de dados, de fatores pré-existentes que possam influenciar a evasão dos alunos de graduação da UNIPAMPA.

1.2.2 Objetivos Específicos

- Auxiliar no embasamento teórico para a criação de políticas de prevenção da evasão na UNIPAMPA.
- Apresentar uma visão geral sobre o processo de manipulação de dados para a realização de uma análise adequada através das técnicas de mineração de dados.
- Fornecer informações que possam ser utilizadas por grupos de trabalho em evasão da universidade para auxiliar a manipulação de outras bases de dados disponíveis pela instituição, seja informações acadêmicas de alunos de outros cursos, ou até mesmo novas bases de dados relevantes ao problema da evasão que possam surgir através de uma coleta de dados adicional.

Além disso, as ferramentas desenvolvidas neste trabalho podem ser utilizadas para auxiliar a realização de novos experimentos com Mineração de Dados para a identificação da propensão à evasão na UNIPAMPA, por meio da obtenção de novos conjuntos de dados dos alunos.

1.3 Estrutura do Trabalho

O trabalho está dividido da seguinte forma: inicialmente é descrito o processo da Descoberta de Conhecimento em Banco de Dados (DCBD), que é uma metodologia para encontrar padrões em grandes quantidades de dados, e também são descritas as técnicas mais comuns nela empregadas.

Em seguida é discutido brevemente o problema da evasão escolar, em especial para o

caso do ensino superior. Após este tratamento teórico é descrito o trabalho prático, que consiste na aplicação das técnicas da DCBD tendo como estudo de caso a evasão de alunos ocorrida no período de funcionamento da UNIPAMPA.

Para a abordagem desses dados, foi delimitado o conjunto de dados a ser utilizado somente para o curso de Engenharia de Computação do Campus Bagé, pois são os conjuntos de dados disponíveis no momento.

As técnicas aqui utilizadas e as ferramentas desenvolvidas neste trabalho poderão ser adaptadas para o uso nos conjuntos de dados dos outros cursos e de semestres posteriores em trabalhos futuros.

1.4 Trabalhos Relacionados

Várias abordagens utilizando técnicas de mineração de dados já foram empregadas para o problema da evasão na universidade. Um exemplo pode ser visto em Manhães et al. (2012), onde um grupo de pesquisa fez um estudo sobre a utilização da mineração de dados para traçar o perfil dos alunos regulares e evadidos da UFRJ, tendo como foco a comparação do desempenho para diversos algoritmos de mineração de dados.

Outro estudo interessante foi realizado no Centro Universitário Luterano de Ji-Paraná Cestaro (2006), onde o autor descreve técnicas de classificação aplicadas na identificação da propensão à evasão, tendo como característica uma quantidade limitada de dados disponíveis para a mineração.

Um exemplo de trabalho utilizando mineração de dados para identificar diversos grupos de alunos por meio de agrupamento por similaridade pode ser visto em Campello e Lins (2008), entretanto detalhes da implementação não são detalhados.

Em Obsivac et al. (2012) o autor foi além e utilizou dados relativos a forma que os estudantes se relacionam entre si, utilizando informações como troca de *e-mails* e fóruns de discussão participados pelos alunos.

2 REFERENCIAL TEÓRICO

Neste capítulo será abordado o embasamento teórico envolvido na realização deste trabalho. Primeiro será descrito o problema da evasão. Em seguida é dada uma introdução breve sobre Bancos de Dados, que são coleções de dados armazenados em um computador. Por fim é falado sobre as técnicas da Descoberta de Conhecimento em Banco de Dados, que consiste no processo de descobrir novos conhecimentos em coleções de dados.

2.1 Evasão Escolar

O sistema educacional é uma das formas mais tradicionais para o desenvolvimento econômico de um país. A educação serve para capacitar a população em vários níveis: partindo dos conhecimentos básicos para a convivência em uma sociedade civilizada, até um nível avançado do conhecimento voltado para pesquisa, ensino e desenvolvimento de novas tecnologias (SOUZA, 2011).

É sabido que o Brasil possui ensino público gratuito desde o ensino básico até a pós-graduação (SOUZA, 2011). Segundo Morais (2011), a evasão tem por consequência o gasto elevado do dinheiro público. Por ser um país emergente onde a desigualdade social é aparente, tais gastos devem ser otimizados a fim de melhor utilizar o dinheiro público.

Ao contrário do ensino básico, onde a evasão geralmente é caracterizada por questões como a exclusão social, a evasão no ensino superior é um assunto de maior complexidade de identificar suas razões e tratá-las, visto que consiste de pessoas alfabetizadas e aptas para entrar no mercado de trabalho (GARCIA; ABDALA; MATSUSHITA, 2000).

A evasão muitas vezes passa despercebida pelos professores, visto que alguns alunos que evadem não trancam a matrícula, e apenas deixam de frequentar as aulas. Com isto, nem sempre é possível fazer um acompanhamento adequado com tais alunos.

De acordo com Garcia, Abdala e Matsushita (2000), a evasão no ensino superior é um problema de difícil mensuração, porque só pode ser compreendida após acompanhar uma geração completa de alunos, sendo que esta geração ocorre em sete anos, sendo, portanto, um estudo envolvendo todo o ciclo desta geração de alunos.

Várias razões podem ser observadas para a evasão no ensino superior, como, por exemplo, a dificuldade na adaptação às disciplinas iniciais do curso; evasão de um curso de menor concorrência como forma de ingresso em um de maior concorrência por meio de transferência interna; desmotivação do aluno em relação à afinidade com a área de estudo do

curso; ou até na baixa demanda do curso para o mercado de trabalho (ALVES; ALVES, 2010).

Entretanto, a evasão também pode ser consequência de problemas de difícil identificação, como por exemplo, problemas pessoais: sejam eles familiares, financeiros, ou afetivos (HARNIK, 2005). Tal classe de problemas é de difícil identificação por ser, geralmente, omitida pelo aluno.

A identificação da propensão à evasão escolar então se torna importante, visto que medidas podem ser tomadas para sua prevenção, seja na criação de políticas para a distribuição de bolsas aos estudantes ou até mesmo auxílio de um conselheiro vocacional ou de um psicólogo.

2.2 Banco de Dados

Bancos de dados são conjuntos de dados com uma estrutura regular, organizados de forma que algoritmos computacionais possam facilmente encontrar a informação desejada (DATE, 2003). Um banco de dados visa atender primariamente aos processos de inclusão, armazenamento, manipulação e consulta dos dados em questão.

Bancos de dados podem ser utilizados de diversas formas, como por exemplo, na verificação e manutenção da disponibilidade de produtos em estoque ou para manter uma relação atualizada de alunos matriculados em uma determinada instituição.

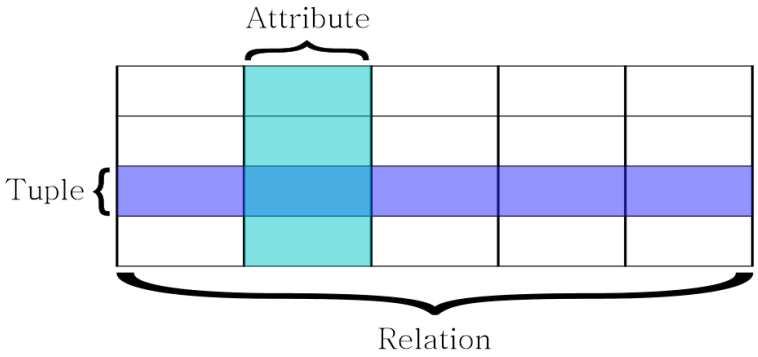
Em bancos de dados, uma das formas utilizadas para tratar tais dados é utilizando o *modelo relacional*. A seguir são descritas as características principais do modelo relacional de forma simplificada (DATE, 2003):

- Relações: Também conhecidas como tabelas, são as estruturas organizadas em linhas e colunas onde são armazenados os dados. É definida como um conjunto de *tuplas* que compartilham do mesmo *atributo*. Um exemplo de relação pode ser visto na figura 1.
- Tuplas: Consistem de uma lista ordenada de dados relacionados. Cada tupla na tabela possui a mesma estrutura. Também são conhecidas como *registros* ou *instâncias*.
- Atributos: Consistem na especificação de um determinado objeto da tupla, ou seja, são as características que estão sendo avaliadas na tabela. Um atributo pode ser representado em um formato numérico ou categórico. Um exemplo de atributo numérico é *idade*. Para atributos categóricos podemos ter, por exemplo, o sexo do aluno, tendo por categorias o sexo masculino e feminino. Outro exemplo para atributo categórico pode ser a *cidade*, sendo as diversas cidades possíveis categorias para o

atributo.

- Chaves: Uma chave consiste em um ou mais atributos que irão garantir a unicidade dos registros. Existem dois tipos de chaves:
    - Chave Primária: é uma chave única que será utilizada para a identificação dos registros da tabela.
    - Chave Estrangeira: uma chave declarada como estrangeira em uma tabela será relacionada com a chave primária de outra tabela, e irá garantir que cada registro nesta tabela possua apenas registros que possuem a mesma chave primária, visando manter a integridade das duas tabelas.

Figura 1 – Exemplo de relação



Fonte: (WIKIPEDIA, 2012a)

A popularização dos dispositivos eletrônicos como computadores e *smartphones* em conjunto com a *Internet* permitiu que atividades que fazemos no dia-a-dia possam ser armazenadas de forma *on-line*. Desta forma, horários de acesso a *websites* são utilizados na identificação padrões de acesso para diferentes perfis de usuários, assim como históricos de compras em lojas virtuais são utilizados para realizar sugestões de futuras compras (ABERNETHY, 2010).

O armazenamento de tais informações tem várias implicações, tanto no ponto de vista técnico, como a necessidade de unidades de armazenamento cada vez maiores, como também no ponto de vista social, envolvendo questões de privacidade e confidencialidade das informações coletadas (BERRY; LINOFF, 2004).

Ao armazenar grandes quantidades de dados por longos períodos de tempo, podem-se obter diversos dados estatísticos indicando a ocorrência de certos padrões, comportamentos ou tendências dentro de grandes grupos (HALL et al., 2009). Entretanto, com uma quantidade enorme de dados disponíveis, se torna impossível até mesmo para um especialista identificar

manualmente os padrões expressos nos dados.

Na seção 2.3 é descrita uma metodologia que utiliza o auxílio de computadores para a abordagem de grandes quantidades de dados.

2.3 Descoberta de Conhecimento em Banco de Dados

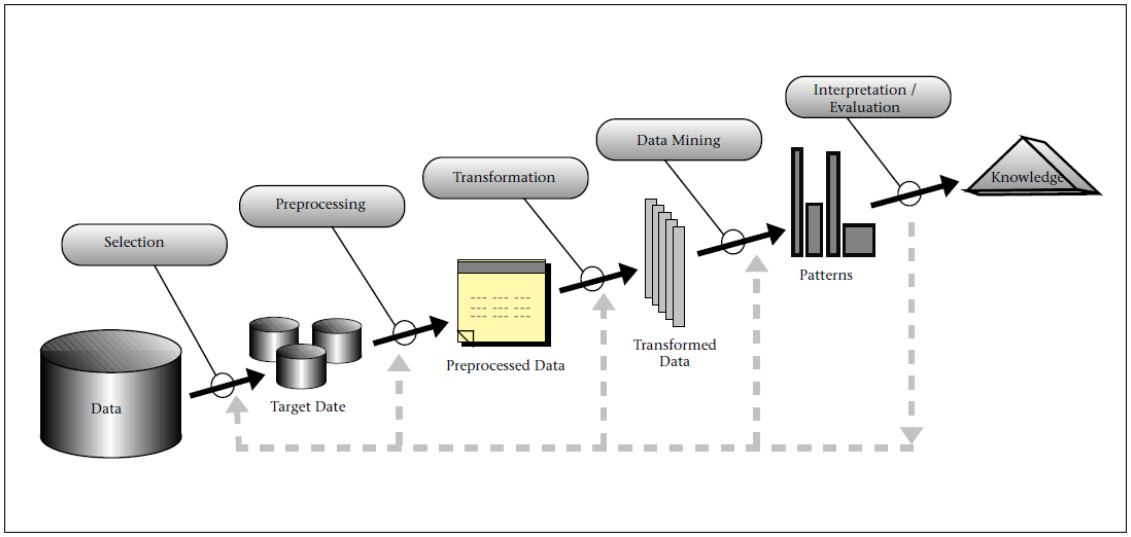
A Descoberta de Conhecimento em Banco de Dados (DCBD) (também conhecida pela sigla KDD, originária do seu termo em inglês *Knowledge Discovery in Databases*) consiste no processo de identificação automática de padrões não-triviais em dados (FAYYAD; SHAPIRO; SMYTH, 1996). Para isto, são aplicadas uma ou mais técnicas de Mineração de Dados para extrair padrões e avaliá-los nos dados (SYMEONIDIS; MITKAS, 2005).

A DCBD é um processo interativo e iterativo (SYMEONIDIS; MITKAS, 2005), o que significa que o processo irá interagir com quem está aplicando o mesmo e será progressivamente refinado. Embora a DCBD e a Mineração de Dados sejam dois termos frequentemente utilizados de forma intercambiável (GOEBEL; GRUENWALD, 1999), o termo Mineração de Dados é utilizado para denotar o procedimento da aplicação dos algoritmos nos dados enquanto a DCBD é um processo mais amplo, envolvendo desde a fase inicial, onde os dados serão preparados até a fase de avaliação e aplicação dos resultados.

Existem diversos modelos para descrever o processo da DCBD, divididos em três categorias (CIOS et al., 2007):

- Modelos acadêmicos: Os modelos da DCBD tiveram seu início no ambiente acadêmico, que foram criados para definir a sequência de passos necessária para guiar os usuários das ferramentas de Mineração de Dados. A criação de um modelo acadêmico surgiu com a necessidade de auxiliar a execução da DCBD em um domínio arbitrário. Um modelo acadêmico tradicional é o proposto por Fayyad, Shapiro e Smyth (1996).
- Modelos industriais: Logo após o desenvolvimento dos modelos acadêmicos, foram criados modelos orientados a negócio, com o objetivo de ter um modelo padronizado para a DCBD, tendo por características, por exemplo, nomenclatura de fácil compreensão e boa documentação. Modelos industriais conhecidos são, por exemplo, o modelo proposto por Cabena et al. (1998) assim como o CRISP-DM (CHAPMAN et al., 2000).
- Modelos híbridos: Combina aspectos dos modelos acadêmicos e modelos industriais. Um modelo híbrido conhecido é proposto em Cios e Kurgan (2005).

Figura 2 – Etapas do processo da DCBD.



Fonte: (FAYYAD; SHAPIRO; SMYTH, 1996)

Na figura 2 é demonstrado o modelo proposto por Fayyad, Shapiro e Smyth (1996), que é o primeiro modelo criado para a DCBD, sendo um modelo acadêmico largamente utilizado (CIOS et al., 2007). Sob um alto nível de abstração a DCBD pode ser dividida em três etapas (CAMARGO, 2010): Pré-Processamento, Mineração de Dados e Pós-Processamento. Cada uma dessas etapas é descrita a seguir:

2.3.1 Pré-Processamento

A etapa de pré-processamento inicia quando os dados já foram coletados e armazenados de alguma forma, seja em um arquivo texto, um arquivo de marcação, uma planilha eletrônica, ou em um banco de dados relacional. Com isto os dados irão passar por um processo a fim de garantir que o resultado não seja alterado, sejam através de dados faltantes, erros na entrada, ou erros de escala de representação. A qualidade dos dados é o fator mais importante que irá influenciar diretamente na qualidade dos resultados da análise (MYATT, 2006). As principais etapas são: Seleção dos Dados, Limpeza dos Dados, Integração dos Dados, Redução de ruídos e detecção de *outliers*, Transformação dos Dados e Redução de Dimensionalidade. Cada uma dessas etapas é descrita a seguir.

2.3.1.1 Seleção dos Dados

Após a coleta dos dados, são verificados os atributos e instâncias a serem utilizados para a aplicação dos algoritmos de mineração de dados. Os atributos podem ser escolhidos de duas formas: a primeira consiste na escolha dos atributos possivelmente relevantes pelo especialista do domínio. A segunda forma consiste na aplicação de técnicas de redução de dimensionalidade, que são utilizados quando o conhecimento do domínio é mínimo (CAMARGO, 2010).

2.3.1.2 Limpeza dos Dados

Os dados, de forma geral, não estão inicialmente em um formato pronto para serem minerados, pois frequentemente apresentam falhas na sua coleta ou no seu armazenamento, possivelmente por erros manuais. Tais erros podem ocorrer na forma de ruídos, dados omitidos e valores atípicos, que poderão ocasionar a alteração do resultado do algoritmo aplicado. Várias abordagens podem ser utilizadas para tratar dados omitidos (SYMEONIDIS; MITKAS, 2005):

- Ignorar o registro completamente.
- Preenchimento manual do registro.
- Identificar o dado faltante com um caractere especial.
- Usar a média do atributo para o preenchimento do dado faltante.
- Usar a média da classe[2] do atributo para o preenchimento do dado faltante.
- Preenchimento do dado com o valor mais provável.

2.3.1.3 Integração dos Dados

Dados podem ser obtidos de várias fontes de dados diferentes. Para a manipulação correta, tais dados devem ser unificados em um único conjunto de dados. A integração dos dados é feita em três níveis, que são descritos a seguir (SYMEONIDIS; MITKAS, 2005):

- Integração dos conjuntos de dados: consiste em identificar entidades idênticas ou equivalentes de fontes de dados diferentes. Um exemplo é fundir dois conjuntos de dados que possuam um atributo chave primária em comum.
- Detecção e resolução de conflitos nos dados: consiste em manipular os dados que estão em formatos de representação ou escala diferentes de forma a ficarem em uma

[2] Termo definido posteriormente na seção 2.3.2.1 Classificação

única unidade de representação. Um exemplo é utilizar conjuntos de dados que contém atributos com dados em escala métrica e escala imperial.

- Manipulação dos dados redundantes: consiste na identificação e eliminação de atributos redundantes, pois o mesmo atributo pode ter nomenclatura diferente em conjuntos de dados diferentes. Um exemplo é um atributo identificador (ID) em dois conjuntos de dados diferentes.

2.3.1.4 Redução de ruídos e detecção de *outliers*

Um ruído é uma variável que apresenta erro, seja por causa de problemas na coleta, na entrada dos dados, na transmissão dos dados, ou por falhas devido a limitações tecnológicas (CAMARGO, 2010).

Outliers podem ocorrer de duas formas: a primeira (*outliers* inválidos) ocorre quando o valor da variável está fora do intervalo esperado. Com isto deve-se tentar descobrir manualmente o valor correto da variável. A segunda forma (*outliers* válidos) ocorre quando o valor da variável está dentro da faixa do intervalo, mas não segue a tendência dos valores para o atributo. Apesar de poder ter um impacto negativo na execução dos algoritmos, esses valores não devem ser descartados, pois são características do conjunto de dados em questão. (CAMARGO, 2010).

2.3.1.5 Transformação dos Dados

Técnicas comuns de transformação de dados incluem: normalização, mapeamento de valores, discretização e agregação (MYATT, 2006). Estas são descritas a seguir:

- Normalização: É o processo de converter atributos numéricos para um novo intervalo de valores, utilizando de funções matemáticas. As técnicas mais populares de normalização são (SYMEONIDIS; MITKAS, 2005):
    - Normalização *min-max*: É aplicada uma transformação linear nos dados. Considerando *v* como sendo um valor entre o intervalo [min_A,max_A], sendo min_A e max_A os valores mínimo e máximo possíveis para o atributo, a normalização *min-max* irá mapear *v* para *v'* que estará contido dentro de um novo intervalo [$novoMin_A$,$novoMax_A$], através da seguinte equação:

$$v' = \frac{v - min_A}{max_A - min_A}(novoMax_A - novoMin_A) + novoMin_A \quad (1)$$

- Normalização *z-score*: É aplicado para normalizar o atributo através da sua média ($media_A$) e desvio padrão ($stddev_A$), por meio da seguinte equação:

$$v' = \frac{v - media_A}{stddev_A} \quad (2)$$

- Normalização em escala decimal: O valor do atributo é normalizado através do deslocamento da vírgula de sua parte decimal:

$$v' = \frac{v}{10^j} \quad (3)$$

onde *j* é o menor inteiro que satisfaz $\max(|v'|) < 1$.

- Mapeamento de valores: Ocorre quando se deseja analisar atributos ordinais, quando o algoritmo trabalha apenas com valores numéricos. As instâncias do atributo então são convertidas para um equivalente numérico. Exemplo: *pequeno* é substituído por *0*, *médio* é substituído por *1* e *grande* é substituído por *2*.
- Discretização: Em algumas situações, executar algoritmos de mineração de dados em um intervalo de valores pode não ser tão eficiente quanto executar em uma versão discretizada do intervalo. Um exemplo para a discretização é converter um atributo numérico *idade* nas categorias *criança*, *jovem*, *adulto* e *idoso*.
- Agregação: Ocorre quando o atributo desejado na análise não está presente no conjunto de dados, mas pode ser gerado através de um ou mais atributos presentes. Um exemplo está em obter o atributo idade através do atributo relativo à data de nascimento.

A transformação de certos atributos pode ser necessária, visto que muitos algoritmos de mineração de dados terão dificuldades em interpretar os dados no formato original (MYATT, 2006).

2.3.1.6 Redução de Dimensionalidade

A redução de dimensionalidade consiste na utilização de técnicas para identificação e eliminação de atributos redundantes ou de baixa relevância para o problema. Tais técnicas envolvem a aplicação de algoritmos específicos[3], e são divididas em duas categorias (CUNNINGHAM, 2007):

- Transformação de Características: Consiste em técnicas para transformar os atributos do conjunto de dados em novos atributos com um número mais compacto de

[3] Não abordados neste trabalho.

dimensões, mas mantendo o máximo possível de informação dos atributos originais. A transformação de características pode ser dividida em duas categorias:

- Extração de Características: Consiste na realização de um mapeamento dos atributos do conjunto de dados original em um novo conjunto de atributos equivalentes.
- Geração de Características: Consiste na descoberta de informações faltantes nos atributos do conjunto de dados original, e na construção de novos atributos que sintetizam essas novas informações para o conjunto de dados.

- Seleção de Características: Consiste na localização do melhor subconjunto de atributos do conjunto para representar o conjunto de dados original.

O processo de transformação de características pode resultar em atributos sem significado físico para o especialista no domínio. A seleção de características, entretanto, por manter os atributos originais mais relevantes para o conjunto de dados, irá manter a capacidade de interpretação do problema (CUNNINGHAM, 2007).

2.3.2 Tarefas de Mineração de Dados

Uma vez que a etapa de pré-processamento foi executada e o conjunto de dados está organizado de uma forma considerada aceitável pelo especialista do domínio, a etapa de Mineração de Dados é executada, onde um ou mais algoritmos serão executados. Esses algoritmos são classificados quanto às tarefas que eles realizam. Tais tarefas são usualmente divididas em quatro categorias: Classificação, Regressão, Regras de Associação e Agrupamento por similaridade (FAYYAD; SHAPIRO; SMYTH, 1996). Estas categorias são descritas a seguir.

2.3.2.1 Classificação

A tarefa de classificação consiste na construção de uma função ou modelo que determina a classe de um objeto baseado em seus atributos (SUMATHI; SIVANANDAM, 2006).

Essa função será construída por meio do mapeamento dos atributos em um conjunto de classes previamente definidas pelo especialista no domínio. Para cada amostra do conjunto que será construído o modelo, supõe-se que a sua classe seja conhecida. (CAMARGO, 2010) Tal conjunto é denominado conjunto de treinamento. Após tal função ser inferida, o modelo é

utilizado para determinar a classe para um conjunto de amostras que ainda não possuem tais classes definidas. Este conjunto é denominado conjunto de testes.

Tabela 1 – Exemplo de conjunto de dados utilizado em classificação

Atributo 1	**Atributo 2**	…	**Atributo *n***	**Classe**
18	30	…	28	Sim
21	14	…	30	Não
25	26	…	4	Sim
…	…	…	…	…
4	6	…	25	Não

Fonte: o autor

A classificação utiliza o processo de aprendizado supervisionado, ou seja, através de um conjunto de dados de treinamento que será classificado em possíveis grupos, será inferida uma função pra classificar os dados restantes do conjunto. O algoritmo de classificação tem por característica o modelo resultar em uma saída discreta.

Algoritmos de classificação são utilizados quando se deseja (SYMEONIDIS; MITKAS, 2005):

- Aplicar um esquema de segmentação de forma a identificar grupos mais importantes.
- Identificar possíveis relações entre as variáveis de tal forma a levar um entendimento de como uma variável pode afetar outra.
- Gerar uma representação visual da forma em que as variáveis se relacionam, como por exemplo, numa estrutura de árvore.
- Simplificar os atributos e categorias para manter as características essenciais.
- Identificar variáveis importantes em um conjunto de dados.

Algoritmos populares de classificação incluem: ID3, C4.5, Máquinas de Vetores Suporte (SVM), K Vizinhos mais Próximos (kNN), CART e Método de Bayes Ingênuo (WU et al., 2007).

2.3.2.2 Regressão

A tarefa de regressão é similar à classificação, exceto pelo fato de que a regressão trabalha com valores contínuos, permitindo assim que seja encontrada uma função que modele os dados com o menor erro possível (CAMARGO, 2010).

2.3.2.3 Regras de Associação

Associação é uma tarefa de mineração de dados utilizada para descobrir a probabilidade da co-ocorrência de itens em uma coleção. A relação entre itens que estão co-ocorrendo são expressas como regras de associação (ORACLE, 2012).

O primeiro problema proposto por Agrawal, Imielinski e Swami (1993) a ser solucionado pelas regras de associação envolve a associação entre a compra de produtos em um supermercado. Para este problema, deseja-se descobrir quais produtos um cliente poderá comprar em conjunto com frequência. Uma regra na forma {leite, pão} → {manteiga} indica que clientes que compram leite e pão, também irão comprar manteiga.

Seguindo a definição de Agrawal, Imielinski e Swami (1993), o problema da mineração de regras de associação é dado como: Seja $I = \{i_1, i_2, ... , i_n\}$ um conjunto de *n* atributos chamados *itens.* Seja $D = \{t_1, t_2, ... , t_n\}$ um conjunto de transações chamado *base de dados*. Cada transação em D tem uma ID única de transação, e contém um subconjunto de itens em I. Uma regra é definida como uma implicação na forma $X \rightarrow Y$ onde $X,Y \subseteq I$ e $X \cap Y = \emptyset$. O conjunto de itens X é chamado de *antecedente* da regra e o conjunto de itens Y é chamado de *consequente* da regra.

Agrawal, Imielinski e Swami (1993) ainda definem os seguintes conceitos:

- Suporte transacional mínimo *s* – a união dos itens que apresentam o *antecedente* e o *consequente* da regra está presente em um mínimo de *s*% de transações no banco de dados.
- Confiança mínima *c* – pelo menos *c*% das transações na base de dados que satisfazem o *antecedente* da regra também satisfazem o *consequente* da regra.

2.3.2.4 Agrupamento por similaridade (clusterização)

Ao contrário do processo de classificação onde as classes são previamente conhecidas, o processo de agrupamento por similaridade, também conhecido por clusterização, irá dividir as instâncias no número de classes desejadas, também conhecidas como *clusters*. As instâncias então são classificadas de tal forma que as similaridades inter-classes sejam minimizadas, e as similaridades intra-classe maximizadas (SUMATHI; SIVANANDAM, 2006). O processo de aprendizado para a clusterização é o não-supervisionado, ou seja, não há um processo de treinamento para a classificação dos objetos. Segundo Abernethy (2010), a

clusterização pode ser o método de mineração de dados de maior utilidade a ser utilizado, visto que ao separar o conjunto de dados em grupos, pode-se rapidamente obter algumas conclusões. O processo de agrupamento por similaridade é utilizado quando (SYMEONIDIS; MITKAS, 2005):

- O conhecimento do domínio é mínimo e deseja-se compreender melhor os parâmetros de entrada.
- Uma grande quantidade de dados existe, possuindo um alto grau de estruturas lógicas, assim como um grande número de variáveis a serem analisadas.

Algoritmos tradicionais para a aplicação da clusterização são: *k-means* e *EM* (WU et al., 2007).

Um algoritmo de clusterização tradicional segue a seguinte estrutura (ABERNETHY, 2010):

1. Normalização de todos os atributos do conjunto de dados, para a faixa de intervalo de 0 a 1.
2. Dado o número de *clusters* desejados, obter aleatoriamente o número de amostras do conjunto de dados para ser o centro inicial de cada *cluster*.
3. Calcular a distância entre cada amostra do conjunto de dados e o centro de cada *cluster*, utilizando o método dos mínimos quadrados.
4. Atribuir cada amostra a um *cluster*, baseado na distância mínima do centro de cada *cluster*.
5. Calcular o centróide, que consiste na média de cada atributo usando apenas os membros de cada *cluster*.
6. Calcular a distância entre cada nova amostra e os centróides criados. Se o centróide do *cluster* for alterado, a iteração será refeita desde o terceiro passo do algoritmo.

2.3.3 Algoritmos para mineração de dados

Para realizar as tarefas descritas anteriormente, vários algoritmos podem ser utilizados. Entretanto, algoritmos diferentes aplicados para o mesmo conjunto de dados criam formas diferentes de visualizar os resultados para o problema em questão (FAYYAD; SHAPIRO; SMYTH, 1996). A interpretação dos resultados para diferentes domínios de problemas podem ser favorecidos por determinados algoritmos e prejudicados em outros. A seguir analisamos os tipos de algoritmos utilizados para suportar tais tarefas e suas principais aplicações:

2.3.3.1 Árvores de Decisão

Árvores de decisão são modelos utilizados em classificação para representar visualmente uma relação entre os atributos de forma que as decisões serão realizadas ao percorrer os diversos nós da árvore (WITTEN; HALL, 2011). Árvores de decisão são atrativas devido à facilidade de visualização e da interpretação dos seus modelos resultantes (SUMATHI; SIVANANDAM, 2006).

Árvores de decisão podem ser utilizadas para classificação de dados categóricos (árvores de classificação) como também para quando a saída consiste em números reais (árvores de regressão).

No trabalho proposto por Quinlan (1986) pode ser visto um exemplo de conjunto de dados em que pode ser induzida uma árvore de decisão para verificação da viabilidade de um determinado jogo. Através dos jogos realizados foram registrados os atributos relativos às características do ambiente. A partir dos resultados, uma classe foi atribuída pelo especialista do domínio para cada instância, indicando se o jogador deve ou não jogar naquela situação. O conjunto de dados pode ser visto na tabela 2, e uma possível árvore de decisão para o conjunto pode ser vista na figura 3.

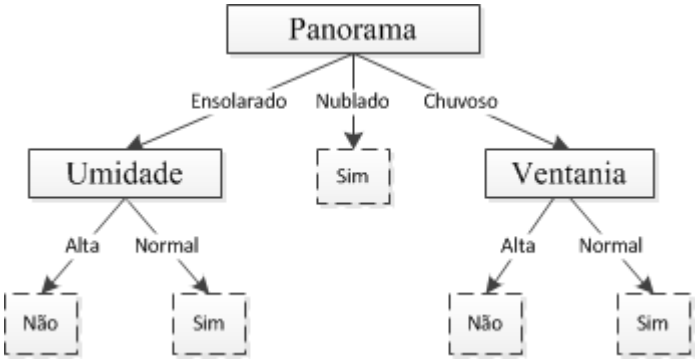
Tabela 2 – Exemplo de conjunto de dados para a indução da árvore de decisão

Panorama	**Temperatura**	**Umidade**	**Ventania**	**Jogar?**
Ensolarado	Quente	Alta	Não	Não
Ensolarado	Quente	Alta	Sim	Não
Nublado	Quente	Alta	Não	Sim
Chuvoso	Moderado	Alta	Não	Sim
Chuvoso	Frio	Normal	Não	Sim
Chuvoso	Frio	Normal	Sim	Não
Nublado	Frio	Normal	Sim	Sim
Ensolarado	Moderado	Alta	Não	Não
Ensolarado	Frio	Normal	Não	Sim
Chuvoso	Moderado	Normal	Não	Sim
Ensolarado	Moderado	Normal	Sim	Sim
Nublado	Moderado	Alta	Sim	Sim
Nublado	Quente	Normal	Não	Sim

Chuvoso	Moderado	Alta	Sim	Não

Fonte: (QUINLAN, 1986)

Figura 3 – Árvore de decisão para o exemplo



Fonte: (QUINLAN, 1986)

O procedimento de criação de árvores de decisão consiste na definição de um nó raiz e recursivamente criar nó folhas através de um determinado critério para a divisão dos dados. A seguir é executado um processo de *poda*, que consiste em diminuir a altura da árvore resultante para diminuir sua complexidade.

O algoritmo *ID3* (QUINLAN, 1986) foi um dos primeiros algoritmos criados para árvores de decisão, posteriormente aprimorado para o algoritmo *C4.5* (GARCIA; ALVARES, 2000). O funcionamento do algoritmo *C4.5*, utilizando um conjunto de treinamento *S*, é descrito a seguir (WU et al., 2007):

1. Se todas as instâncias em S pertencem a mesma classe ou *S* é pequeno, a árvore é um nó folha marcado com a classe mais frequente em *S*.
2. Caso contrário, escolher uma instância teste baseado em um único atributo com dois ou mais possíveis resultados. Marcar esta instância teste como raiz da árvore, com uma ramificação para cada possibilidade da instância. Particionar o conjunto *S* em subconjuntos $S_1, \ldots, S_n$ de acordo com a possibilidade para cada caso, e aplicar o procedimento recursivamente para cada subconjunto.

Ao contrário do algoritmo *ID3*, onde atributos categóricos necessitam ser discretizados, o algoritmo *C4.5*, pode ser utilizado com atributos categóricos e numéricos. Para atributos categóricos, cada categoria terá uma ramificação na árvore de decisão. Para atributos numéricos, a árvore será dividida em duas ramificações, que melhor dividirão o atributo.

Uma árvore de decisão necessita de um critério que irá definir o processo da sua

divisão em ramificações (*splitting*). Um dos critérios mais populares, e utilizado no algoritmo C4.5 é o ganho de informação, que mede o quanto é informativo um atributo (GARCIA; ALVARES, 2000). Outro critério conhecido é o critério de *Gini* (Wu et al., 2007), utilizado, por exemplo, no algoritmo CART.

De forma geral, o algoritmo CART difere-se do C4.5 em alguns aspectos, como por exemplo o número de ramificações (CART só gera modelos binários), método da poda, e formas de tratar instâncias com valores desconhecidos. (Wu et al., 2007).

Para o caso geral, o algoritmo para construção de árvores de decisão segue a seguinte estrutura (KOTSIANTIS, 2007):

1. Verificar os casos base.
2. Para cada atributo *a*
    1. Encontrar o ganho da informação normalizada da divisão em *a*.
3. Escolher o a_{best} para ser o atributo com o ganho da informação normalizada mais alta.
4. Criar um nó de decisão no_x que divida a_{best}.
5. Repetir o procedimento nas sublistas dividindo a_{best}, e adicionando os nós como filhos de no_x.

2.3.3.2 K vizinhos mais próximos

K vizinhos mais próximos (ou kNN – *k-Nearest Neighbor*) consiste na classificação de registros utilizando critérios de similaridade. A estratégia consiste em achar *k* registros no conjunto de treinamento que mais se aproximam ao registro do conjunto de teste, e irá classificar o registro com a classe predominante nos *k* vizinhos (WU et al., 2007). Considerando o conjunto de treinamento *T* e o conjunto de teste *C*, o algoritmo geral do kNN é descrito a seguir (KOTSIANTIS, 2007):

1. Para cada registro de teste em *C*:
    1. Encontre os *k* registros em *T* de acordo com uma métrica de distância.
    2. Classe resultante = classe mais frequente nos *k* registros mais próximos.

Diversas métricas podem ser utilizadas para o cálculo da distância. Tais métricas podem ser vistas em (KOTSIANTIS, 2007).

2.3.3.3 Classificação Bayesiana

Redes Bayesianas pertencem a uma ramificação dos algoritmos de aprendizado supervisionados chamada algoritmos de aprendizado estatístico, que consiste na construção de um modelo probabilístico de cada instância para cada classe (KOTSIANTIS, 2007). Na classificação Bayesiana o modelo é um grafo orientado construído através do teorema de probabilidades de Bayes. O modelo gerado representa a relação causal entre os atributos (SUMATHI; SIVANANDAM, 2006).

Redes Bayesianas, entretanto, também podem ser utilizadas para gerar árvores de decisão. Um algoritmo utilizado é o *NBTree* (KOHAVI, 1996).

2.3.3.4 Máquina de Vetores Suporte

Segundo Kotsiantis (2007), Máquinas de Vetores Suporte (ou SVM – *Support Vector Machines*) é a técnica de aprendizado supervisionado mais nova existente. Uma SVM consiste em um sistema de classificação binário que, dado duas possíveis classes, estas serão separadas através do conceito de *margem*, que são os lados de um *hiper-plano* que dividem as duas classes. Se uma separação linear dessas duas classes for possível, um *hiper-plano* ótimo que divide essas classes pode ser encontrado (KOTSIANTIS, 2007). Uma abordagem detalhada de SVMs vai além do escopo deste trabalho, e pode ser vista em (BURGES, 1998).

2.3.3.5 Regras de Associação

Um algoritmo frequentemente utilizado em regras de associação é o *Apriori* (AGRAWAL; IMIELIŃSKI; SWAMI, 1993).

O principio de funcionamento do *Apriori* consiste nas seguintes etapas (GOLDSCHMIDT; PASSOS, 2005):

1. Encontrar todos os itens frequentes (que satisfaçam a condição de suporte mínimo).
2. A partir do conjunto de itens resultantes das iterações da etapa (1), serão geradas as regras de associação que satisfazem a condição de confiança mínima.

Na tabela 3 é demonstrado um conjunto de dados exemplo para a aplicação do algoritmo *Apriori*. Cada item do conjunto pode ser visto, por exemplo, como um produto adquirido em um supermercado. Com a aplicação do algoritmo, deseja-se encontrar quais produtos que foram comprados em conjunto com maior frequência. O exemplo foi extraído de Wikipedia (2012b):

Tabela 3 – Conjunto de dados exemplo para o experimento de regras de associação

Transação	**Item 1**	**Item 2**	**Item 3**	**Item 4**
1	Sim	Sim	Sim	Sim
2	Sim	Sim	Não	Não
3	Não	Sim	Sim	Sim
4	Não	Sim	Sim	Não
5	Sim	Sim	Não	Sim
6	Não	Não	Sim	Sim
7	Não	Sim	Não	Sim

Fonte: (WIKIPEDIA, 2012b)

O *Apriori* primeiramente irá encontrar a frequência de todos os itens no conjunto de dados, ou seja, o suporte para cada item, e são mostrados na tabela 4. Para este exemplo, vamos delimitar que o suporte mínimo desejado é 3/7.

Tabela 4 – Frequência dos itens no conjunto de dados

Item	**Suporte**
1	3/7
2	6/7
3	4/7
4	5/7

Fonte: (WIKIPEDIA, 2012b)

Com isto, serão verificadas as possíveis associações, dado um suporte mínimo. Se a associação tiver um suporte menor que o suporte mínimo, ela não será incluída. Todos os itens da tabela 4 atingiram o suporte mínimo desejado, logo eles passam para a próxima etapa do algoritmo. Na tabela 5 são geradas todas as associações possíveis para 2 itens.

Tabela 5 – Associação para 2 itens

Item	**Suporte**
{1,2}	3/7
{1,3}	1/7

{1,4}	2/7
{2,3}	3/7
{2,4}	4/7
{3,4}	3/7

Fonte: (WIKIPEDIA, 2012b)

Para as associações encontradas, somente os pares {1,2}, {2,3}, {2,4} e {3,4} satisfazem a condição de suporte mínimo. Para encontrar a associação de 3 itens, as demais associações serão desconsideradas. Na tabela 6 é mostrada a associação para 3 itens encontrada.

Tabela 6 – Associação para 3 itens

Item	**Suporte**
{2,3,4}	2/7

Fonte: (WIKIPEDIA, 2012b)

Para este exemplo, a associação {2,3,4} não satisfaz a condição de suporte mínima, logo ela é descartada.

2.3.3.6 Redes Neurais

Devido a dificuldade de computadores resolverem problemas não-algorítmicos, surgiu a necessidade de criação de um modelo que se inspirasse no funcionamento do cérebro humano, devido a sua natural capacidade de resolver tais problemas (CAMARGO, 2010). Desta forma, foram criadas as técnicas hoje conhecidas como Redes Neurais.

Segundo Myatt (2006), uma rede neural consiste em um modelo matemático que faz predições baseado em uma série de variáveis de entrada.

Uma rede neural consistirá de um conjunto de nós da camada de entrada, um conjunto de nós da camada de saída e, entre eles, um conjunto de nós da camada oculta. Cada nó da rede é conectado à todos os nós da camada adjacente. Por fim, cada conexão tem um peso associado a ele. Antes do processo de aprendizado, pesos entre a faixa de valores –1 e +1 são atribuídos às conexões aleatoriamente, sendo ajustados no processo de aprendizado (MYATT, 2006).

O modelo descrito acima, conhecido como *Feedforward neural network* é a abordagem clássica para uma rede neural. Entretanto, vários modelos para redes neurais existem, como, por exemplo. *Radial Basis Function networks* (BISHOP, 1995), *Self-Organizing Maps* (KOHONEN, 1982), *Learning Vector Quantization* (SOMERVUO; KOHONEN, 1999), entre outros.

Uma dificuldade no uso das redes neurais está na extração de regras da rede, visto que consistem em um modelo de caixa-preta e, portanto, com pouca possibilidade de interpretação dos seus resultados (CAMARGO, 2010). Entretanto, técnicas foram propostas para permitir a descoberta do conhecimento em redes neurais (TICKLE et al., 1998), embora nenhuma dessas técnicas seja amplamente utilizada (CAMARGO, 2010).

2.3.4 Avaliação da mineração de dados

Uma vez que o algoritmo de mineração de dados foi aplicado em uma amostra e o modelo foi gerado, tal modelo deve ser avaliado de forma a verificar a sua qualidade para a aplicação no restante dos dados que não foram avaliados pelo algoritmo. As técnicas mais comuns são descritas a seguir:

2.3.4.1 Particionamento dos dados

Os dados geralmente são separados em dois conjuntos. Um para o treinamento e o outro para avaliação do modelo. Duas técnicas comuns são utilizadas (CAMARGO, 2010):

- Holdout: A técnica de *holdout* é aplicada quando há uma grande quantidade de dados. Consiste em dividir os dados aleatoriamente em dois conjuntos de forma a não se sobreporem. O tamanho desses conjuntos são geralmente 75% para o treinamento do modelo e 25% para testes. Entretanto, existe outra técnica, como a de amostragem aleatória, onde as partições são feitas aleatoriamente e o procedimento repetido um número de vezes e retirado a média das repetições.
- Validação Cruzada: Técnicas de validação cruzada são utilizadas quando há uma quantidade limitada de dados. As técnicas mais populares são *n-fold* e *leave-one-out*. Na primeira o conjunto de dados é particionado em *n* partições de tamanhos iguais e o procedimento repetido *n* número de vezes. Em cada iteração uma delas é utilizada para teste e as outras para o treinamento do modelo. Na técnica *leave-one-out* é similar, com a diferença que a exatidão do modelo é obtida através da medição da exatidão de

cada amostra e depois obtida a média das exatidões (CAMARGO, 2010).

2.3.4.2 Matriz de Confusão

A matriz de confusão consiste em uma matriz quadrada onde os atributos do problema são expressos nas linhas e os atributos da predição são expressos nas colunas. Cada entrada na matriz corresponde a um número inteiro de instâncias que foram classificadas corretamente ou incorretamente pelo método (BRAMER, 2007).

Uma instância classificada corretamente será marcada na entrada correspondente à linha do atributo do problema e à coluna do atributo para o modelo predito. A matriz de confusão apresenta os atributos na mesma ordem das linhas e das colunas, logo os valores corretamente preditos são armazenados na diagonal principal da matriz (BRAMER, 2007).

2.3.4.3 Medição de desempenho dos modelos

Várias técnicas existem para medir o desempenho dos modelos. A seguir são descritas algumas das mais utilizadas:

Verdadeiros e falsos positivos e negativos

Quando existem apenas duas classes, podemos denominar elas como positiva (+) e negativa (–). A matriz de confusão para este conjunto de classes irá consistir de 4 entradas, que iremos denominar *TP*, *FP*, *TN* e *FN*, demonstrada na tabela 7 (BRAMER, 2007).

Tabela 7 – Verdadeiros e falsos positivos e negativos

		Classe da Predição	
		+	-
Classe Real	+	TP	FN
	-	FP	TN

Fonte: (BRAMER, 2007)

- TP (*True Positive*): Número de instâncias positivas que são classificadas como positivas.

- TN (*True Negative*): Número de instâncias negativas que são classificadas como negativas.
- FP (*False Positive*): Número de instâncias positivas que são classificadas como negativas.
- FN (*False Negative*): Número de instâncias negativas que são classificadas como positivas.

Os termos “Falsos Positivos” e “Falsos Negativos” também são utilizados na literatura para descrever os dois tipos de erros de classificação (BRAMER, 2007):

- Falsos Positivos (Erros Tipo 1): ocorre quando as instâncias que deveriam ser classificadas como negativas são classificadas como positivas.
- Falsos Negativos (Erros Tipo 2): ocorre quando as instâncias que deveriam ser classificadas como positivas são classificadas como negativas.

Exatidão

A exatidão do modelo pode ser calculada através do número de observações classificadas corretamente dividido pelo número total de observações (MYATT; JOHNSON, 2009), ou seja:

$$\text{exatidão} = \frac{\text{TP}+\text{TN}}{\text{TP}+\text{FP}+\text{TN}+\text{FN}} \quad (4)$$

Sensibilidade

A sensibilidade consiste no número de observações positivas classificadas como positivas, ou seja, a taxa de verdadeiros positivos. É calculada através do número de observações positivas classificadas corretamente divididas pelo número total de observações positivas (MYATT; JOHNSON, 2009), ou seja:

$$\text{sensibilidade} = \frac{\text{TP}}{\text{TP}+\text{FN}} \quad (5)$$

Especificidade

A especificidade consiste no número de observações negativas classificadas como negativas, ou seja, aa taxa de verdadeiros negativos. É calculada através do número de observações negativas classificadas corretamente dividido pelo número total de observações

negativas (MYATT; JOHNSON, 2009), ou seja:

$$\text{especificidade} = \frac{\text{TN}}{\text{TN+FP}} \quad (6)$$

Precisão

A precisão é o número de verdadeiros positivos dividido pelo número total de observações positivas, ou seja:

$$\text{precisão} = \frac{\text{TP}}{\text{TP+FP}} \quad (7)$$

Estatística *Kappa*

A estatística *Kappa* é uma medida de concordância entre as categorias preditas e observadas em um conjunto de dados, corrigida para uma concordância que ocorre por chance (WITTEN; HALL, 2011).

A equação para *Kappa* é dada a seguir (LANDIS; KOCH, 1977):

$$\kappa = \frac{\text{Pr(a)} - \text{Pr(e)}}{1 - \text{Pr(a)}} \quad (8)$$

onde *Pr(a)* é a concordância relativa observada e *Pr(e)* é a probabilidade de concordância ocorrida por chance.

Para melhor entendimento da estatística *Kappa*, vejamos o exemplo da tabela 8, para o repasse de dinheiro para um determinado conjunto de pessoas. Neste caso, dois avaliadores (A e B) decidem (Sim ou Não) se a verba será repassada para cada instância. O exemplo é retirado de Wikipedia (2012c):

Tabela 8 – Matriz de confusão para o exemplo

		B Sim	**B** Não
A	Sim	20	5
A	Não	10	15

Fonte: (WIKIPEDIA, 2012c)

Pr(a) consiste na porcentagem de instâncias que houve concordância. Para este caso Pr(a) = (20 + 15)/50 = 0.70.

Para o cálculo de Pr(e), é preciso notar que:

- O avaliador A disse “Sim” para 25 pessoas e “Não” para 25 pessoas. Logo, ele disse “Sim” 50% do tempo.
- O avaliador B disse “Sim” para 30 pessoas e “Não” para 20 pessoas. Logo, ele disse “Sim” 60% do tempo.

A probabilidade de ambos dizerem “Sim” aleatoriamente é de 0.50 * 0.60 = 0.30, e a probabilidade de ambos dizerem “Não” é de 0.50 * 0.40 = 0.20. Logo, a probabilidade de concordância ocorrida por chance é Pr(e) = 0.3 + 0.2 = 0.5.

Aplicando a equação (8), temos que $\kappa = 0.4$.

Na tabela 9, é exibida a relevância de *Kappa* para os modelos gerados, proposto por propostos por Landis e Koch (1977), para diversas faixas de valores.

Tabela 9 – Interpretação para valores de Kappa.

Estatística Kappa	**Interpretação**
< 0	Sem concordância
0 a 0,20	Leve
0,21 a 0,40	Considerável
0,41 a 0,60	Moderada
0,61 a 0,80	Substancial
0,81 a 1,00	Excelente

Fonte: (LANDIS; KOCH, 1977)

2.3.5 Pós-Processamento

Esta etapa consiste na inspeção e na tradução do conhecimento obtido para uma forma fácil de ser compreendida aos interessados nos problema em questão. Com isto, este conhecimento será analisado de tal forma que possa ser utilizado futuramente na prática.

A forma que os resultados serão interpretados pode ser fortemente influenciada pelo algoritmo em questão. Modelos de caixa-preta, como por exemplo, redes neurais, frequentemente podem ser difíceis de serem interpretados, ao contrário dos modelos de caixa-branca, como, por exemplo, árvores de decisão, onde a forma que os dados se relacionam podem ser facilmente verificados (HAND; SMYTH; MANNILA, 2001).

De forma adicional, os resultados podem ser avaliados e validados com a ajuda de um profissional da área, quando o modelo em questão envolver domínios críticos ou de difícil

interpretação como, por exemplo, na área da saúde ou na indústria aeroespacial.

Outra questão a ser levantada está na consistência do modelo para a predição de dados no futuro. Um modelo pode ser coerente até um determinado momento e perder sua relevância após certo tempo, devido à adição de novas variáveis causadas por eventos externos. Um modelo econômico, por exemplo, pode ser fortemente influenciado pela ocorrência de algum desastre natural.

3 MATERIAIS E MÉTODOS

Neste capítulo são descritos as ferramentas de *hardware* e *software* utilizadas para o desenvolvimento do trabalho, assim como a metodologia utilizada.

3.1 Materiais

Para a aplicação dos algoritmos de mineração de dados será utilizada a ferramenta *Weka* (HALL et al., 2009). *Weka* (*Waikato Environment for Knowledge Analysis*) é um *software* que contém uma coleção de algoritmos e ferramentas para a modelagem preditiva e para a análise dos dados. Os algoritmos abrangidos envolvem as tarefas de classificação, regressão, clusterização, regras de associação assim como ferramentas para importar o conjunto de dados e para selecionar os atributos a serem utilizados nas análises. Também possui uma interface gráfica para o suporte de tais tarefas e para a visualização de gráficos de dispersão. *Weka* é desenvolvido em *Java* e licenciado sob a *GNU General Public License*, o que significa que ele é multiplataforma e permite que seu código fonte seja estudado e alterado.

O ambiente utilizado para a análise dos dados contém a seguinte configuração de hardware:

- Processador Intel Core I5-2520M (Sandy Bridge, 2.5GHz, 3MB L3 Cache)
- 4 GB de RAM DDR3 1333 MHz
- Disco Rígido de 250GB 7200 RPM

A configuração de software é:

- Microsoft Windows 7 Professional
- Microsoft Excel 2010
- Microsoft Visual C++ 2010 Express
- Java SE Runtime Environment (Java 7 Update 9)
- Weka 3.6.8

A configuração mínima para a execução dos algoritmos pode variar dependendo do tamanho do conjunto de dados a ser utilizado e dos algoritmos utilizados. Na execução dos experimentos foi possível observar que a configuração utilizada é suficiente para os algoritmos utilizados no conjunto de dados em questão.

3.2 Métodos

Os conjuntos de dados foram obtidos na Secretaria Acadêmica do Campus Bagé da Universidade Federal do Pampa através de dois sistemas. O primeiro é o SIE (Sistema de Informações para Ensino), sistema onde estão concentrados os dados relativos à vida acadêmica de todos os alunos da universidade. O segundo é o portal do SISU (Sistema de Seleção Unificada), que contém dados relativos aos ingressantes pelo ENEM, como a classificação geral dos ingressantes, assim como as notas obtidas no ENEM, separados por área.

Os dados foram repassados no formato *XLS*, que é o formato padrão utilizado no *Microsoft Excel*, e são posteriormente convertidos para o formato *CSV* (*Comma-Separated Values* ou em português: Valores Separados por Vírgula), que consiste num arquivo em formato texto onde os registros são separados por vírgulas, por sua facilidade de manipulação através de linguagens de programação. O formato *CSV* é aceito como arquivo de entrada no *Weka*, que internamente converte para o formato *ARFF*, que é o formato utilizado pelo *Weka*.

Os dados utilizados foram somente do curso de Engenharia de Computação, a fim de limitar o escopo do trabalho e o tempo de mineração dos dados. Os nomes dos alunos foram omitidos da análise por questões de confidencialidade. Entretanto, para fins de manter um atributo capaz de ser utilizado para a integração dos dados com os conjuntos de dados obtidos posteriormente, um atributo numérico único para cada aluno que é comum entre os conjuntos de dados é mantido. Para este caso foi utilizado o número da matrícula do aluno.

Além das técnicas de programação utilizadas para o pré-processamento dos conjuntos de dados, são utilizadas técnicas para a extração de informações nesses conjuntos de dados:

- Visualização – Através da visualização de gráficos de atributos dos alunos regulares e evadidos, deseja-se um melhor entendimento no domínio estudado, e com isto, permitir uma melhor escolha na preparação dos dados para a mineração, e com isto, realizar uma análise mais precisa dos dados.
- Classificação – Dado um conjunto de dados dos alunos regulares e evadidos, deseja-se obter regras que caracterizem o perfil do aluno com propensão a evadir. Para isto são aplicados diversos algoritmos de classificação, como por exemplo, árvores de decisão, a fim de identificar uma correlação nos dados dos alunos pré-existentes.
- Clusterização – Com a aplicação da clusterização, as instâncias serão separadas em um número determinado de grupos (*clusters*), para verificar as semelhanças dos atributos dos alunos contidos nesses grupos.

- Regras de Associação – Com a aplicação de técnicas de associação deseja-se verificar quais relações entre os atributos tem por consequência a evasão do aluno.

4 EXECUÇÃO DO TRABALHO

Neste capítulo é descrito a parte prática do trabalho, onde são executadas as etapas da Descoberta de Conhecimento em Banco de Dados. Também são descritos os resultados obtidos nos experimentos.

4.1 Dados disponíveis

Para a realização deste trabalho foram aplicados esforços para a obtenção dos dados dos alunos e seus respectivos desempenhos acadêmicos. Tal atividade demandou mais tempo que o esperado. Inicialmente não havia informações de quais atributos estavam de fato disponíveis no banco de dados da Universidade.

Até certo momento acreditava-se na existência de dados socioeconômicos dos alunos, que poderiam trazer uma quantidade maior de resultados para o trabalho. Também se acreditava na existência de conjunto de dados informando se o aluno foi bolsista e o tipo de bolsa fornecida ao aluno.

Outros conjuntos de dados foram verificados existentes, mas não foi possível a sua extração. Alguns exemplos verificados são: naturalidade do aluno (cidade e estado), nacionalidade, estado civil e etnia. Foi possível verificar a existência de tais atributos no Portal do Aluno da Universidade[4], mas não foi encontrada uma forma de extrair tais atributos pelo SIE[5].

Também houve dificuldade na obtenção dos dados, devido ao fato de que os dados disponíveis estão distribuídos entre diversos setores da universidade, como por exemplo, os dados do SISU, que tem um portal específico e não estão integrados ao SIE, além de serem mantidos por diferentes responsáveis. Além disso, compromissos de confidencialidade dificultam a obtenção de tais dados. No momento de execução deste trabalho, os seguintes conjuntos de dados estavam disponíveis:

- Relação completa de alunos: Contendo dados como sexo, data de nascimento, forma e período de ingresso, como também forma e período de evasão para os alunos evadidos.
- Relação de notas do ENEM: Contendo dados de desempenho acadêmico no ENEM por área, apenas para alunos ingressantes pelo SISU em 2011.

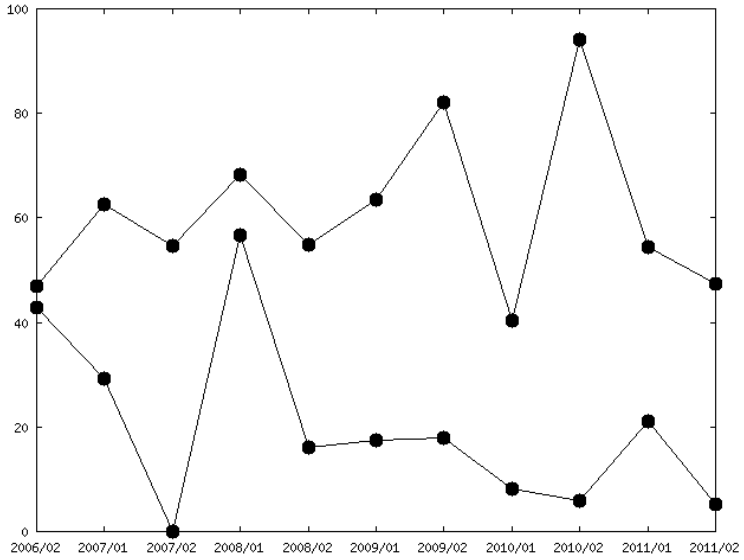
[4] Verificado pelo autor.
[5] Informado pela Secretaria Acadêmica.

- Local de residência: Contendo a cidade e CEP que o aluno informou para o cadastro do sistema acadêmico. Entretanto, este conjunto de dados não informa os dados da naturalidade do aluno.
- Desempenho acadêmico: Informa cada disciplina cursada para cada aluno, assim como a situação final, notas, trancamentos parciais e totais.
- Relação de disciplinas do curso: Contendo o código das disciplinas do curso de Engenharia de Computação, nome das disciplinas, Número de créditos, carga horária e o período que as disciplinas são oferecidas.

4.1.1 Descrição dos dados

Através da ferramenta desenvolvida para visualização de estatísticas das disciplinas por semestre (*AprDisciplinas*) foi possível gerar gráficos da porcentagem do total das reprovações nas disciplinas (soma das reprovações por nota, reprovações por frequência e trancamentos). Na figura 4 é exibido na parte superior o gráfico de reprovações totais por semestre e na parte inferior são exibidas as reprovações por nota para a disciplina de Algoritmos e Programação, também ofertada no primeiro semestre do curso de Engenharia de Computação. O período de maior número de desistências foi em 2010/2, onde a disciplina teve 94% de reprovações, sendo apenas 6% reprovações por nota, ou seja, houve 88% de desistências (reprovações por frequência e trancamentos).

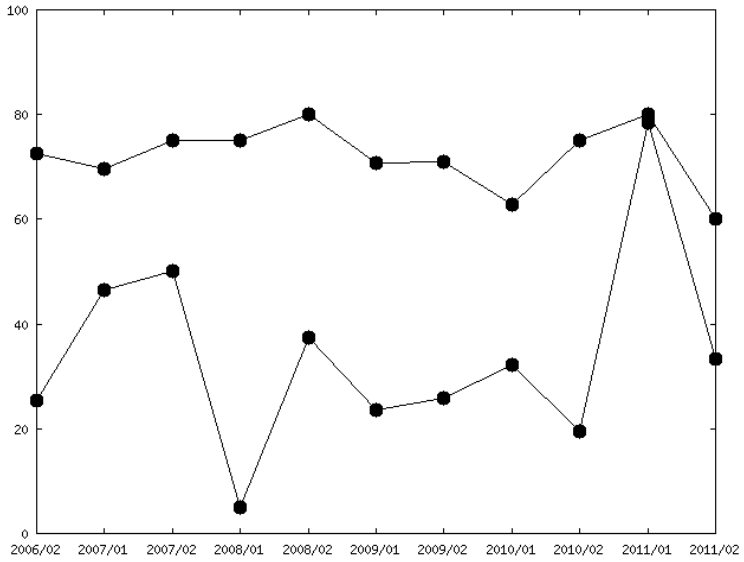
Figura 4 – Total de reprovações (acima) e reprovações por nota (abaixo) - Algoritmos e Programação



Fonte: o autor

Na figura 5 é exibido na parte superior o gráfico da porcentagem das reprovações totais por semestre e na parte inferior é exibido a porcentagem das reprovações por nota para a disciplina de Cálculo I, disciplina ofertada no primeiro semestre do curso de Engenharia de Computação. No semestre de 2010/2 houve a maior porcentagem de desistências para esta disciplina. De 75% de reprovações na disciplina, apenas 19% consistiram de reprovações por nota, ou seja, 56% dos alunos matriculados desistiram da disciplina.

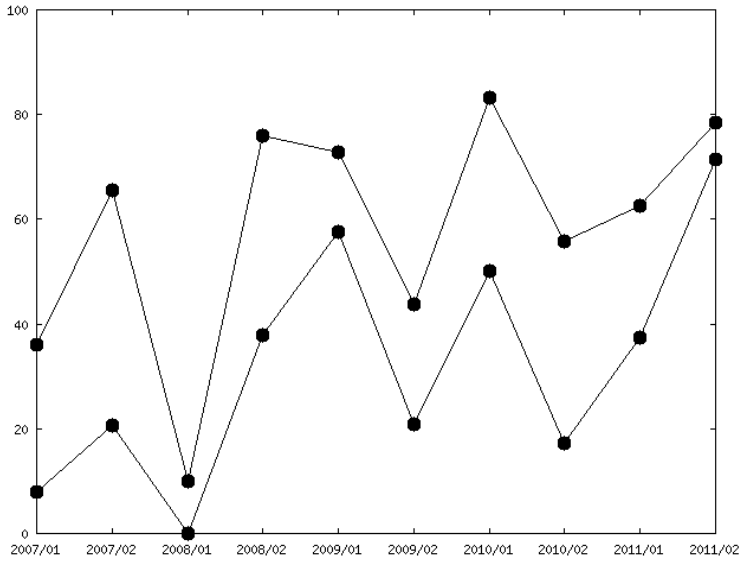
Figura 5 – Total de reprovações (acima) e reprovações por nota (abaixo) - Cálculo I



Fonte: o autor

Na figura 6 é exibido na parte superior o gráfico de reprovações totais por semestre e na parte inferior é exibido as reprovações por nota para a disciplina de Física I, disciplina oferecida no segundo semestre do curso de Engenharia de Computação. A maior porcentagem de desistências ocorreu em 2007/02, onde 65% dos alunos matriculados da Engenharia de Computação reprovaram, sendo 20% reprovações por nota, restando 45% de desistências.

Figura 6 – Total de reprovações (acima) e reprovações por nota (abaixo) - Física I



Fonte: o autor

Também foi gerada uma tabela para estimar a quantidade de evadidos por número de semestres que o aluno é regular na UNIPAMPA e está expresso na tabela 10, onde pode ser visto que a maior incidência de evadidos está nos primeiros semestres do curso. Através dos gráficos das figuras 4, 5, 6 e na tabela 10 é verificado que ainda há uma grande concentração de desistências no primeiro semestre. Sugere-se que sejam criados trabalhos para investigar a evasão no primeiro semestre consequente das desistências dos alunos, o que não é contemplado neste trabalho.

Tabela 10 – Semestres cursados e número de alunos evadidos

Semestres Cursados	**Número de Evadidos**
1	76
2	56
3	20
4	19
5	10
6	7
7	3
8	2
9	1
10	2

Fonte: o autor

Através da ferramenta *ConverteCidades* foi possível verificar para a amostra de 192 alunos a porcentagem de alunos evadidos oriundos de Bagé e de outras cidades, que pode ser visto na tabela 11. O atributo relativo ao estado do aluno, entretanto, não está disponível para verificar a taxa de evasão apenas das cidades mais distantes.

Tabela 11 – Porcentagem de alunos evadidos para a amostra

Cidade	**Alunos Regulares**	**Alunos Evadidos**	**Total de Alunos**	**% Evadidos**
Bagé	89	27	116	23.28%
Outras Cidades	50	26	76	34.21%

Fonte: o autor

A tabela 11 indica que alunos oriundos de outras cidades tiveram um maior percentual de evasão.

Na tabela 12 são demonstradas as taxas de evasão de cada turma que ingressou na UNIPAMPA. Pode ser notado que a taxa de evadidos atinge valores acima de 50%.

Tabela 12 – Taxas de evasão para as turmas que ingressaram na UNIPAMPA

Semestre	**Ingressos**	**Evadidos**	**% Evadidos**
200602	52	35	67.31
200701	31	18	58.06
200801	61	38	62.30
200802	8	5	62.50
200901	61	39	63.93
200902	6	4	66.67
201001	60	31	51.67
201002	4	1	25.00
201101	53	25	47.17

Fonte: o autor

4.2 Pré-processamento

A etapa do pré-processamento foi iniciada com a verificação manual dos atributos de cada conjunto de dados, para verificar a relevância de tais atributos para a realização do trabalho. Foi verificado que o conjunto de dados do local de residência dos alunos está incompleto, pois apresenta apenas 239 registros dos 407 alunos totais, sendo que 7 registros não informam a cidade e 43 registros eram endereços diferentes para alunos já cadastrados. Ou seja, só havia registro para 189 alunos. Além disso, devido ao fato de que alunos de outras localidades podem informar um endereço de Bagé, foi decidido que a utilização deste conjunto de dados não seria utilizado para os experimentos. Entretanto, a ferramenta *GeraAnálise* descrita posteriormente pode ser facilmente adaptada para integrar este conjunto de dados nas análises posteriores. Isto permite que tais dados sejam utilizados em trabalhos futuros, caso for observado uma melhoria na qualidade desses dados. Na tabela 13 são descritos os atributos do conjunto de dados "Relação Completa de Alunos".

Tabela 13 – Organização dos atributos do conjunto de dados "Relação Completa de Alunos"

Atributo	**Domínio**	**Descrição**
ID_PESSOA	Numérico	Número identificador do aluno
SEXO	Categórico	Sexo
DT_NASCIMENTO	Data	Data de Nascimento
FORMA_INGRESSO	Categórico	Forma de ingresso no curso
FORMA_EVASAO	Categórico	Forma de evasão do curso
COD_CURSO	Categórico	Código do curso
MATR_ALUNO	Numérico	Matrícula do aluno
NUM_VERSAO	Numérico	Ano da Versão do Software de Gestão
ANO_INGRESSO	Numérico	Ano do Ingresso
INGRESSO	Categórico	Semestre de Ingresso
EVASAO	Categórico	Semestre da Evasão
ANO_EVASAO	Numérico	Ano da Evasão
PERIODO_EVA_ITEM	Numérico	Semestre da evasão em formato numérico
CARACTER	Categórico	"/" para os alunos evadidos

Fonte: o autor

As categorias possíveis para o conjunto de dados da tabela 13 são:

- SEXO - M, F.
- FORMA_INGRESSO - ENEM - Exame Nacional do Ensino Médio, Portador de

Diploma, Processo Seletivo - Vestibular, Reingresso, Reopção - Mobilidade Interna (para curso/habilitação área relacionada), Transf. Interna Por Reopção de Curso, Transferência, Transferência Edital de Vagas, Transferência EX-OFFICIO, Transferência Interna, Transferência Voluntária ou Externa (oriundo de outra instituição).

- FORMA_EVASAO - Abandono, Aluno Regular, Cancelamento, Classificado e Não Matriculado, Desligamento, Formado, Transf. Interna Por Reopção de Curso, Transferência, Transferência Interna.
- COD_CURSO - BAEC[6].
- INGRESSO - 1. Semestre, 2. Semestre.
- EVASAO - 1. Semestre, 2. Semestre, Período Letivo Especial I - Verão, Período Letivo Especial II - Verão.
- CARACTER - /.

Na tabela 14 são descritos os atributos do conjunto de dados "Relação de notas do ENEM".

Tabela 14 – Organização dos atributos do conjunto de dados "Relação de notas do ENEM" (apenas ingressantes em 2011)

Atributo	**Domínio**	**Descrição**
Item	Numérico	Número identificador do aluno
Número de inscrição	Numérico	Número de inscrição do candidato
Nota em Linguagem e códigos	Numérico	Nota do Candidato no ENEM
Nota em Ciências humanas	Numérico	Nota do Candidato no ENEM
Nota em Ciências da natureza	Numérico	Nota do Candidato no ENEM
Nota em Matemática	Numérico	Nota do Candidato no ENEM
Nota em Redação	Numérico	Nota do Candidato no ENEM
Nota no Curso	Numérico	Nota do Candidato no ENEM
Data Nasc.	Data	Data de Nascimento

Fonte: o autor

Por último, na tabela 15 é descrito o conjunto de dados "Desempenho acadêmico".

[6] Código do curso

Tabela 15 – Organização dos atributos do conjunto de dados “Desempenho acadêmico”

Atributo	**Domínio**	**Descrição**
ID_PESSOA	Numérico	Código identificador
ID_ALUNO	Numérico	Código identificador
MATR_ALUNO	Numérico	Matrícula do aluno
NUM_VERSAO	Numérico	Versão do software
NOME_CURSO	Categórico	Nome do curso
COD_CURSO	Categórico	Código do curso
ID_VERSAO_CURSO	Numérico	Código identificador
ANO	Numérico	Ano da disciplina cursada
COD_ATIV_CURRIC	Categórico	Código da disciplina
NOME_ATIV_CURRIC	Categórico	Nome da disciplina
CREDITOS	Numérico	Número de créditos
MEDIA_FINAL	Numérico	Média final do aluno na disciplina
DESCR_SITUACAO	Categórico	Situação do aluno na disciplina
PERIODO	Categórico	Semestre que o aluno cursou a disciplina
ID_CURSO_ALUNO	Numérico	Indeterminado
SITUACAO_ITEM	Numérico	Situação do aluno (numérico)
TOTAL_CARGA_HORARIA	Numérico	Total de horas da disciplina
FORMA_INGRESSO	Categórico	Forma de Ingresso
ANO_INGRESSO	Numérico	Ano do ingresso
FORMA_EVASÃO	Categórico	Forma de Evasão
ANO_EVASÃO	Numérico	Ano da evasão
SEXO	Categórico	Sexo

Fonte: o autor

A partir dos conjuntos de dados foi idealizada e desenvolvida uma ferramenta para a realização do pré-processamento necessário para a aplicação dos algoritmos de mineração dos dados nos conjuntos de dados da universidade.

Apesar dos conjuntos de dados estarem armazenadas no formato *XLS*, o formato *csv* foi utilizado na ferramenta desenvolvida visto que o formato *XLS* é um formato binário e proprietário, portanto, sua manipulação através de linguagens de programação é dificultada. Devido ao fato de que os *softwares* de planilha eletrônica populares possuem uma forma fácil de salvar as planilhas no formato *csv*, e o formato *csv* ser um formato texto, sua utilização se

torna atrativa para ser processado com uma linguagem de programação. O *Weka* aceita como entrada o formato *csv*.

O código fonte da ferramenta desenvolvida está disponível para *download* na *Internet*[7], e é descrita a seguir:

A ferramenta de geração do arquivo de análise (*GeraAnálise*) processa três conjuntos de dados: "Desempenho acadêmico", "Relação completa de alunos" e "Relação de disciplinas do curso", e gera um novo conjunto de dados descritos na tabela 16. As tarefas de pré-processamento contidas na ferramenta são:

- Seleção: Foi realizada tanto a seleção dos atributos, como a seleção das instâncias a serem utilizadas na mineração.

  A ferramenta é responsável por remover as instâncias que não podem ser utilizadas para a análise. Tais instâncias consistem de alunos que evadiram logo após o término do primeiro semestre – que não será possível prevenir a evasão dos mesmos, devido aos dados do desempenho acadêmico não estarem disponíveis para tais alunos antes da evasão – e de alunos que estão cursando o primeiro semestre, e, portanto, sem informações sobre o desempenho acadêmico do semestre.
- Integração: A ferramenta também integra atributos contidos nos conjuntos de dados "Desempenho acadêmico" e "Relação completa de alunos'.
- Transformação: Foi utilizada a técnica de agregação para a obtenção de diversos atributos, como por exemplo, médias das disciplinas e idade do aluno. Também foi utilizada a técnica de generalização para reduzir as classes de saída para *Regular* e *Evadido*, assim como para simplificar o atributo relativo à forma de ingresso do aluno.

Os atributos relativos ao desempenho acadêmico do aluno são calculados até o semestre anterior ao semestre que ocasionou a evasão, pois caso contrário, não é possível utilizar o modelo para predizer a evasão. Exemplo: Um aluno ingressou em 2006/02 e evadiu em 2008/01. Os atributos então são calculados somente para 2006/02 e 2007/01. Apenas dois atributos são utilizados para o semestre de 2007/02 deste aluno: (*diferenca_creditos* e *razao_creditos*. A utilização desses dois atributos são propostos em Obsivac et al. (2012).

O conjunto de dados gerado através da ferramenta *GeraAnálise* possui o formato descrito na tabela 16.

Tabela 16 – Atributos do conjunto de dados após a execução da ferramenta *GeraAnálise*

[7] Disponível em https://sites.google.com/site/alanot/

Atributo	Domínio	Descrição
matricula	Numérico	Matrícula do aluno
creditos_cursados_ec	Numérico	Total de créditos que o aluno foi aprovado no curso
disciplinas_concluidas_ec	Numérico	Total de disciplinas que o aluno foi aprovado
disciplinas_cursadas	Numérico	Total de disciplinas que o aluno se matriculou
creditos_cursados_outros	Numérico	Total de créditos obtidos em outros cursos sem aproveitamento
disciplinas_concluidas_outros	Numérico	Total de disciplinas que o aluno foi aprovado em outros cursos
semestres_concluidos	Numérico	Total de semestres concluídos
segundatentativa_disciplina	Numérico	Total de disciplinas que o aluno cursou mais de uma vez
reprovacoes_recuperadas	Numérico	Número de reprovações que o aluno foi aprovado posteriormente
total_reprovacoes_desistencias	Numérico	Total de reprovações e trancamentos
media_ponderada	Numérico	Média ponderada entre as notas nas disciplinas e créditos obtidos
media_disciplinas	Numérico	Média de todas as disciplinas cursadas
razao_creditos	Numérico	Razão entre os créditos obtidos e créditos do semestre atual
diferenca_creditos	Numérico	Diferença entre os créditos obtidos e créditos do semestre atual
sexo	Categórico	Sexo do aluno
idade	Numérico	Idade do aluno no ingresso
tipoingresso	Categórico	IngressoTradicional,Transferência, Reingresso, Diplomado
periodoingresso	Numérico	200602, 200701, …
repr_calc1	Categórico	Reprovou em Cálculo I
repr_algoritmos	Categórico	Reprovou em Algoritmos e Programação
repr_introd_arqcomp	Categórico	Reprovou em Introdução à Arquitetura de Computadores
repr_introd_engcomp	Categórico	Reprovou em Introdução à Engenharia de

		Computação[8]
repr_ga	Categórico	Reprovou em Geometria Analítica[9]
repr_primeirosem	Numérico	Número de reprovações no primeiro semestre
semestres_cursados	Numérico	Número de semestres cursados
situacao	Categórico	Regular ou Evadido

Fonte: o autor

Outras ferramentas também foram desenvolvidas. A primeira com a finalidade de verificar a taxa de desistências em contraste com reprovações por nota, para verificar como seriam organizados os atributos na ferramenta *GeraAnálise*. A segunda ferramenta foi criada para verificar se o conjunto de dados do local de residência dos alunos poderia ser utilizado, assim como estimar uma taxa de evadidos para alunos com o local de residência em Bagé e de outras cidades. Essas ferramentas são descritas a seguir:

- Desempenho geral nas disciplinas *AprDisciplinas*: Esta ferramenta tem por entrada o conjunto de dados do desempenho acadêmico. Através desta ferramenta é gerado um arquivo resultante contendo o número de aprovações, reprovações, trancamentos e desistências para cada semestre que foi disponibilizada a disciplina cursada pelos alunos. Apesar de ter sido idealizada para analisar as disciplinas do curso de Engenharia de Computação, pode ser aplicado nos conjuntos de dados de desempenho acadêmico dos outros cursos. Através do arquivo resultante podem ser gerados gráficos para analisar como variam as aprovações, reprovações e trancamentos em um período de tempo, e com isto verificar se tais reprovações consistem em um problema local ou se persistem sempre que a disciplina for ofertada.
- Ferramenta para eliminar cidades redundantes (*ConverteCidades*): Esta ferramenta tem por entrada um arquivo *csv* contendo a matrícula e a cidade do aluno, e converte todas as instâncias para "Bagé" e "ForaDeBagé". Também elimina registros duplicados para alunos com duas cidades cadastradas, prevalecendo "ForaDeBagé" quando o aluno tem endereço de outra cidade e de Bagé. Esta ferramenta pode ser aprimorada para diferenciar registros de cidades vizinhas de registros de cidades de outros estados, se um atributo relativo ao estado do aluno estiver disponível.

Através das ferramentas criadas, conjuntos de dados relativos a outros cursos também

[8] Abrange a disciplina de Prática Integrada de Engenharia Computacional , não mais ofertada, sendo ambas equivalentes.
[9] Abrange a disciplina de Álgebra Linear e Geometria Analítica, não mais ofertada, sendo ambas equivalentes.

podem ser gerados.

4.3 Mineração de Dados e interpretação dos resultados

Segundo Maimon e Rokach (2010), na etapa de mineração de dados é realizada a escolha da tarefa de Mineração de Dados apropriada, onde é decidido se será realizado classificação, regressão ou clusterização; a escolha do algoritmo de mineração de dados, onde serão considerados algoritmos que podem favorecer a precisão ou a interpretação do problema; e a execução do algoritmo de mineração de dados, para a verificação dos resultados. No Anexo A pode ser visto a tabela 25, onde são comparadas as vantagens e desvantagens de cada técnica utilizada em Mineração de Dados. Segundo Obsivac et al. (2012), a utilização dos algoritmos *J48 (C4.5)*, *IB1*, *PART*, *SMO* (implementação de uma SVM) e *NaiveBayes* cobrem todos os tipos de algoritmos de aprendizado supervisionado. Além destes algoritmos, serão testados o algoritmo de árvore de decisão *SimpleCart* (implementação do CART) e o algoritmo de aprendizagem por regras de indução *JRip*. Por último é testado o algoritmo de redes neurais *MultilayerPerceptron*, a fim de verificar seu desempenho em relação aos demais algoritmos.

O critério de escolha dos algoritmos foi baseado nos trabalhos propostos por Obsivac et al. (2012) e Dekker (2009).

A comparação dos diversos algoritmos surge devido ao fato de que um algoritmo de aprendizado não traz resultados uniformemente melhores para todos os conjuntos de dados, e também para decidir na escolha do algoritmo mais adequado para o conjunto de dados (KOTSIANTIS, 2007).

Para a tarefa de clusterização foi utilizado o algoritmo *K-Means* e para a tarefa de regras de associação foi utilizado o algoritmo *Apriori*, visto que são os algoritmos mais utilizados para essas tarefas (WU et al., 2007).

O experimento consiste em gerar um novo conjunto de dados a partir dos conjuntos de dados “Desempenho acadêmico”, “Relação completa de alunos” e “Relação de disciplinas do curso”. Este conjunto de dados é gerado através da ferramenta *GeraAnálise* descrita anteriormente. Em seguida são aplicados os algoritmos citados anteriormente no conjunto de dados, e, se necessário, realizada a remoção de alguns atributos através de uma estratégia de tentativa e erro, para verificar se há ganho ou perda no desempenho dos algoritmos. Neste trabalho foram verificados três aspectos para os experimentos de classificação realizados:

- Desempenho dos algoritmos aplicados: Foram verificadas as taxas de Verdadeiros

Positivos (TP) para a classe de evadidos, caracterizando os evadidos que foram identificados corretamente, a taxa de Falsos Positivos (FP), caracterizando alunos evadidos que foram classificados como regulares, e a taxa de Falsos Negativos (FN), caracterizando os alunos regulares que foram classificados como evadidos, sendo assim, alunos supostamente em risco de evasão.

- Regras resultantes dos modelos: Foram verificadas as regras geradas pelos modelos de árvores de decisão e regras de associação a fim de identificar os atributos possivelmente relacionados com a ocorrência da evasão do aluno.
- Refinamento do modelo: Através da remoção de alguns atributos utilizados no experimento, deseja-se verificar se há alteração no desempenho dos algoritmos, assim como nas regras geradas pelo modelo. Diversas tentativas foram realizadas para agregar um conjunto de dados com um bom desempenho na aplicação dos algoritmos. O modelo apresentado neste trabalho é o modelo que foi possível obter melhores resultados na realização dos experimentos.

4.3.1 Classificação

Os resultados a seguir foram obtidos após a aplicação dos algoritmos de mineração de dados no conjunto de dados em que foi realizado o pré-processamento anteriormente. Todos os algoritmos testados foram utilizados com seus parâmetros padrões e a validação cruzada foi realizada separando o conjunto em 10 *folds*, conforme proposto em Obsivac et al. (2012) para fins de comparação.

Os atributos utilizados para este experimento foram:

- Sexo
- Idade do aluno no ano do ingresso
- Forma de ingresso (IngressoTradicional (ENEM ou Vestibular), Transferências (englobando todos os tipos de transferências),Reingresso, Diplomado.
- Ano e semestre de ingresso no formato YYYYMM (MM = 01 para 1^{o} semestre e MM = 02 para 2^{o} semestre). Isto faz com que seja possível a comparação de datas (e.g. 200702 < 200801)
- Número de créditos obtidos com a aprovação nas disciplinas do curso de Engenharia de Computação
- Número de disciplinas em que o aluno obteve aprovação no curso de Engenharia de

Computação

- Número total de disciplinas cursadas no curso de Engenharia de Computação
- Número total de disciplinas em que o aluno obteve aprovação em outros cursos, que não fazem parte do currículo do curso de Engenharia de Computação
- Número de créditos obtidos com a aprovação nas disciplinas de outros cursos, que não fazem parte do currículo do curso de Engenharia de Computação
- Número de semestres concluídos no curso de Engenharia de Computação
- Número de vezes que o aluno cursou uma disciplina mais de uma vez
- Número total de reprovações e trancamentos do aluno
- Média de todas as disciplinas cursadas pelo aluno (exceto trancamentos)
- Média ponderada das disciplinas em que o aluno foi aprovado e o número de créditos obtidos pelo aluno
- Razão entre o número de créditos obtidos com o número de créditos do semestre que o aluno está cursando
- A diferença entre o número de créditos obtidos e o número de créditos do semestre que o aluno está cursando
- Reprovou em Cálculo I
- Reprovou em Geometria Analítica
- Reprovou em Introdução a Engenharia de Computação
- Reprovou em Introdução a Arquitetura de Computadores
- Número de semestres cursados
- Classe de saída para aluno: Regular ou Evadido

O experimento foi realizado para alunos ingressantes no período de 2006/2 a 2011/2. O conjunto de dados consiste em uma amostra de 215 instâncias, sendo que 126 dessas instâncias consistem de alunos regulares e 89 de evadidos.

Na tabela 17 são descritos os resultados para o experimento, obtidos para os diversos algoritmos.

Tabela 17 – Resultados da aplicação de diversos algoritmos para o experimento

Algoritmo	**Regulares**			**Evadidos**			**Kappa**
	TP	**FP**	**Precisão**	**TP**	**FP**	**Precisão**	
J48	0.881	0.18	0.874	0.82	0.119	0.83	0.7023

IB1	0.817	0.315	0.786	0.685	0.183	0.726	0.507
PART	0.889	0.191	0.868	0.809	0.111	0.837	0.7013
NaiveBayes	0.579	0.09	0.901	0.91	0.421	0.604	0.4557
SMO	0.968	0.236	0.853	0.764	0.032	0.944	0.7534
SimpleCart	0.913	0.236	0.846	0.764	0.087	0.861	0.6881
JRip	0.944	0.247	0.844	0.753	0.056	0.905	0.7149
MultilayerPerceptron	0.937	0.146	0.901	0.854	0.063	0.905	0.797

Fonte: o autor

Verificando os índices de *Kappa*, para o experimento, pode ser visto que apesar de nenhum dos experimentos ter obtido um valor considerado excelente de acordo com a tabela 9, diversos algoritmos apresentaram valores de *Kappa* considerados substanciais, sendo eles *J48, PART, SMO, SimpleCart, JRip, MultilayerPerceptron*. Os valores de verdadeiros positivos para estes algoritmos estão entre 75% a 85%, o que são valores excelentes para um problema de tamanha complexidade que é a evasão. Entretanto, para entender melhor os atributos considerados pelos modelos, iremos interpretar os resultados realizando a verificação do modelo gerado pelas árvores de decisão, para verificar se o modelo pode ser utilizado para a predição da evasão.

A aplicação de um algoritmo de árvore de decisão irá exibir entre parênteses no nó folha o número de casos que foram satisfeitos na regra, e uma classe associada para a regra. Quando a regra possui casos em que a etapa de validação cruzada classificou incorretamente, o número de casos classificados incorretamente vai ser exibido após uma barra (”/´´) dentro dos parênteses. Com o algoritmo de árvore de decisão *SimpleCart* foi possível gerar a seguinte árvore:

CART Decision Tree
disciplinas_cursadas < 20.5
| periodoingresso < 200901.5: Evadido(55.0/2.0)
| periodoingresso >= 200901.5
| | disciplinas_cursadas < 6.5
| | | periodoingresso < 201101.5: Evadido(16.0/2.0)
| | | periodoingresso >= 201101.5: Regular(3.0/0.0)
| | disciplinas_cursadas >= 6.5: Regular(28.0/6.0)

disciplinas_cursadas >= 20.5: Regular(91.0/12.0)
Number of Leaf Nodes: 5
Size of the Tree: 9

A interpretação dos resultados descritos acima envolve um entendimento do conjunto de dados. Sabendo que o conjunto de dados envolve alunos que ingressaram entre os períodos de 2006/02 e 2011/02. Através das regras geradas, houve um grande número de alunos evadidos que ingressaram até 2009/01. Com isto, a regra gerada informa que alunos que ingressaram desde 2006/02 até 2009/01 e cursaram até 20 disciplinas evadiram, isto é, alunos que não cursaram apenas 20 disciplinas no período compreendido em 6 semestres, ou 3 anos, evadiram.

O atributo relativo às disciplinas cursadas informa que o aluno apenas se matriculou na disciplina, não que concluiu. Levando em conta que o semestre regular do aluno tem em média 5 disciplinas, o aluno em 6 semestres teria 30 disciplinas cursadas. Entretanto, para 3 disciplinas, que é o mínimo permitido de disciplinas por semestre na UNIPAMPA, o aluno em 6 semestres cursaria apenas 18. Isto indica que alunos que cursam o mínimo de disciplinas permitidas no curso tendem a evadir em 3 anos.

Outra regra gerada foi que alunos que ingressaram entre o período de 2009/02 a 2011/01 e não cursaram 7 disciplinas evadiram. O raciocínio desenvolvido acima vale também para esta regra, mas o período diminuiu para 1 ano.

O algoritmo J48 obteve a seguinte árvore:

```
J48 pruned tree
------------------
disciplinas_cursadas <= 20
|   periodoingresso <= 200901: Evadido (57.0/2.0)
|   periodoingresso > 200901
|   |   disciplinas_cursadas <= 6
|   |   |   periodoingresso <= 201101: Evadido (18.0/2.0)
|   |   |   periodoingresso > 201101: Regular (3.0)
|   |   disciplinas_cursadas > 6
|   |   |   periodoingresso <= 201001
|   |   |   |   semestres_cursados <= 3: Evadido (3.0)
|   |   |   |   semestres_cursados > 3
```

```
|  |  |  |  |  repr_algoritmos = nao: Regular (10.0/1.0)
|  |  |  |  |  repr_algoritmos = sim: Evadido (3.0/1.0)
|  |  |  periodoingresso > 201001: Regular (18.0)
disciplinas_cursadas > 20
|  semestres_concluidos <= 1
|  |  periodoingresso <= 200902
|  |  |  semestres_cursados <= 6: Evadido (6.0/1.0)
|  |  |  semestres_cursados > 6
|  |  |  |  periodoingresso <= 200602
|  |  |  |  |  semestres_cursados <= 11: Evadido (4.0)
|  |  |  |  |  semestres_cursados > 11: Regular (4.0)
|  |  |  |  periodoingresso > 200602: Regular (39.0/3.0)
|  |  periodoingresso > 200902: Regular (17.0)
|  semestres_concluidos > 1: Regular (33.0)

Number of Leaves  :  13
Size of the tree :  25
```

Além das regras já contempladas anteriormente, a árvore informa que alunos que cursaram 20 disciplinas e tem pelo menos 1 semestre concluído são regulares.

A árvore também gerou regras indesejadas envolvendo alunos que reprovaram em Algoritmos e Programação que são regulares, e alunos que não reprovaram em Algoritmos e Programação que são evadidos.

Com este experimento pode-se concluir que houve uma mudança no perfil dos alunos ingressos nos períodos de ingresso antes e depois de 2009/01. As informações obtidas na árvore são relevantes para indicar que alunos que cursam o mínimo de disciplinas permitidas, tem propensão a evadir.

4.3.2 Clusterização

Os resultados de clusterização a seguir foram obtidos utilizando o algoritmo *SimpleKMeans*, que é a implementação do *Weka* para o algoritmo *K-means*. Para este experimento, o único parâmetro do algoritmo a ser ajustado é o número de *clusters*.

Para este conjunto de dados, tem-se a intenção de obter dois tipos de informações com

a clusterização:

1. Identificação de diferentes perfis de alunos, tanto para regulares como para evadidos, verificando diferenças de características para tais perfis.
2. Identificação de possíveis atributos que sejam comuns entre os alunos evadidos que não são contemplados pelas árvores de decisão.

Ao aumentar o número de *clusters* gerados, pode-se verificar se os valores médios dos atributos são mantidos para os diversos *clusters*, ou se há uma variação, indicando possíveis ruídos, ou seja, alunos evadidos e regulares que não seguem a tendência das características dos demais evadidos.

Os resultados são baseados no conjunto de dados do experimento da classificação.

Na tabela 18 são exibidos os centróides de para cada atributo no conjunto de dados obtidos com a aplicação do *K-Means*, que indicam o valor médio do atributo. Para os atributos categóricos o centróide é a categoria mais frequente. A mesma tabela pode ser vista como a clusterização para um único *cluster*.

Tabela 18 – Centróide para cada atributo

Atributo	**Centróide**
creditos_cursados_ec	50.2884
disciplinas_concluidas_ec	12.8186
disciplinas_cursadas	22.3302
creditos_cursados_outros	1.2093
disciplinas_concluidas_outros	0.4047
semestres_concluidos	0.9395
segundatentativa_disciplina	3.7674
reprovacoes_recuperadas	0.6512
total_reprovacoes_desistencias	9.9628
media_ponderada	7.0941
media_disciplinas	4.5422
razao_creditos	3.0399
diferenca_creditos	33.0791
Sexo	M
Idade	22.786
Tipoingresso	IngressoTradicional

Periodoingresso	200859.8
repr_calc1	sim
repr_algoritmos	nao
repr_introd_arqcomp	sim
repr_introd_engcomp	nao
repr_alga	sim
repr_primeiro	2.0977
semestres_cursados	6.014
Situação	Regular

Fonte: o autor

A tabela 18 não traz resultados úteis para o problema, porque que ela consiste apenas da média dos atributos para todos os alunos da amostra. Entretanto, pode ser observado que reprovações em Cálculo I, Introdução a Arquitetura de Computadores e Álgebra Linear e Geometria Analítica são frequentes para os alunos.

Experimentos podem ser realizados com uma infinidade de *clusters*. Para este trabalho, os experimentos realizados foram delimitados para 2, 3, 4 e 5 *clusters*.

Na tabela 19 são exibidos os resultados para 2 clusters.

Tabela 19 – Resultados do experimento para 2 *clusters*

	Cluster 1 (108)	**Cluster 2 (107)**
creditos_cursados_ec	77.0648	23.2617
disciplinas_concluidas_ec	19.0556	6.5234
creditos_cursados_outros	2.0556	0.3551
disciplinas_concluidas_outros	0.6667	0.1402
semestres_concluidos	1.6667	0.2056
segundatentativa_disciplina	3.1296	4.4112
reprovacoes_recuperadas	0.6944	0.6075
total_reprovacoes_desistencias	8.9815	10.9533
media_ponderada	7.1065	7.0816
media_disciplinas	5.4508	3.6252
razao_creditos	4.3849	1.6823
diferenca_creditos	58.4167	7.5047

Sexo	M	M
Idade	22.037	23.5421
Tipoingresso	IngressoTradicional	IngressoTradicional
Periodoingresso	200866.037	200853.5701
repr_calc1	nao	Sim
repr_algoritmos	nao	Sim
repr_introd_arqcomp	nao	Sim
repr_introd_engcomp	nao	Não
repr_alga	nao	Sim
repr_primeiro	0.9444	3.2617
semestres_cursados	6.9167	5.1028
Situação	Regular	Evadido

Fonte: o autor

Na tabela 19 podem ser extraídas diversas informações. Pode ser interpretado que alunos evadidos têm em média 3 reprovações no primeiro semestre, 6 disciplinas que obtiveram aprovação e uma média de 5 semestres cursados. Tal informação indica que os evadidos tem em média 1 aprovação por semestre.

A média de idade e de reprovações posteriormente recuperadas tem pouca diferença para as duas classes.

A média ponderada também tem pouca diferença para as duas classes, o que pode sugerir que é um atributo de pouca relevância para o problema.

Alunos regulares, entretanto, têm em média 2 créditos cursados em outros cursos, o que pode indicar que alunos regulares tendem a cursar disciplinas em outros cursos. Sabendo que o curso de Engenharia de Computação é noturno, isto pode indicar que se o aluno regular cursa alguma disciplina de outro curso, ele não possui vínculo empregatício.

Na tabela 20 são exibidos os resultados para 3 clusters.

Tabela 20 – Resultados do experimento para 3 *clusters*

	Cluster 1 (74)	**Cluster 2 (96)**	**Cluster 3 (45)**
creditos_cursados_ec	72.2973	19.4375	79.9111
disciplinas_concluidas_ec	17.8919	5.3958	20.3111
creditos_cursados_outros	2.2162	0.0208	2.0889

disciplinas_concluidas_outros	0.6892	0.0104	0.7778
semestres_concluidos	1.527	0.1458	1.6667
segundatentativa_disciplina	2.9324	4	4.6444
reprovacoes_recuperadas	0.5811	0.4792	1.1333
total_reprovacoes_desistencias	8.1892	10.25	12.2667
media_ponderada	7.1304	7.1145	6.9908
media_disciplinas	5.5016	3.5179	5.1498
razao_creditos	4.3322	1.4452	4.3168
diferenca_creditos	53.9459	3.875	61.0667
sexo	M	M	M
idade	22.2568	23.7396	21.6222
tipoingresso	IngrTradicional	IngrTradicional	IngrTradicional
periodoingresso	200866.1	200855.4	200859
repr_calc1	Não	Sim	Não
repr_algoritmos	Não	Sim	Não
repr_introd_arqcomp	Não	Sim	Não
repr_introd_engcomp	Não	Não	Não
repr_alga	Não	Sim	Sim
repr_primeiro	0.6486	3.3229	1.8667
semestres_cursados	6.5676	4.6667	7.9778
situacao	Regular	Evadido	Regular

Fonte: o autor

O cluster 3 tem por característica um grande número de alunos que reprovaram em Álgebra Linear e Geometria Analítica e são regulares, o que pode indicar que tal atributo seja irrelevante para predizer a evasão.

Da mesma forma que no experimento anterior, a média ponderada não tem grande diferença para a classe dos evadidos e regulares, o que também pode indicar que o atributo seja irrelevante para o problema.

A média do número de reprovações e desistências para o *cluster* dos evadidos é um valor intermediário entre os demais *clusters*, o que pode indicar que tal atributo também tem pouca capacidade de predizer a evasão.

Em geral, os dois clusters de alunos regulares tem poucas diferenças entre si. As diferenças mais significativas foram que o cluster 3 tem um maior número de créditos

cursados, disciplinas concluídas e reprovações em disciplinas. Tal informação por si só não traz conhecimento útil para tratar da evasão. Entretanto, com a realização deste experimento, foi possível notar atributos que podem indicar ser irrelevantes. Sugere-se que em futuras análises sejam removidos tais atributos.

Com o *cluster* adicional, a média dos atributos para o *cluster* dos evadidos foi alterada, mas houve mudança apenas nos atributos dos créditos cursados e disciplinas concluídas, diferindo apenas em 1 disciplina e 4 créditos.

Na tabela 21 foram gerados 3 *clusters* para os alunos regulares. Da mesma forma que o experimento anterior, houve uma pequena variação para a média dos atributos dos evadidos. Entretanto, há uma grande diferença entre os alunos regulares, surgindo três perfis de alunos: alunos no fim do curso (*cluster* 1), alunos no meio do curso (*cluster* 4) e alunos mais recentes (*cluster* 3).

Analisando a tabela, podem ser extraídas duas informações: o *cluster* 3 é um ano mais recente que o *cluster* dos evadidos, o que pode sugerir que, como o aluno já reprovou em Introdução a Arquitetura de Computadores, este grupo é propenso a evadir. Outra indicação é que 6 disciplinas concluídas podem sugerir que o aluno recuperou a disciplina reprovada, logo a conclusão do primeiro semestre faz com que o aluno tenha baixa propensão a evadir. Logo, tal grupo pode (ou não) ser propenso a evadir.

Por último, analisando a tabela 22, observa-se que foi obtido dois perfis de alunos evadidos.

O perfil do aluno evadido descrito anteriormente manteve suas características. Entretanto, há um novo *cluster* caracterizando alunos evadidos (*cluster* 3). Tal cluster apresenta uma nova informação: alunos evadidos com um bom desempenho acadêmico, tendo média geral superior que 2 dos *clusters* dos alunos regulares. Tais alunos tem por característica um menor número de reprovações, assim como nenhuma reprovação no primeiro semestre, tendo em média 3 semestres cursados. Tais alunos podem caracterizar um grupo de alunos que evadem por motivos que ainda não podem ser observados diretamente nos conjuntos de dados, dentre eles uma possível transferência para outro curso ou outra universidade, ou até mesmo por motivos pessoais.

Recomenda-se nos trabalhos futuros identificar os alunos contemplados pelo novo *cluster* dos alunos evadidos para entender melhor a evasão dos mesmos.

Tabela 21 – Resultados do experimento para 4 *clusters*

	Cluster 1 (43)	**Cluster 2 (78)**	**Cluster 3 (38)**	**Cluster 4 (56)**
creditos_cursados_ec	124.1628	19.4359	23.8421	54.4821
disciplinas_concluidas_ec	30.3721	5.5897	6.4211	13.75
creditos_cursados_outros	4.2326	0.1026	0.2105	1.1071
disciplinas_concluidas_outros	1.3256	0.0513	0.0789	0.4107
semestres_concluidos	3.0465	0.1538	0.2368	0.8929
segundatentativa_disciplina	2.4884	4.8974	2.2105	4.2321
reprovacoes_recuperadas	0.7674	0.6026	0.2632	0.8929
total_reprovacoes_desistencias	8.0698	11.6923	7.2895	10.8214
media_ponderada	7.2996	7.0965	7.0715	6.9483
media_disciplinas	6.4294	3.335	4.6302	4.7149
razao_creditos	7.247	1.4732	1.4619	3.0624
diferenca_creditos	105.093	4.0641	6.6316	36.1429
sexo	M	M	M	M
idade	21.6047	23.5513	22.7105	22.6786
tipoingresso	IngressoTradicional	IngressoTradicional	IngressoTradicional	IngressoTradicional
periodoingresso	200792	200849.9	200956.4	200860.2
repr_calc1	Não	Sim	Não	Sim
repr_algoritmos	Não	Sim	Não	Não
repr_introd_arqcomp	Não	Sim	Sim	Não
repr_introd_engcomp	Não	Não	Não	Não
repr_alga	Não	Sim	Não	Sim
repr_primeiro	0.4186	3.5513	1.2895	1.9107
semestres_cursados	9.0233	5.1667	4.1579	6.1429
situacao	Regular	Evadido	Regular	Regular

Fonte: O autor

Tabela 22 – Resultados do experimento para 5 *clusters*

	Cluster 1 (32)	**Cluster 2 (78)**	**Cluster 3 (33)**	**Cluster 4 (48)**	**Cluster 5 (24)**
creditos_cursados_ec	141.5938	19.2308	28.9697	66.5208	26.3333
disciplinas_concluidas_ec	34.25	5.5385	7.6061	16.7708	7.1667
creditos_cursados_outros	4.75	0.1026	0.4848	1.4583	0.5833
disciplinas_concluidas_outros	1.5	0.0513	0.1818	0.5208	0.1667
semestres_concluidos	3.6875	0.1538	0.2727	1.1667	0.2917
segundatentativa_disciplina	3.25	4.8462	0.9091	4.7708	2.875
reprovacoes_recuperadas	0.9688	0.6026	0.1212	1.1042	0.2083
total_reprovacoes_desistencias	9.625	11.6026	4.697	11.9167	8.4167
media_ponderada	7.2611	7.0851	7.2385	7.0116	6.8669
media_disciplinas	6.3441	3.2998	5.4892	4.8881	4.1838
razao_creditos	8.296	1.4626	2.018	3.5151	1.6126
diferenca_creditos	122.3438	3.9872	11.8788	47.4167	9.0833
sexo	M	M	M	M	M
idade	21.125	23.4744	23.4242	22.3125	22.8333
tipoingresso	IngressoTradicional	IngressoTradicional	IngressoTradicional	IngressoTradicional	Transferência
periodoingresso	200751.3	200852.5	200880	200857.5	201005.5
repr_calc1	Não	Sim	Não	Sim	sim
repr_algoritmos	Não	Sim	Não	Não	nao
repr_introd_arqcomp	Não	Sim	Não	Não	nao
repr_introd_engcomp	Não	Não	Não	nao	nao
repr_alga	Não	Sim	Não	sim	nao
repr_primeiro	0.5313	3.5641	0.8788	2.0417	1.2083
semestres_cursados	10.125	5.1154	3.6061	6.9375	4.9167
situacao	Regular	Evadido	Evadido	Regular	Regular

Fonte: o autor

4.3.3 Regras de Associação

O algoritmo utilizado para regras de associação foi o *Apriori*. Antes de aplicar o algoritmo, é necessária a alteração de alguns parâmetros: O primeiro a ser modificado é o *car*, que informa o algoritmo que a associação desejada é com a classe de saída. Também é necessário informar o índice no conjunto de dados em que a classe de saída se encontra.

A confiança mínima utilizada para os experimentos foi 0,9, que é o valor padrão para o *Apriori* no Weka.

Com isto, para a aplicação do Apriori foi necessário a discretização de todos os atributos numéricos. O Weka possui uma opção para a discretização, e está localizada em *Filters* na aba *Preprocessing*. Para isto, é necessário escolher a opção *unsupervised -> attribute -> discretize.*

Também é necessário escolher o número de *bins*, que consiste no número de categorias em que as faixas de valores serão discretizadas. O experimento foi realizado efetuando discretizações entre 2 a 10 bins, utilizando o parâmetro *useEqualFrequency* em *True*. Isto significa que a discretização irá dividir em categorias que terão o mesmo número de instâncias entre elas. A utilização desta técnica é fortemente recomendada em Oracle (2012).

Na tabela 23 são demostradas as faixas de valores para 2 *bins*. Os valores representados por *–inf* e *inf* na nomenclatura da faixa de valores são utilizados pelo Weka para denotar os valores mínimo e máximo para o atributo. Esses valores também são demonstrados na tabela 23.

Tabela 23 – Exemplo de discretização para 2 *bins*

Atributo	**Faixas de valores**	**Mínimo e Maximo para o Atributo**
creditos_cursados_ec	(-inf-29]	[2-216]
	(29-inf)	
disciplinas_concluidas_ec	(-inf-7.5]	[1-51]
	(7.5-inf)	
disciplinas_cursadas	(-inf-19.5]	[2-62]
	(19.5-inf)	
creditos_cursados_outros	(-inf-1]	[0-84]
	(1-inf)	

disciplinas_concluidas_outros	(-inf-0.5]	[0-26]
	(0.5-inf)	
semestres_concluidos	(-inf-0.5]	[0-9]
	(0.5-inf)	
segundatentativa_disciplina	(-inf-2.5]	[0-26]
	(2.5-inf)	
reprovacoes_recuperadas	(-inf-0.5]	[0-5]
	(0.5-inf)	
total_reprovacoes_desistencias	(-inf-8.5]	[1-38]
	(8.5-inf)	
media_ponderada	(-inf-7.03716]	[5.833-8.957]
	(7.03716-inf)	
media_disciplinas	(-inf-4.524975]	[0.572-8.654]
	(4.524975-inf)	
razao_creditos	(-inf-1.944445]	[0.083-19.2]
	(1.944445-inf)	
diferenca_creditos	(-inf-13]	[-26-88]
	(13-inf)	
Idade	(-inf-20.5]	[17-56]
	(20.5-inf)	
Periodoingresso	(-inf-200851.5]	[200602-201102]
	(200851.5-inf)	
repr_primeiro	(-inf-2.5]	[0-5]
	(2.5-inf)	
semestres_cursados	(-inf-5.5]	[2-15]
	(5.5-inf)	

Fonte: o autor

Através da aplicação do *Apriori*, foram geradas 10 regras para cada discretização (valor padrão no Weka), que podem ser vistas no Anexo B, totalizando 90 regras.

As discretizações para 2 a 9 *bins* geraram regras somente associando atributos a alunos regulares. Para 10 *bins*, entretanto, foram geradas regras associando os atributos com alunos evadidos.

Primeiramente analisamos os atributos das associações dos evadidos, que podem ser

vistos para a discretização em 10 *bins*:

- Todas as regras geradas para os alunos evadidos informam que não houve nenhum semestre concluído, ou seja, isso indica que o aluno evadido tem no mínimo 1 reprovação em cada semestre cursado por ele.
- Todas as regras geradas para os alunos evadidos informam que o aluno reprovou em Algoritmos e Programação.
- Todas as regras geradas para os alunos evadidos informam que o aluno não reprovou em Álgebra Linear e Geometria Analítica, o que não significa nada para o problema.
- Os resultados apresentam 5 regras envolvendo reprovação em Cálculo I, todas elas em conjunto com reprovação em Algoritmos e Programação.
- Nenhuma reprovação recuperada, o que pode possivelmente indicar alunos que evadiram após reprovarem mais de uma vez nas disciplinas, ou alunos que evadiram logo após obter uma reprovação.

Desta forma, de acordo com as regras geradas, quando o aluno reprova em Algoritmos e Programação, podem ocorrer diversas situações que há a propensão da evasão:

- Aluno reprova em Algoritmos e Programação no primeiro semestre.
- Aluno reprova em Algoritmos e Programação no primeiro semestre e reprova em alguma disciplina no segundo semestre.
- Aluno reprova duas vezes em Algoritmos e Programação.
- Aluno reprova em Calculo I e em Algoritmos e Programação.

Para as demais discretizações, que consistem de regras generalizando o conjunto de alunos regulares, foi verificado que cada discretização generalizou as regras de uma forma diferente. Para um número maior de *bins*, a maioria das regras geralmente envolve um alto número de disciplinas e/ou créditos cursados, e possuem um menor suporte. Logo, para a análise dos atributos que caracterizam um maior número de alunos regulares e envolvendo um maior número de atributos, podemos verificar para as discretizações em 2 *bins* que:

Alunos que cursaram 20 disciplinas que têm pelo menos 1 semestre concluído e não reprovaram em Algoritmos e Programação, são regulares. Considerando que alunos regulares cursam em média 6 disciplinas por semestre, isto pode indicar que alunos que cursaram 3 semestres e não reprovaram em Algoritmos e Programação tem baixa propensão a evadir.

Para 3 *bins*, as regras envolvendo o número de disciplinas concluídas não trazem nenhuma informação adicional, visto que, devido a discretização, pode-se insinuar que alunos que concluíram apenas 14 de 26 disciplinas cursadas são regulares, o que não é

necessariamente verdade.

Para 3 *bins*, uma regra interessante gerada é que alunos que cursaram no mínimo 26 disciplinas são regulares, e esta regra independe de reprovações. Logo, supondo que o aluno cursou em média 6 disciplinas por semestre, após 5 semestres que ele é aluno da instituição ele tem baixa propensão a evadir.

As demais discretizações envolvem regras similares às regras aqui já discutidas para os alunos regulares, caracterizando uma observação esperada, logo não serão discutidas. Entretanto, elas podem ser vistas no Anexo B.

4.4 Visão geral das técnicas utilizadas

Através da aplicação das diferentes técnicas neste trabalho, foi possível obter novos conhecimentos sobre os alunos evadidos do curso de Engenharia de Computação da UNIPAMPA.

Em linhas gerais, foi possível identificar que:

1. A reprovação em três disciplinas: Algoritmos e Programação, Introdução a Arquitetura de Computadores e Cálculo I são frequentes para os alunos evadidos.
2. Há alunos com um bom desempenho acadêmico que evadiram em até 3 semestres do curso. Através dos atributos utilizados neste trabalho, nada pode ser afirmado ainda sobre tais alunos.
3. Houve alunos que cursaram o mínimo de disciplinas permitidas pela UNIPAMPA, e evadiram em até 3 anos de curso.
4. Houve uma possível mudança no perfil dos alunos evadidos que ingressaram após 2009/01. Tais alunos que ingressaram após este período são caracterizados pelo menor tempo de permanência no curso.

Na tabela 24 é sintetizado o comparativo do conhecimento obtido por cada técnica.

Tabela 24 – Comparativo das técnicas utilizadas

Conhecimento	**Árvores de Decisão**	**Regras de Associação**	**Clusterização**
Tempo de permanência do aluno na universidade influencia na evasão	X	X	X

Grupos de alunos evadidos possuem diferentes características			X
Disciplinas cursadas pelo aluno influenciam na evasão		X	X
Há grupos de alunos evadidos com um bom desempenho acadêmico			X
Há grupos de alunos regulares com características similares aos alunos evadidos			X
Aluno propenso a evadir que ingressou após 2009/01 permanece menos tempo na universidade	X		
Identificação de atributos possivelmente irrelevantes para a predição da propensão a evasão			X

Fonte: o autor

Apesar de ser observado que a clusterização forneceu mais conhecimento sobre o problema, as demais técnicas também demonstraram ser úteis para a obtenção de conhecimento. Através da aplicação da técnica de regras de associação, o conhecimento extraído é expresso na forma de regras do tipo "SE...ENTÃO". Desta forma o conhecimento é mais facilmente interpretável.

5 TRABALHOS FUTUROS

Alguns trabalhos futuros são sugeridos a seguir:

- Com a obtenção dos dados de desempenho acadêmico contendo dados relativos aos próximos semestres, poderá ser verificado se as regras geradas se mantêm, assim como também pode ser verificado as variações para os índices obtidos nos experimentos aqui descritos.
- Da mesma forma que foram gerados modelos para o curso de Engenharia de Computação, modelos para outros cursos também podem ser gerados.
- Desenvolvimento de uma *interface* gráfica para as ferramentas desenvolvidas neste trabalho.
- Desenvolver meios de integrar os modelos gerados no banco de dados da Universidade.
- Obter dados relativos às notas parciais dos alunos nas disciplinas de primeiro semestre, e com isto, realizar um estudo para prever a aprovação, reprovação ou desistência do aluno nas disciplinas, assim como na evasão dos mesmos.
- Explorar as técnicas de Redução de Dimensionalidade, especialmente na Seleção de Características, com a finalidade de verificar novos resultados, ao encontrar subconjuntos otimizados dos atributos.

6 CONSIDERAÇÕES FINAIS

Através das diversas aplicações de árvores de decisão só foi possível afirmar que se o aluno não cursou 20 disciplinas em até 3 anos, ele tem tendência a evadir. Interpretando este resultado, foi possível identificar que alunos que cursam 3 disciplinas por semestre (o mínimo permitido pela UNIPAMPA), tem tendência a evadir em 3 anos de curso.

Com a técnica de clusterização, foi possível identificar 2 perfis de alunos evadidos. O primeiro perfil é constituído de alunos que cursaram em média 5 semestres, e caracterizado pela reprovação em 3 disciplinas do primeiro semestre do curso: Cálculo I, Algoritmos e Programação e Introdução a Arquitetura de Computadores. Isto pode sugerir que estas disciplinas são fundamentais para o entendimento dos conteúdos aprendidos posteriormente no curso, e a reprovação nelas pode ser indício de que o aluno não terá um bom aproveitamento no curso, tendo por consequência a evasão. Essas disciplinas também são pré-requisitos para várias disciplinas do curso, o que reforça essa ideia.

O segundo perfil de alunos identificados pela clusterização é de evadidos com um bom desempenho acadêmico, que não possuem reprovações em disciplinas do primeiro semestre. Tais alunos cursam em média 3 semestres do curso. O motivo da evasão de tais alunos ainda não é bem compreendido e pode envolver características não contempladas neste trabalho, como, por exemplo, uma eventual transferência para outro curso ou outra universidade, como também na evasão por motivos socioeconômicos. Sugere-se que estes casos sejam estudados em trabalhos futuros.

Também foram identificados grupos de alunos regulares que cursam disciplinas em outros cursos. Isto pode indicar alunos que pretendem realizar transferência para outro curso, mas apenas se matricularam nas disciplinas do curso desejado. Por sugestão, um atributo a ser incluído nas bases de dados é o curso da disciplina que o aluno se matriculou. Através deste atributo pode ser verificado o turno do curso que a disciplina foi cursada, assim como também será possível verificar se o aluno cursa tais disciplinas em um único curso.

Outra característica identificada na clusterização é que a média ponderada não tem diferença significativa entre alunos evadidos e alunos regulares, o que pode indicar que este atributo é irrelevante para o problema. Outro atributo possivelmente irrelevante para os alunos evadidos é a reprovação em Álgebra Linear e Geometria Analítica, pois não ocorre com frequência para tais alunos.

Através das regras de associação, foi possível observar que alunos que reprovam em Algoritmos e Programação são propensos a evadir. Com as regras também é possível

interpretar que, se o aluno reprovar em Algoritmos e Programação e não evadir, se ele obtiver reprovação em alguma disciplina no segundo semestre, ele também é propenso a evadir.

Através das regras de associação também foi possível identificar que alunos com 26 disciplinas cursadas, ou seja, 5 semestres, tem baixa propensão a evadir. Logo, quanto mais o aluno progride no curso, menor a tendência a evadir.

Devido à complexidade de tratar o problema da evasão, nota-se a relevância de utilizar técnicas de mineração de dados no auxílio da identificação de possíveis razões associadas.

Este é o primeiro estudo envolvendo a aplicação de algoritmos de mineração de dados para identificação da propensão a evasão na UNIPAMPA. Com ele pode ser visto que há muito que melhorar na qualidade dos dados obtidos dos alunos, seja através de questionários, ou através de fichas cadastrais, assim como na integração de tais dados no banco de dados da Universidade.

Com a realização deste trabalho, foi possível observar um grande potencial nas técnicas aplicadas. Tem-se a intenção de que as análises descritas neste trabalho possam ser melhoradas com a obtenção de novos conjuntos de dados. Recomenda-se a utilização dos mesmos algoritmos a fim de comparação.

Para o experimento, apenas 215 alunos puderam ser utilizados para a mineração. Os índices dos modelos gerados poderão ser revisitados com o passar do tempo, e novos resultados obtidos, visto que haverá os dados do desempenho acadêmico dos alunos que ingressaram em 2012 e anos seguintes, assim como novos dados de evasão.

Das técnicas aplicadas, as que trouxeram resultados mais expressivos foram a clusterização e aplicação de regras de associação, visto que com elas foi possível explorar melhor o conjunto de dados, o que não foi possível com a classificação. Recomenda-se utilizar essas técnicas em trabalhos futuros. A inspeção das regras de associação geradas, entretanto, pode ser uma tarefa trabalhosa, visto que muitas regras geradas podem possuir pouca diferença entre elas, ou envolver a interpretação de associações entre atributos de alta relevância para o problema com atributos de baixa relevância.

A etapa do pré-processamento demanda o maior tempo no processo. Através das ferramentas de suporte desenvolvidas neste trabalho, o pré-processamento envolvido nos trabalhos futuros levará menos tempo para ser concluído. Novos conjuntos de dados podem então ser integrados nos conjuntos de dados já utilizados, e com isto, novos resultados serão obtidos.

REFERÊNCIAS

ABERNETHY, M. **Data mining with WEKA, Part 2: Classification and clustering.** May 2010. Disponível em: <http://www.ibm.com/developerworks/opensource/library/os-weka2/index.html>. Acesso em: 11 de novembro de 2012.

AGRAWAL, R.; IMIELIŃSKI, T.; SWAMI, A. **Mining association rules between sets of items in large databases.** SIGMOD Rec., ACM, New York, NY, USA, v. 22, p. 207–216, June 1993. ISSN 0163-5808.

ALVES, T. W.; ALVES, V. V. **Fatores determinantes da evasão universitária: uma análise a partir dos alunos da Unisinos.** IV Encontro de Economia Catarinense, 2010.

BERRY, M. J. A.; LINOFF, G. B. **Data mining techniques: for marketing, sales, and customer relationship management.** 2nd. ed. Indianapolis: Wiley Publishing Inc, 2004.

BISHOP, C. M. **Neural Networks for Pattern Recognition.** New York, NY, USA: Oxford University Press, Inc., 1995. ISBN 0198538642.

BRAMER, M. **Principles of Data Mining.** Portsmouth, UK: Springer, 2007. ISBN 978-1-84628-765-7.

BURGES, C. J. C. **A tutorial on support vector machines for pattern recognition.** Data Min. Knowl. Discov., Kluwer Academic Publishers, Hingham, MA, USA, v. 2, n. 2, p. 121–167, jun. 1998. ISSN 1384-5810.

CABENA, P. et al. **Discovering data mining: from concept to implementation.** Upper Saddle River, NJ, USA: Prentice-Hall, Inc., 1998. ISBN 0-13-743980-6.

CAMARGO, S. da S. **Um modelo neural de aprimoramento progressivo para redução de dimensionalidade.** Tese (Doutorado) — Universidade Federal do Rio Grande do Sul, Junho 2010.

CAMPELLO, A. de V. C.; LINS, L. N. **Metodologia de análise e tratamento da evasão e**

retenção em cursos de graduação de instituições federais de ensino superior. XXVIII Encontro Nacional de Engenharia de Produção, Associação Brasileira de Engenharia de Produção, Rio de Janeiro, RJ, Brasil, 10 2008. Disponível em: <http://www.abepro.org.br/biblioteca/enegep2008_TN_STO_078_545_11614.pdf>. Acesso em: 11 de novembro de 2012.

CAP, NUDEPE E PROPLAN. **Unipampa 2011: compromisso de todos!** IV Seminário de Desenvolvimento Profissional Docentes — Planejamento e avaliação da aprendizagem na Educação Superior, Santana do Livramento, Fevereiro 2011. Disponível em: <http://eventos.unipampa.edu.br/seminariodocente/files/2011/03/UNIPAMPA-2011-compromisso-de-todos1.pdf>. Acesso em: 11 de novembro de 2012.

CESTARO, R. **Mineração de dados aplicada à identificação de alunos propensos à evasão do CEULJI/ULBRA de Ji-Paraná/RO**. 96 p. Monografia (Graduação) — Centro Universitário Luterano de Ji-Paraná, Ji-Paraná, Rondônia, 2006.

CHAPMAN, P. et al. **CRISP-DM 1.0.** August 2000. Disponível em: <ftp://ftp.software.ibm.com/software/analytics/spss/support/Modeler/Documentation/14/UserManual/CRISP-DM.pdf>. Acesso em: 11 de novembro de 2012.

CIOS, K. J.; KURGAN, L. A. **Trends in data mining and knowledge discovery.** In: IN: PAL N.R., JAIN, L.C. AND TEODERESKU, N. (EDS.), KNOWLEDGE DISCOVERY IN ADVANCED INFORMATION SYSTEMS. [S.l.]: Springer, 2005. p. 200–2.

CIOS, K. J. et al. **Data Mining: A Knowledge Discovery Approach.** 1st. ed. [S.l.]: Springer, 2007. Hardcover. ISBN 0387333339.

CUNNINGHAM, P. **Dimension Reduction.** School of Computer Science and Informatics, University College Dublin. August 2007.

DATE, C. **An Introduction to Database Systems.** 8th. ed. Boston, MA, USA: Addison-Wesley Longman Publishing Co., Inc., 2003. ISBN 0321197844.

DEKKER, G. W. **Predicting students drop out: a case study.** 22 p. Dissertação (Mestrado)

– Department of Electrical Engineering, Eindhoven University of Technology. Eindhoven, Netherlands. April, 2009.

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

GARCIA, N. M.; ABDALA, A. O.; MATSUSHITA, A. M. **Aceleração de aprendizagem - um inibidor da evasão na universidade.** Acta Scientiarum. Human and Social Sciences, v. 22, 2000.

GARCIA, S. C.; ALVARES, L. O. **Árvores de decisão – algoritmos id3 e c4.5.** Cadernos de Informática, Instituto de Informática da UFRGS, Porto Alegre - RS, v. 1, 2000. Disponível em: <http://seer.ufrgs.br/cadernosdeinformatica/issue/view/996>. Acesso em: 11 de novembro de 2012.

GOEBEL, M.; GRUENWALD, L. **A survey of data mining and knowledge discovery software tools.** SIGKDD Explorations, v. 1, p. 20–33, 1999.

GOLDSCHMIDT, R.; PASSOS, E. **Data Mining – Um Guia Prático.** Rio de Janeiro: Elsevier, 2005.

HALL, M. et al. **The weka data mining software: an update.** SIGKDD Explor. Newsl., ACM, New York, NY, USA, v. 11, p. 10–18, November 2009. ISSN 1931-0145. Disponível em: <http://doi.acm.org/10.1145/1656274.1656278>. Acesso em: 11 de novembro de 2012.

HAND, D. J.; SMYTH, P.; MANNILA, H. **Principles of data mining.** Cambridge, MA, USA: MIT Press, 2001. ISBN 0-262-08290-X.

HARNIK, S. **Gasto com ensino superior é 6,7 vezes maior do que com educação básica.** Outubro 2005. http://www1.folha.uol.com.br/folha/educacao/ult305u17930.shtml. Acesso em: 11 de novembro de 2012.

IWASSO, S. **Cresce diferença entre pública e privada.** Dezembro 2010. <http://www.estadao.com.br/noticias/impresso,cresce-diferenca-entre-publica-e-

privada,650961,0.htm>. Acesso em: 11 de novembro de 2012.

JOSÉ, A. R.; ANDREOLI, G. S. **A evasão na unipampa: Diagnosticando processos, acompanhando trajetórias e itinerários de formação.** Bagé, RS, Brasil, Outubro 2011. Disponível em: <http://porteiras.r.unipampa.edu.br/portais/cap/files/2010/07/Relat%C3%B3rio_final_evas%C3%A3o-na-UNIPAMPA_out20111.pdf> Acesso em: 11 de novembro de 2012.

KOHAVI, R. **Scaling Up the Accuracy of Naive-Bayes Classifiers: A Decision-Tree Hybrid**. In: Second International Conference on Knowledge Discovery and Data Mining, 202-207, 1996.

KOHONEN, T. **Self-organized formation of topologically correct feature maps.** Biological Cybernetics, Springer Berlin / Heidelberg, v. 43, p. 59–69, 1982. ISSN 0340-1200. 10.1007/BF00337288.

KOTSIANTIS, S. B. **Supervised machine learning: A review of classification techniques.** Informatica, p. 249–268, 2007.

LANDIS, J. R.; KOCH, G. G. **The measurement of observer agreement for categorical data.** Biometrics, International Biometric Society, v. 33, n. 1, p. pp. 159–174, 1977. ISSN 0006341X.

MAIMON, O.; ROKACH, L. (Ed.). **Data Mining and Knowledge Discovery Handbook**, 2nd ed. [S.l.]: Springer, 2010. ISBN 978-0-387-09822-7.

MANHÃES, L. M. B. et al. **Identificação dos fatores que influenciam a evasão em cursos de graduação através de sistemas baseados em mineração de dados: Uma abordagem quantitativa.** VIII Simpósio Brasileiro de Sistemas de Informação, Sociedade Brasileira de Computação, São Paulo - SP, p. 468–479, 2012. Disponível em: <www.lbd.dcc.ufmg.br/colecoes/sbsi/2012/0046.pdf>. Acesso em: 11 de novembro de 2012.

MORAIS, A. **Evasão cresce e dá prejuízo no PR.** fevereiro 2011. Disponível em: <http://www.gazetadopovo.com.br/vidaecidadania/conteudo.phtml?id=1096410> Acesso em:

11 de novembro de 2012.

MYATT, G. J. **Making Sense of Data: A Practical Guide to Exploratory Data Analysis and Data Mining**. 1st. ed. [S.l.]: Wiley-Interscience, 2006. ISBN 978-0470074718.

MYATT, G. J.; JOHNSON, W. P. **Making Sense of Data II: A Practical Guide to Exploratory Data Analysis and Data Mining.** 1st. ed. [S.l.]: Wiley, 2009. ISBN 978-0470222805.

OBSIVAC, T. et al. **Predicting drop-out from social behaviour of students.** In: YACEF, K. et al. (Ed.). Proceedings of the 5th International Conference on Educational Data Mining. [S.l.: s.n.], 2012. p. 103–109.

ORACLE. **Oracle® Data Mining Concepts 11*g* Release 1 (11.1).** Novembro 2012. Disponível em <http://docs.oracle.com/cd/B28359_01/datamine.111/b28129/index.htm> Acesso em: 11 de novembro de 2012.

QUINLAN, J. R. **Induction of Decision Trees.** In: Machine Learning 1.Kluwer, Boston, 1986. p. 81-106.

SILVA FILHO, R. L. L.; MOTEJUNAS, P. R., HIPOLITO, O.; LOBO, M. B. C. M. **A evasão no ensino superior brasileiro.** Cad. Pesqui., São Paulo, v.37, n.132, p.641-659, 2007.

SOMERVUO, P.; KOHONEN, T. **Self-organizing maps and learning vector quantization for feature sequences.** Neural Processing Letters, Springer Netherlands, v. 10, p. 151–159, 1999. ISSN 1370-4621. 10.1023/A:1018741720065. Disponível em: <http://dx.doi.org/10.1023/A:1018741720065>. Acesso em: 11 de novembro de 2012.

SOUZA, I. M. P. de. **A Qualidade do Ensino Público Brasileiro.** Abril 2011. <http://www.pedagogiaaopedaletra.com/posts/a-qualidade-do-ensino-publico-brasileiro/>. Acesso em: 11 de novembro de 2012.

SUMATHI, S.; SIVANANDAM, S. N. **Introduction to data mining and its applications.** In: . [S.l.]: Studies in Computational Intelligence, 2006.

SYMEONIDIS, A. L.; MITKAS, P. A. **Agent Intelligence Through Data Mining** (Multiagent Systems, Artificial Societies, and Simulated Organizations). Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2005. ISBN 0387243526.

TICKLE, A. et al. **The truth will come to light: directions and challenges in extracting the knowledge embedded within trained artificial neural networks.** Neural Networks, IEEE Transactions on, v. 9, n. 6, p. 1057 –1068, nov 1998. ISSN 1045-9227.

UNIPAMPA. **Projeto Institucional.** Agosto 2009. Disponível em: <www.unipampa.edu.br/portal/arquivos/PROJETO_INSTITUCIONAL_16_AG0_2009.pdf>. Acesso em: 11 de novembro de 2012.

WIKIPEDIA. **Relation (database).** Disponível em: <http://en.wikipedia.org/wiki/Relation_(database)>. Acesso em: 28 de novembro de 2012.

_____. **Cohen's kappa.** Disponível em: <http://en.wikipedia.org/wiki/Cohen's_kappa>. Acesso em: 28 de novembro de 2012.

_____. **Apriori Algorithm.** Disponível em: <http://en.wikipedia.org/wiki/Apriori_algorithm>. Acesso em: 28 de novembro de 2012.

WITTEN, E. F. I. H.; HALL, M. A. **Data Mining: Practical machine learning tools and techniques.** 3rd. ed. [S.l.]: Morgan Kaufmann, 2011. 628p. ISBN 978-0-12-374856-0.

WU, X. et al. **Top 10 algorithms in data mining.** Knowl. Inf. Syst., Springer-Verlag New York, Inc., New York, NY, USA, v. 14, p. 1–37, December 2007. ISSN 0219-1377.

ANEXO A - Comparativo de algoritmos de aprendizado

Tabela 25 – Comparativo dos algoritmos de aprendizado

	Árvores de Decisão	Redes Neurais	Naïve Bayes	kNN	SVM	Rule-learners
Exatidão em geral	**	***	*	**	****	**
Velocidade de aprendizado com respeito ao número de atributos e instâncias	***	*	****	****	*	**
Velocidade da classificação	****	****	****	*	****	****
Tolerância com dados faltantes	***	*	****	*	**	**
Tolerância com atributos irrelevantes	***	*	**	**	****	**
Tolerância com atributos redundantes	**	**	*	**	***	**
Tolerância com atributos altamente interdependentes	**	***	*	*	***	**
Utilização de atributos discretos, binários e contínuos	****	*** (não discretos)	*** (não contínuos)	*** (não diretamente discretos)	** (não discretos)	*** (não diretamente contínuos)
Tolerância a ruídos	**	**	***	*	**	*
Tratamento do risco de overfitting	**	*	***	***	**	**
Tentativa de aprendizado incremental	**	***	****	****	**	*
Habilidade de explicação / transparência do conhecimento / classificações	****	*	****	**	*	****
Alterações de parâmetros do modelo	***	*	****	***	*	***

Fonte: (KOTSIANTIS, 2007)

ANEXO B – Regras de associação geradas para o experimento

- 2 bins

1. disciplinas_cursadas='(19.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao 58 ==> situacao=Regular 55 conf:(0.95)
2. creditos_cursados_ec='(29-inf)' disciplinas_cursadas='(19.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao 57 ==> situacao=Regular 54 conf:(0.95)
3. disciplinas_concluidas_ec='(7.5-inf)' disciplinas_cursadas='(19.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao 57 ==> situacao=Regular 54 conf:(0.95)
4. disciplinas_cursadas='(19.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 57 ==> situacao=Regular 54 conf:(0.95)
5. creditos_cursados_ec='(29-inf)' disciplinas_cursadas='(19.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 57 ==> situacao=Regular 54 conf:(0.95)
6. disciplinas_cursadas='(19.5-inf)' media_disciplinas='(4.524975-inf)' 63 ==> situacao=Regular 58 conf:(0.92)
7. creditos_cursados_ec='(29-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao 63 ==> situacao=Regular 58 conf:(0.92)
8. disciplinas_concluidas_ec='(7.5-inf)' disciplinas_cursadas='(19.5-inf)' media_disciplinas='(4.524975-inf)' 63 ==> situacao=Regular 58 conf:(0.92)
9. disciplinas_concluidas_ec='(7.5-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao 63 ==> situacao=Regular 58 conf:(0.92)
10. creditos_cursados_ec='(29-inf)' semestres_concluidos='(0.5-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 63 ==> situacao=Regular 58 conf:(0.92)

- 3 bins

1. disciplinas_concluidas_ec='(13.5-inf)' disciplinas_cursadas='(25.5-inf)' 63 ==> situacao=Regular 58 conf:(0.92)
2. disciplinas_cursadas='(25.5-inf)' diferenca_creditos='(30.5-inf)' 63 ==> situacao=Regular 58 conf:(0.92)
3. disciplinas_cursadas='(25.5-inf)' 75 ==> situacao=Regular 69 conf:(0.92)
4. creditos_cursados_ec='(49-inf)' disciplinas_cursadas='(25.5-inf)' 61 ==> situacao=Regular 56 conf:(0.92)
5. disciplinas_cursadas='(25.5-inf)' razao_creditos='(3.105125-inf)' 61 ==> situacao=Regular 56 conf:(0.92)
6. disciplinas_cursadas='(25.5-inf)' sexo=M 61 ==> situacao=Regular 56 conf:(0.92)
7. creditos_cursados_ec='(49-inf)' disciplinas_concluidas_ec='(13.5-inf)' disciplinas_cursadas='(25.5-inf)' 61 ==> situacao=Regular 56 conf:(0.92)
8. disciplinas_concluidas_ec='(13.5-inf)' disciplinas_cursadas='(25.5-inf)' diferenca_creditos='(30.5-inf)' 61 ==> situacao=Regular 56 conf:(0.92)
9. disciplinas_concluidas_ec='(13.5-inf)' disciplinas_cursadas='(25.5-inf)' repr_introd_engcomp=nao 61 ==> situacao=Regular 56 conf:(0.92)
10. disciplinas_cursadas='(25.5-inf)' repr_introd_engcomp=nao 72 ==> situacao=Regular 66 conf:(0.92)

- 4 bins

1. creditos_cursados_ec='(67-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
2. disciplinas_concluidas_ec='(20.5-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
3. diferenca_creditos='(47-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
4. creditos_cursados_ec='(67-inf)' disciplinas_concluidas_ec='(20.5-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
5. creditos_cursados_ec='(67-inf)' diferenca_creditos='(47-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
6. disciplinas_concluidas_ec='(20.5-inf)' diferenca_creditos='(47-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
7. creditos_cursados_ec='(67-inf)' disciplinas_concluidas_ec='(20.5-inf)' diferenca_creditos='(47-inf)' repr_algoritmos=nao 47 ==> situacao=Regular 46 conf:(0.98)
8. disciplinas_concluidas_ec='(20.5-inf)' repr_introd_arqcomp=nao 46 ==> situacao=Regular 45 conf:(0.98)
9. creditos_cursados_ec='(67-inf)' disciplinas_concluidas_ec='(20.5-inf)' repr_introd_arqcomp=nao 46 ==> situacao=Regular 45 conf:(0.98)
10. creditos_cursados_ec='(67-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 46 ==> situacao=Regular 45 conf:(0.98)

- 5 bins

1. disciplinas_concluidas_ec='(22.5-inf)' repr_algoritmos=nao 41 ==> situacao=Regular 41 conf:(1)
2. disciplinas_concluidas_ec='(22.5-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 40 ==> situacao=Regular 40 conf:(1)
3. creditos_cursados_ec='(92.5-inf)' repr_algoritmos=nao 39 ==> situacao=Regular 39 conf:(1)
4. disciplinas_concluidas_ec='(22.5-inf)' repr_introd_arqcomp=nao 39 ==> situacao=Regular 39 conf:(1)
5. creditos_cursados_ec='(92.5-inf)' disciplinas_concluidas_ec='(22.5-inf)' repr_algoritmos=nao 39 ==> situacao=Regular 39 conf:(1)
6. disciplinas_concluidas_ec='(22.5-inf)' repr_algoritmos=nao repr_introd_arqcomp=nao 39 ==> situacao=Regular 39 conf:(1)
7. creditos_cursados_ec='(92.5-inf)' repr_introd_arqcomp=nao 38 ==> situacao=Regular 38 conf:(1)
8. creditos_cursados_ec='(92.5-inf)' disciplinas_concluidas_ec='(22.5-inf)' repr_introd_arqcomp=nao 38 ==> situacao=Regular 38 conf:(1)
9. creditos_cursados_ec='(92.5-inf)' repr_algoritmos=nao repr_introd_arqcomp=nao 38 ==> situacao=Regular 38 conf:(1)
10. creditos_cursados_ec='(92.5-inf)' repr_algoritmos=nao repr_introd_engcomp=nao 38 ==> situacao=Regular 38 conf:(1)

- 6 bins

1. creditos_cursados_ec='(105-inf)' 35 ==> situacao=Regular 34 conf:(0.97)
2. disciplinas_concluidas_ec='(25.5-inf)' 35 ==> situacao=Regular 34 conf:(0.97)

3. diferenca_creditos='(89-inf)' 35 ==> situacao=Regular 34 conf:(0.97)
4. creditos_cursados_ec='(105-inf)' repr_introd_engcomp=nao 35 ==> situacao=Regular 34 conf:(0.97)
5. disciplinas_concluidas_ec='(25.5-inf)' repr_introd_engcomp=nao 35 ==> situacao=Regular 34 conf:(0.97)
6. diferenca_creditos='(89-inf)' repr_introd_engcomp=nao 35 ==> situacao=Regular 34 conf:(0.97)
7. disciplinas_cursadas='(40.5-inf)' 34 ==> situacao=Regular 33 conf:(0.97)
8. creditos_cursados_ec='(105-inf)' disciplinas_concluidas_ec='(25.5-inf)' 34 ==> situacao=Regular 33 conf:(0.97)
9. creditos_cursados_ec='(105-inf)' diferenca_creditos='(89-inf)' 34 ==> situacao=Regular 33 conf:(0.97)
10. creditos_cursados_ec='(105-inf)' disciplinas_concluidas_ec='(25.5-inf)' repr_introd_engcomp=nao 34 ==> situacao=Regular 33 conf:(0.97)

- 7 bins

1. diferenca_creditos='(97-inf)' 30 ==> situacao=Regular 30 conf:(1)
2. diferenca_creditos='(97-inf)' repr_introd_engcomp=nao 30 ==> situacao=Regular 30 conf:(1)
3. creditos_cursados_ec='(113-inf)' 28 ==> situacao=Regular 28 conf:(1)
4. disciplinas_concluidas_ec='(27.5-inf)' 28 ==> situacao=Regular 28 conf:(1)
5. creditos_cursados_ec='(113-inf)' repr_introd_engcomp=nao 28 ==> situacao=Regular 28 conf:(1)
6. disciplinas_concluidas_ec='(27.5-inf)' repr_introd_engcomp=nao 28 ==> situacao=Regular 28 conf:(1)
7. diferenca_creditos='(97-inf)' tipoingresso=IngressoTradicional 28 ==> situacao=Regular 28 conf:(1)
8. diferenca_creditos='(97-inf)' repr_algoritmos=nao 28 ==> situacao=Regular 28 conf:(1)
9. diferenca_creditos='(97-inf)' repr_introd_arqcomp=nao 28 ==> situacao=Regular 28 conf:(1)
10. diferenca_creditos='(97-inf)' tipoingresso=IngressoTradicional repr_introd_engcomp=nao 28 ==> situacao=Regular 28 conf:(1)

- 8 bins

1. disciplinas_concluidas_ec='(27.5-inf)' 28 ==> situacao=Regular 28 conf:(1)
2. disciplinas_concluidas_ec='(27.5-inf)' repr_introd_engcomp=nao 28 ==> situacao=Regular 28 conf:(1)
3. diferenca_creditos='(103-inf)' 26 ==> situacao=Regular 26 conf:(1)
4. disciplinas_concluidas_ec='(27.5-inf)' diferenca_creditos='(103-inf)' 26 ==> situacao=Regular 26 conf:(1)
5. disciplinas_concluidas_ec='(27.5-inf)' tipoingresso=IngressoTradicional 26 ==> situacao=Regular 26 conf:(1)
6. disciplinas_concluidas_ec='(27.5-inf)' repr_algoritmos=nao 26 ==> situacao=Regular 26 conf:(1)
7. disciplinas_concluidas_ec='(27.5-inf)' repr_introd_arqcomp=nao 26 ==> situacao=Regular 26 conf:(1)
8. diferenca_creditos='(103-inf)' repr_introd_engcomp=nao 26 ==> situacao=Regular 26 conf:(1)

9. disciplinas_concluidas_ec='(27.5-inf)' diferenca_creditos='(103-inf)' repr_introd_engcomp=nao 26 ==> situacao=Regular 26 conf:(1)
10. disciplinas_concluidas_ec='(27.5-inf)' tipoingresso=IngressoTradicional repr_introd_engcomp=nao 26 ==> situacao=Regular 26 conf:(1)

- 9 bins

1. creditos_cursados_ec='(118-inf)' 25 ==> situacao=Regular 25 conf:(1)
2. disciplinas_concluidas_ec='(28.5-inf)' 25 ==> situacao=Regular 25 conf:(1)
3. disciplinas_cursadas='(44.5-inf)' 25 ==> situacao=Regular 25 conf:(1)
4. creditos_cursados_ec='(118-inf)' disciplinas_concluidas_ec='(28.5-inf)' 25 ==> situacao=Regular 25 conf:(1)
5. creditos_cursados_ec='(118-inf)' repr_introd_engcomp=nao 25 ==> situacao=Regular 25 conf:(1)
6. disciplinas_concluidas_ec='(28.5-inf)' repr_introd_engcomp=nao 25 ==> situacao=Regular 25 conf:(1)
7. creditos_cursados_ec='(118-inf)' disciplinas_concluidas_ec='(28.5-inf)' repr_introd_engcomp=nao 25 ==> situacao=Regular 25 conf:(1)
8. diferenca_creditos='(111-inf)' 24 ==> situacao=Regular 24 conf:(1)
9. creditos_cursados_ec='(118-inf)' diferenca_creditos='(111-inf)' 24 ==> situacao=Regular 24 conf:(1)
10. disciplinas_concluidas_ec='(28.5-inf)' diferenca_creditos='(111-inf)' 24 ==> situacao=Regular 24 conf:(1)

- 10 bins

1. semestres_concluidos='(-inf-0.5]' reprovacoes_recuperadas='(-inf-0.5]' repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
2. semestres_concluidos='(-inf-0.5]' repr_calc1=sim repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
3. creditos_cursados_outros='(-inf-1]' semestres_concluidos='(-inf-0.5]' reprovacoes_recuperadas='(-inf-0.5]' repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
4. creditos_cursados_outros='(-inf-1]' semestres_concluidos='(-inf-0.5]' repr_calc1=sim repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
5. disciplinas_concluidas_outros='(-inf-0.5]' semestres_concluidos='(-inf-0.5]' reprovacoes_recuperadas='(-inf-0.5]' repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
6. disciplinas_concluidas_outros='(-inf-0.5]' semestres_concluidos='(-inf-0.5]' repr_calc1=sim repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
7. creditos_cursados_outros='(-inf-1]' disciplinas_concluidas_outros='(-inf-0.5]' semestres_concluidos='(-inf-0.5]' reprovacoes_recuperadas='(-inf-0.5]' repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
8. creditos_cursados_outros='(-inf-1]' disciplinas_concluidas_outros='(-inf-0.5]' semestres_concluidos='(-inf-0.5]' repr_calc1=sim repr_algoritmos=sim repr_alga=nao 24 ==> situacao=Evadido 24 conf:(1)
9. disciplinas_concluidas_ec='(22.5-34]' repr_algoritmos=nao 22 ==> situacao=Regular 22 conf:(1)
10. semestres_concluidos='(-inf-0.5]' repr_calc1=sim repr_algoritmos=sim repr_introd_arqcomp=sim repr_alga=nao 22 ==> situacao=Evadido 22 conf:(1)