Mineração de dados para identificação de fatores de reprovação no ensino superior

Bruna Luiza Dutra

Universidade Federal de Uberlândia
Faculdade de Computação
Bacharelado em Sistemas de Informação

Monte Carmelo - MG
2019

Bruna Luiza Dutra

Mineração de dados para identificação de fatores de reprovação no ensino superior

Trabalho de Conclusão de Curso apresentado à Faculdade de Computação da Universidade Federal de Uberlândia, Minas Gerais, como requisito exigido parcial à obtenção do grau de Bacharel em Sistemas de Informação.

Área de concentração: Ciência da Computação

Orientador: Dr. Murillo Guimarães Carneiro

Monte Carmelo - MG
2019

Dedico este trabalho a Deus e todas as pessoas que contribuíram direta ou indiretamente para o seu desenvolvimento.

Agradecimentos

Agradeço primeiramente a Deus, por ter me concedido força e saúde durante todos os anos da graduação, em especial durante o desenvolvimento deste projeto. A Ele eu devo minha gratidão.

A toda minha família, ao meu pai Luiz Carlos, minha mãe Ivone e meu irmão Gustavo, por estarem ao meu lado acreditando em mim e por todo o incentivo durante os anos de faculdade.

Em especial, agradeço minha avó Maria que nos momentos de maior aflição me apoiava e rezava por mim.

Ao meu namorado Rafael, que sempre esteve ao meu lado me motivando e ajudando nesta jornada, além disso me oferecendo conforto e ânimo nos momentos mais difíceis.

Aos meus colegas de faculdade e amigas de república que fizeram parte da minha formação.

Ao meus professores que contribuíram com a minha formação acadêmica e profissional.

Ao professor e orientador Murillo Carneiro, por toda dedicação e atenção que foram essenciais para que este trabalho fosse concluído.

Agradeço aos professores participantes da banca examinadora, pela disponibilidade de participar neste momento tão importante da mim vida.

"Entregue o seu caminho ao Senhor; confie nele, e Ele agirá."
(Salmos 37:5)

Resumo

O Brasil e diversos países vêm enfrentando o problema da evasão no ensino superior. Especificamente, as taxas de evasão dos cursos da área de computação estão entre as maiores, sendo que aproximadamente um a cada três alunos que ingressam recebe o diploma. No curso noturno de Sistemas de Informação da Universidade Federal de Uberlândia, por exemplo, um estudo recente apontou que houve turmas com taxas de evasão maior do que 70%. Entre os vários motivos para a evasão, um dos mais citados na literatura é a reprovação em disciplinas do curso. Nesse sentido, métodos de Mineração de Dados podem ser adotados para encontrar os fatores que contribuem para o aluno evadir ou reprovar, o que pode ser um ferramenta relevante para a amenização desse problema. Desse modo, partindo de uma bases de dados contendo registros técnicos, sociais e econômicos de alunos do curso de Sistemas de Informação do Campus Monte Carmelo, o objetivo deste trabalho é avaliar técnicas de classificação de dados no problema de predizer o desempenho final do aluno em relação à disciplina de Introdução à Programação de Computadores. O presente estudo também busca identificar os fatores que mais contribuem para o desempenho ruim de um aluno. Os experimentos foram realizados com a ferramenta *Weka*. Um total de seis técnicas de classificação foram consideradas, além de duas situações da base de dados: com e sem atributos faltantes. Para a comparação dos resultados foram utilizadas as métricas acurácia e medida F1. Através de testes estatísticos, é possível afirmar que o modelo de indução de regras JRIP se destacou dentre as demais técnicas. Entre os indicadores que se destacaram estão a renda per capita familiar, nota média em matemática no ensino médio, distância da cidade dos pais até a cidade da universidade, distância da localidade em que o aluno mora até o campus, tipo de escola que cursou no ensino médio e com quem o aluno reside. Espera-se que esses fatores auxiliem os gestores do curso na identificação precoce dos alunos com mais chances de reprovar e permita também o planejamento de ações a fim de reduzir as taxas de evasão.

Palavras-chave: Evasão, Mineração de dados, Classificação, Fatores de reprovação.

Abstract

Brazil and several countries are facing the problem of students dropout in the universities. Particularly, dropout rates of computer science related courses are among the highest ones, with approximately one of three students receiving the diploma. For example, recent studies showed that some classes of the Information Systems course at Federal University of Uberlândia achieved dropout rates greater than 70%. Among the several reasons for students dropout, one of the most cited in the literature is students retention. In this sense, Data Mining methods can be adopted to find indicators that contribute to student dropout or retention, which can be a relevant tool to alleviate this problem. By considering a data set of technical, social and economic features of students from the Information Systems course at Monte Carmelo Campus, this work aims at evaluating data classification techniques in the problem of predicting the students final performance regarding the discipline of Introduction to Computer Programming. The present study also seeks to identify the factors that most contribute to a student's poor performance. The experiments were performed with the *Weka* tool. A total of six classification techniques were considered, in addition to two database situations: with and without missing attributes. To compare the results we adopted two metrics: accuracy and F1. Statistical tests attested the good results of the JRIP rule induction model in comparison with the other techniques. Among the indicators that stand out are: family per capita income, average math grade in high school, distance from parents city to the university city, distance from where student lives to the campus, type of school attended in high school and with who the student resides. It is expected that such indicators can assist course managers in the early identification of students who are most likely to fail and also in the planning of actions to reduce dropout rates.

Keywords: Dropout, Data mining, Data classification, Retention factors.

Lista de ilustrações

Lista de tabelas

Lista de siglas

ARFF *Attribute relation file format*

BSI-MC Bacharelado em Sistemas de Informação da Universidade Federal de Uberlândia - Campus Monte Carmelo

FA Floresta aleatória

IPC Introdução a programação de computadores

IES Instituições de ensino superior

INEP Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira

KDD Descoberta de conhecimento em bases de dados

KNN K-vizinhos mais próximos

MLP Perceptron multicamadas

MD Mineração de Dados

MDE Mineração de Dados Educacionais

RNA Redes neurais artificiais

RIPPER Poda incremental repetida para produzir redução de erro

SEMESP Sindicato das Entidades Mantenedoras de Estabelecimentos de Ensino Superior no Estado de São Paulo

SOFTEX Associação para Promoção da Excelência do Software Brasileiro

SVM Máquina de vetores de suporte

TIC Tecnologia da Informação e Comunicação

TI Tecnologia da Informação

UFU Universidade Federal de Uberlândia

WEKA Waikato *Environment for Knowledge Analysis*

Sumário

Capítulo 1

Introdução

A política de incentivos à formação superior priorizou na última década o aumento do número de vagas no ensino superior, seja pelo aumento do número de campi avançados ou mesmo de novas Instituições de ensino superior (IES) públicas, ou ainda pelo aumento da oferta de financiamento estudantil para universidades privadas (RISTOFF, 2006). Certamente, isso traz benefício ao país, uma vez que torna mais acessível a mão de obra qualificada. Entretanto, cresceu também o número de alunos que deixam a universidade sem concluir a graduação, fenômeno denominado evasão (SANTOS; GIRAFFA, 2015).

O Brasil e diversos outros países vêm enfrentando este desafio de diminuir a taxa de evasão no ensino superior, pois os prejuízos que gera compreendem várias esferas, dentre as quais destacam-se a social, acadêmica e econômica, ocasionando a ociosidade de espaço físico e materiais, tanto das IES públicas quanto das privadas. Além disso, ampliam a falta de mão de obra qualificada no mercado de trabalho (FILHO et al., 2007). Um levantamento realizado pelo Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP), entre os anos de 2010 a 2014, ressaltou que no ano de 2010 a taxa de evasão foi cerca de 11,4% e evidenciou que em 2014 a taxa atingiu 49% (MEC, 2016). Vale destacar que a contínua reprovação nas disciplinas é um dos fatores que mais levam o aluno a evadir da instituição, pois com as reprovações os alunos perdem o estímulo de continuar os estudos, provocando a evasão (FILHO; ARAÚJO, 2017).

De acordo com uma pesquisa realizada pelo Sindicato das Entidades Mantenedoras de Estabelecimentos de Ensino Superior no Estado de São Paulo (SEMESP), foi identificado que especificamente cursos da área da computação possuem umas das maiores taxas de evasão comparado a cursos de outras áreas. Vale destacar que de acordo com a pesquisa foi constatado que a cada três alunos que ingressa no curso de Sistemas de Informação, apenas um irá concluir o curso (HOED, 2016).

Este fenômeno não é diferente nos cursos de computação da Universidade Federal de Uberlândia (UFU). Um estudo realizado no curso noturno de Sistemas de Informação constatou que a taxa de evasão da turma que ingressou no segundo semestre de 2010 chegou a 72%. Além disso realizando comparações entre alunos ingressantes, retidos, eva-

didos e diplomados de turmas de 2009 a 2013, foi observado que a taxa de evasão de modo geral chegou a 50% dos alunos que ingressaram neste mesmo período (DAMASCENO; CARNEIRO, 2018).

As IES vem adotando ações afim de ajudar os alunos e consequentemente reduzir a evasão. Exemplos incluem: acompanhamento psicológico de alunos, bolsas assistenciais, cursos de reforço em disciplinas que possuem maior taxa de reprovação, atendimento com o professor buscando tirar dúvidas da disciplina, cursos de nivelamento, monitorias onde alunos da própria instituição colaboram na aprendizagem dos colegas que possuem dificuldades. Essas ações tem sido o principal recurso adotado nas universidades, mas mesmo assim não tem conseguido efetivamente reduzir a evasão.

Uma das áreas de pesquisa que contribui para lidar com tal desafio é a Mineração de Dados (MD). A MD tem como objetivo extrair informações úteis a partir de conjuntos de dados, a fim de descobrir relações complexas existentes entre eles, gerando novos conhecimentos e consequentemente levando a novas descobertas (WITTEN et al., 2016). É uma área de estudo que vem sendo utilizada na investigação de diversos problemas, por exemplo na educação.

Especificamente neste contexto, tem-se a Mineração de Dados Educacionais (MDE), que em alguns casos também é referido como Ciência de Dados Educacionais (CDE). A MDE tem por finalidade elaborar ou adequar métodos e algoritmos já existentes na MD, para análise de dados coletados em ambientes educacionais, em especial de professores e alunos, com o intuito de obter melhor compreensão dos dados no contexto educacional. A partir de tais análises, a MDE permite ao pesquisador um entendimento melhor quanto a aprendizagem do aluno (BAKER; ISOTANI; CARVALHO, 2011).

Um grupo de docentes do curso de Bacharelado em Sistemas de Informação da Universidade Federal de Uberlândia - Campus Monte Carmelo (BSI-MC) vem lidando com o problema de reduzir a evasão e reprovação nas disciplinas, especialmente dos primeiros períodos. Nesse sentido, este trabalho visa enfrentar tal cenário a partir de uma investigação para identificação do perfil do discente reprovado/evadido. Diante deste contexto, o problema considerado neste trabalho pode ser representado através das questões a seguir. A partir de um conjunto de dados dos discentes do curso de BSI-MC, especificamente cursando disciplinas do 1° e 2° período, é possível utilizar técnicas de MD para obter os fatores importantes para identificação dos alunos com maior probabilidade de evasão/reprovação no curso? Mais do que isso, tais fatores podem auxiliar na proposição de ações efetivas de combate a reprovação e evasão?

1.1 Objetivos e desafios da pesquisa

O trabalho tem como objetivo classificar e identificar os fatores que mais contribuem para a reprovação dos discentes do BSI-MC, aplicando técnicas de MD, de modo a con-

tribuir para a elaboração e implantação de políticas e intervenções no combate à evasão. Os objetivos específicos deste trabalho são:

- Selecionar um conjunto de técnicas de MD relevantes para o problema, tomando como base outros trabalhos relacionados;
- Avaliar empiricamente os algoritmos selecionados sobre a base de dados a fim de encontrar os modelos de classificação mais eficientes;
- Analisar as informações obtidas, bem como extrair os fatores que desempenham papel importante na identificação de discentes com maior probabilidade de reprovação ou evasão.

A partir de dados econômicos, sociais e acadêmicos coletados dos próprios discentes do BSI-MC, a hipótese de pesquisa investigada neste trabalho afirma que é possível identificar os fatores que mais contribuem para a reprovação, desistência ou abandono do curso, além disso predizer o desempenho do aluno em disciplina. Espera-se que os fatores encontrados auxiliem na identificação de alunos propícios a reprovar e consequentemente evadir, bem como permitam o entendimento prático de algumas causas da reprovação/evasão do curso para a comunidade, docentes e discentes. Além disso, espera-se que os mesmos auxiliem os gestores da organização na elaboração e implantação de políticas a fim de reduzir a taxa de evasão e reprovação de alunos.

1.2 Organização da monografia

Os demais capítulos da dissertação estão organizados dessa forma:

- Capítulo 2 fornece a fundamentação teórica sobre evasão no ensino superior, mineração de dados, métricas para avaliar a classificação, ferramenta *Weka* e também descreve os trabalhos relacionados a mineração de dados no combate à evasão;
- O Capítulo 3 é composto pela descrição da base de dados, o tratamento realizado nos valores ausentes, as técnicas que serão empregadas no trabalho, bem como as configurações de parâmetros;
- O Capítulo 4 apresenta como foi realizado os experimentos, bem como os resultados obtidos. Também apresenta sumário dos melhores resultados por algoritmo e testes estatísticos, e a aplicação do algoritmo que obteve melhor performance para a predição de novas amostras;
- O Capítulo 5 apresenta as principais conclusões do trabalho e sugestões de trabalhos futuros.

Capítulo 2

Fundamentação Teórica

Este capítulo apresenta uma fundamentação teórica sobre os principais assuntos envolvidos neste trabalho e também descreve os trabalhos relacionados. Na seção 2.1 discute-se a evasão no ensino superior. Na seção 2.2 apresenta-se conceitos importantes de mineração de dados, a tarefa de classificação e os algoritmos K-vizinhos mais próximos, árvore de decisão, *naive bayes*, redes neurais artificiais e indução de regras. A seção 2.3 descreve as métricas para avaliar a eficiência da classificação. A ferramenta *Weka*, que dispõe dos métodos de investigação considerados neste trabalho, está descrita na seção 2.4. A seção 2.5 apresenta sucintamente a área de mineração de dados educacionais, e a seção 2.6 apresenta os principais trabalhos relacionados na literatura.

2.1 Evasão no ensino superior

A evasão na literatura possui várias definições. Por exemplo (GAIOSO, 2005) define como a interrupção em qualquer período do ciclo do ensino. Já (KIRA, 1998) define que é a saída ou "fuga" do aluno da instituição. Para (LOBO, 2012) existem 3 tipos de evasões:

- Evasão do curso: Ocorre quando o discente desiste do curso atual e migra para outro curso da IES;
- Evasão da instituição: Ocorre quando o estudante se transfere de IES;
- Evasão de sistema: Ocorre quando o aluno desiste do curso, mas não efetua entrada em outro curso ou IES.

No Brasil especificamente na última década a pesquisa sobre a evasão no ensino superior cresceu, sendo que anteriormente a maioria dos trabalhos científicos pesquisavam este fenômeno no ensino básico (MOROSINI et al., 2011). Segundo Filho et al. (2007) entre os anos de 2001 a 2005 a média da taxa de evasão anual das instituições no Brasil foi de aproximadamente 22%, sendo a maior taxa em IES privadas, cuja taxa ficou próximo de

26%. Já nas IES públicas a taxa foi em torno de 12%. O estudo também ressaltou que ao passar dos anos essa taxa poderia aumentar.

Além das IES's, a evasão também afeta o mercado de trabalho. Especificamente na área da computação, pela falta de mão de obra qualificada. O mercado de Tecnologia da Informação e Comunicação (TIC) vem crescendo ao longo dos anos, o que gera uma demanda crescente por profissionais da área. No entanto, as empresas estão tendo dificuldades de encontrar profissionais capacitados. De acordo com o estudo realizado pela Associação para Promoção da Excelência do Software Brasileiro (SOFTEX) em 2013, cerca de 280 mil vagas de emprego da área Tecnologia da Informação (TI) ficarão ociosas no período de 2013 a 2020 (COSTA; MARTINS, 2016).

Apesar dos problemas apresentados, são poucas as IES's brasileiras que possuem programas que propõem medidas no combate à evasão ou investigam o motivo pelo qual o aluno abandona o curso. Algumas instituições relatam que o principal motivo que influencia a evasão do aluno são recursos financeiros (FILHO et al., 2007). Porém, a partir da literatura foi possível verificar que há outros motivos que contribuem para o discente abandonar o curso.

Tinto (1975) defende que para identificar os indicadores que contribuem para a evasão, não basta apenas considerar as características/atributos individuais dos alunos como: status social, experiências do ensino médio, sexo, raça, mas sim as expectativas em relação à carreira, a motivação em estudar aquele determinado curso, a instituição em que estuda, etc. Além disso, fatores externos também podem influenciar o aluno a evadir, como por exemplo a diminuição da oferta de trabalho na área em que cursa.

Souza, Silva e Gessinger (2012) analisaram os principais trabalhos sobre evasão no ensino superior do Brasil ao longo de dez anos. Eles concluíram que entre as principais causas que contribuem para o discente evadir do curso estão a falta de condições financeiras, a interferência da família, a falta de interesse na área de estudo, a reprovação nas disciplinas que envolvem matemática, a localização da IES, a idade do discente, dentre outras. Vale ressaltar que em cursos que a nota mínima para o ingresso são baixas, o índice de evasão é maior.

Estudos e pesquisas de dados de instituições públicas e privadas Lobo (2012) identificaram os fatores que mais contribuem para evasão, dentre elas: a baixa qualidade da educação básica do Brasil, a deficiência no ensino médio, escolha precoce do curso, limitação das políticas de financiamento ao estudante, dentre outras, podendo elas afetar tanto as IES públicas quanto as privadas.

Apesar desses estudos indicarem tendências gerais de evasão, as causas dela podem ter reflexos locais, relacionadas ao próprio curso, instituição e região. Para encontrar tais indicadores é necessário uma pesquisa minuciosa, a fim de encontrar o perfil do provável discente a reprovar/evadir. Nesse sentido, o uso de mineração de dados pode ser bastante útil.

2.2 Mineração de dados

O desenvolvimento das ferramentas de coleta e armazenamento de dados, contribuíram para o crescimento do volume de dados armazenados pelas organizações. Por essa razão, analisar e avaliar esses dados tornou-se um processo não trivial, contribuindo para a popularização da MD (HAN; PEI; KAMBER, 2011). Fayyad, Piatetsky-Shapiro e Smyth (1996) define MD como a aplicação de algoritmos específicos para extrair padrões de dados.

Na literatura há discordância quanto ao significado do termo MD. Para Witten et al. (2016) é o processo de encontrar padrões existentes nos dados. Existem pesquisadores que conceituam MD como sinônimo de Descoberta de conhecimento em bases de dados (KDD), no entanto Fayyad, Piatetsky-Shapiro e Smyth (1996) e Amo (2004) discordam dessa afirmação, pois consideram que a mineração de dados é apenas uma etapa particular do processo do KDD que refere-se a um processo para descoberta de informações em banco de dados grandes e complexos.

O KDD é um processo interativo não trivial para descoberta de padrões. O processo de KDD compreende várias etapas, que abrange a seleção dos dados, pré-processamento, transformação, mineração de dados e interpretação. Na Figura 1 é possível visualizar as cinco etapas do processo do KDD definido por Fayyad, Piatetsky-Shapiro e Smyth (1996).

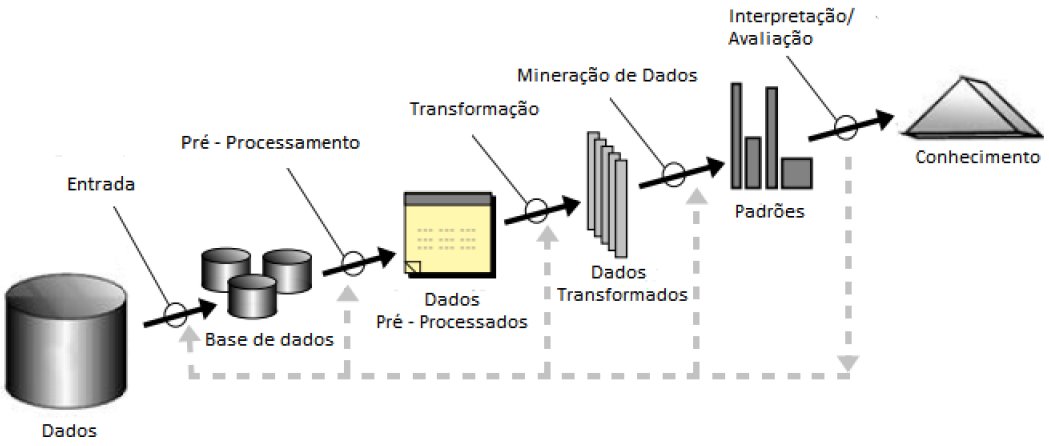

Figura 1 – Fases do processo de descoberta de conhecimento em bases de dados (KDD)

Fonte: ADAPTADA (FAYYAD; PIATETSKY-SHAPIRO; SMYTH, 1996)

- Entrada: Dados que serão processados, afim de descobrir padrões.
- Pré-processamento: Esta etapa tem como objetivo organizar os dados de forma adequada para ser analisado nas próximas fases do processo. Para isso deve verificar

se há dados redundantes, incompletos e discrepantes (*outliers*), se houver, é necessário tratá-los para não comprometer no resultado final. Pode-se realizar a união de dados, seleção manual de registros e atributos para o problema em questão. Em alguns casos, pode ser importante a participação de um especialista para realizar as operações citadas acima;

- Transformação: Visa armazenar os dados de forma apropriada para a aplicação dos algoritmos;

- Mineração: Etapa responsável em aplicar técnicas sobre a base de dados que está sendo analisada, de forma a extrair padrões e informações dos dados;

- Interpretação e avaliação: As informações descobertas na etapa de mineração são analisadas e interpretadas em relação ao objetivo pretendido. Se o resultado não for plausível, o processo pode voltar a qualquer fase anterior para refazer a atividade, e assim podendo obter melhores resultados.

Para descobrir esses padrões e extrair o conhecimento, pode-se realizar vários tipos de tarefas aplicadas individualmente ou em conjunto. Um exemplo de tarefa é a classificação (PETERMANN et al., 2006). A classificação é uma tarefa da MD que tem como objetivo predizer a classe de um objeto desconhecido, a partir de um conjunto de objetos com classes já conhecidas, buscando encontrar relacionamento entre eles (PETERMANN et al., 2006). Para encontrar relacionamentos e atribuir a classe ao objeto desconhecido, a tarefa analisa os atributos dos mesmos, verificando a similaridade, distância ou distribuição dos mesmos. Na classificação os atributos podem ser qualquer tipo de dados, porém o rótulo da classe precisa ser categórica ou discreta. Quando se tem rótulos com valores contínuos, a tarefa passa a ser denominada regressão (HAN; PEI; KAMBER, 2011).

A classificação é uma tarefa que pode ser dividida em duas etapas. Na primeira etapa, denominada treinamento, um modelo (ou classificador) é aprendido a partir dos dados de entrada. Tal modelo estabelece um mapeamento dos atributos em função dos rótulos das classes. Na segunda etapa, denominada teste, o classificador é utilizado para prever a classe de objetos cujos rótulos são desconhecidos (CARNEIRO, 2016).

2.2.1 K-vizinhos mais próximos

K-vizinhos mais próximos (KNN), é um algoritmo de classificação considerado um dos mais simples e fundamentais da MD (PETERSON, 2009). O KNN classifica um objeto desconhecido verificando a classe majoritária dentre os K objetos da base de dados de treinamento que estão mais próximos do mesmo.

O valor de K é definido pelo usuário, preferencialmente sendo um valor ímpar para não haver empate em classificações binárias. A escolha do valor que K assume é importante,

pois influencia diretamente na predição da classe do objeto desconhecido. Para encontrar o melhor valor para K, é necessário realizar vários testes, verificando qual valor obteve melhor resultado. A Figura 2 exemplifica esta situação.

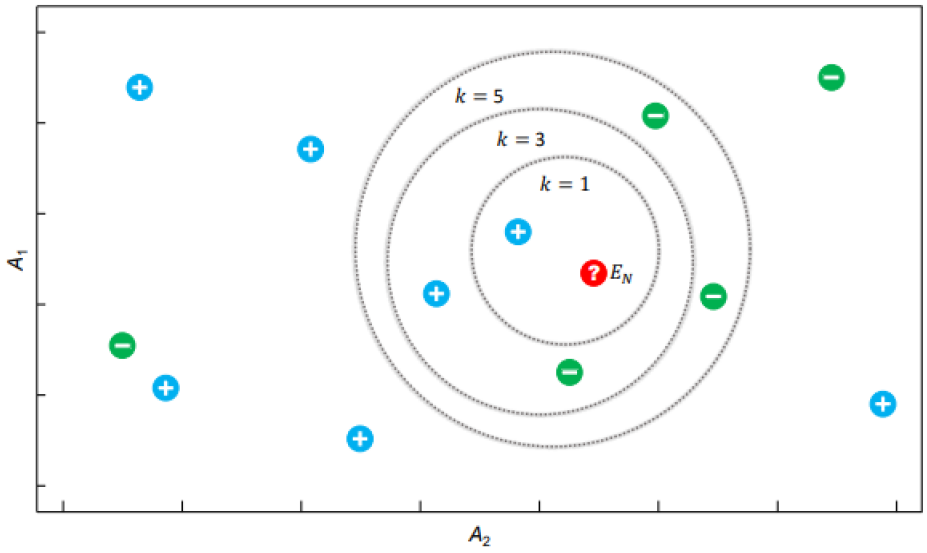

Figura 2 – Exemplo de classificação utilizando a técnica KNN

Fonte: (PARMEZAN, 2014)

Neste exemplo, o K assumiu três valores distintos, sendo o primeiro o valor um (1), o segundo o valor três (3) e o terceiro assumindo valor cinco (5). O objeto que deseja classificar é indicado pelo ponto de interrogação, podendo ser classificado com rótulo de classe na cor verde ou azul.

Para K igual a um (1) e três (3) o classificador atribuiria a classe de cor azul ao objeto desconhecido, se K assumisse o valor cinco (5), o objeto pertenceria a classe verde.

2.2.2 Árvore de decisão

A estrutura é semelhante a um fluxograma, onde os nós internos representam os atributos , as ramificações representam os resultados dos testes e por fim os nós folhas correspondem às classes. Normalmente os nós internos são representados por retângulos enquanto os nós folhas são representados por círculos. (HAN; PEI; KAMBER, 2011).

Para obter bom resultado, a árvore é gerada de acordo com a importância do atributo. O atributo mais importante é representado como a raiz da árvore, e os outros atributos são apresentados subsequencialmente do mais importante para o menos importante. Para encontrar quais atributos são importantes, é necessário calcular a Entropia e Ganho da Informação. A entropia é uma medida estatística, e quanto maior o seu valor, maior é a indecisão sobre qual classe o objeto pertence. Por exemplo se a base de dados contém

duas classes, e a metade dos objetos estão numa classe e a outra metade em outra classe, então a entropia terá valor máximo. Mas, se todas os objetos estão numa mesma classe, então a entropia terá valor zero. Calculado as entropias dos atributos, realiza-se o cálculo do ganho da informação. O ganho da informação é diferença da entropia do nó pai com os nós filhos, sendo que o atributo que obtiver maior ganho é o atributo ideal. Em cada iteração do algoritmo, será escolhido o atributo que tiver o maior ganho. O algoritmo finaliza quando todos os nós contiver nós folhas (AMO, 2004). A Figura 3 exemplifica uma árvore de decisão, a partir da análise dos atributos o algoritmo classifica o objeto com rótulo de classe sim ou não. Há diferentes classificadores baseados em árvore de decisão como J48, Floresta Aleatória, apresentados nas próximas seções (GIASSON et al., 2013).

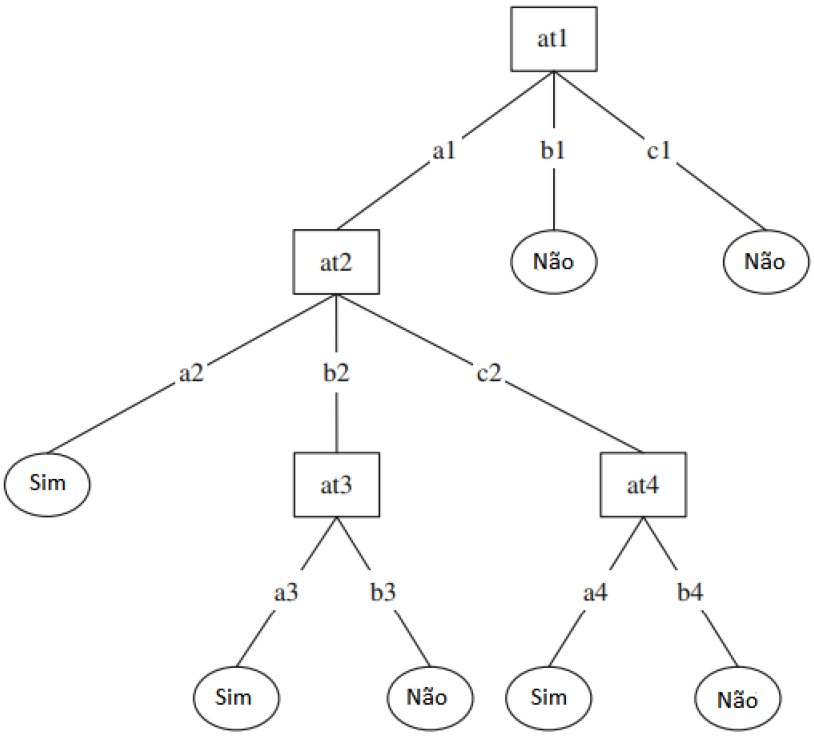

Figura 3 – Exemplo de classificação utilizando uma árvore de decisão

Fonte: ADAPTADA (KOTSIANTIS, 2013)

2.2.2.1 J48

É um algoritmo de classificação, baseado na abordagem dividir para conquistar. A construção da árvore se dá pelo cálculo do ganho da informação de cada atributo. O atributo que possuir o maior valor é o escolhido para dividir o conjunto de dados, este processo ocorre a cada iteração. Se a regra a ser testada em um determinado nó predizer a uma mesma classe, um nó folha é criado. Ao finalizar a criação da árvore, o algoritmo realiza a poda, no qual é calculado a taxa de estimativa de erro, a fim de remover os

atributos menos relevantes para a classificação. De modo geral, a intenção deste algoritmo é diminuir a quantidade de atributos utilizados na predição e ao mesmo tempo selecionar os atributos mais relevantes (LIBRELOTTO; MOZZAQUATRO, 2014).

2.2.2.2 Floresta aleatória

O algoritmo Floresta aleatória (FA) utiliza uma abordagem diferente dos outros algoritmos de árvore de decisão. Para predizer um novo objeto o modelo cria muitas árvores de decisão. Para a criação das árvores, o algoritmo utiliza a abordagem chamada de *bootstrap*, no qual o atributo é escolhido aleatoriamente e o mesmo retorna a base de dados podendo ser escolhido novamente. Após a construção das árvores, verifica-se quais possuem maior ganho de informação, ou seja, as melhores regras para a classificação, e cada uma "vota" em uma classe para o problema, no qual as árvores que possuem melhores tomadas de decisão obtém um peso sobre seu voto. A classe que possuir mais votos é a escolhida para a classificação (LORENZETT; TELÖCKEN, 2016).

2.2.3 Naive bayes

Naive Bayes classifica o objeto de acordo com probabilidades estatísticas. Baseada no teorema de *Bayes*, assume que os atributos de um objeto são independentes, ou seja, analisa os atributos separadamente. Essa suposição é chamada de independência condicional. Quando aplicado a grandes bases de dados possuem resultados de alta precisão e são rápidos em suas respostas (HAN; PEI; KAMBER, 2011).

A partir das fórmulas abaixo calcula-se a probabilidade do objeto pertencer a classe dado o atributo.

$$P(c|X) = \frac{P(X|c) * P(c)}{P(X)} \tag{1}$$

No qual:

- c, representa a classe que está sendo analisada;
- X, os atributos do objeto que está sendo analisado;
- P(c|X), a probabilidade da classe dado o atributo;
- P(X|c), a probabilidade do atributo dado a classe;
- P(c), probabilidade da classe;
- P(X), probabilidade do atributo.

Para cada classe, calcula-se a probabilidade dos atributos individualmente e ao final realiza-se o produtório de todas probabilidades, como na expressão abaixo:

$$P(c|X) = P(x_1|c) * P(x_2|c) * ... * P(x_n|c) * P(c) \quad (2)$$

Posteriormente é realizado a normalização destes valores, e a classe que possuir a maior probabilidade é atribuída ao objeto.

2.2.4 Redes neurais artificiais

Redes neurais artificiais (RNA) são modelos computacionais baseados no sistema nervoso e neurônios biológicos, que tem como finalidade o reconhecimento de padrões. Para o bom desempenho, o modelo utiliza interligações de unidades de processamento chamadas de "neurônios" (HAYKIN, 2007). Basicamente uma rede neural é um conjunto de entradas e saídas conectadas, onde cada entrada contém um peso que determina a sua intensidade. O aprendizado é o ajuste dos pesos de modo a prever o rótulo da classe correta do objeto de entrada (HAN; PEI; KAMBER, 2011).

O primeiro modelo de neurônio artificial foi proposto por *Mcculloch* e *Pitts*, e foi sofrendo várias alterações ao longo dos anos. Em 1950, *Frank Rosenblatt* criou uma rede com múltiplos neurônios que foi conhecida como *Perceptron*, onde os neurônios são organizados em camadas. A entrada da rede neural normalmente está associada aos atributos dos dados, sendo por vezes referidas como camada de entrada. Os neurônios que retornam a saída da rede neural fazem parte da camada de saída. As camadas que estão entre a entrada e a saída são chamadas de camadas internas ou ocultas. O tipo mais comum de rede neural com uma ou mais camadas ocultas é o Perceptron multicamadas (MLP) (KOVÁCS, 2002).

A Figura 4 exemplifica uma rede neural com uma camada oculta. A rede é composta por uma camada de entrada, quatro neurônios ocultos e dois neurônios de saída, ou seja, 3 camadas. Basicamente a função da camada de entrada é receber as entradas e seus respectivos pesos, a camada oculta realiza o processamento destes dados recebidos e a camada de saída é responsável por ajustar o resultado final.

Para realizar os passos da MLP, há dois passos básicos denominados *Feedforward* e *Backpropagation* . No *Feedforward* a propagação do sinal ocorre apenas em um sentido da entrada para a saída, ou seja, contém apenas uma direção. O *Backpropagation* realiza o recalculo dos pesos, corrigindo os pesos em todas as camadas a partir da derivada do erro, iniciando na camada de saída até a entrada (RUELA, 2012).

A RNA possui bom desempenho em bases de dados que possuem ruídos (HAN; PEI; KAMBER, 2011). No entanto, há uma dificuldade em entender como a rede realiza uma classificação, pois analisar o conjunto de pesos não é naturalmente intuitivo como alguns dos algoritmos apresentados anteriormente.

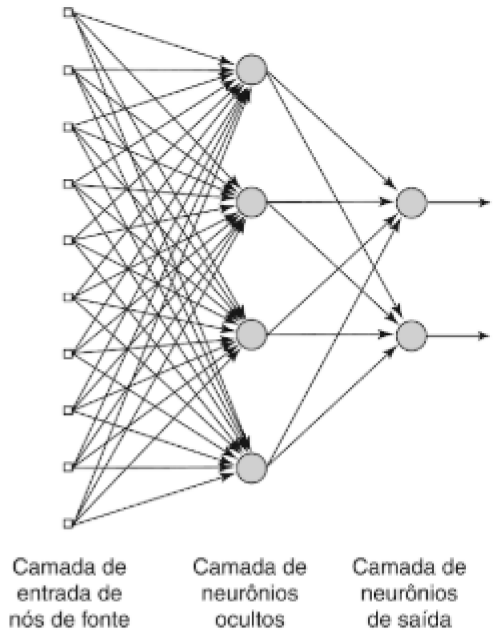

Figura 4 – Rede neural com uma camada oculta.

Fonte: (HAYKIN, 2007)

2.2.5 Indução de regras

Para classificar um novo objeto, a indução de regras analisa padrões e associações. Baseado nesses fatores o classificador gera as regras e com elas realiza a predição de novos objetos. Vale ressaltar que o modelo utiliza o método "se então" para associar dados (GOLLAPUDI, 2016).

Um dos algoritmos mais conhecidos de indução de regras é denominado Poda incremental repetida para produzir redução de erro (RIPPER). Disponível na ferramenta *Weka*, ele recebe o nome de JRIP, sendo um algoritmo de indução de regras baseado em poda para redução do erro. (COHEN, 1995).

Para problemas com duas classes o algoritmo divide a base de dados de treinamento em dois conjuntos. O primeiro conjunto é formado por objetos da classe com maior número de amostras, e com base nesses objetos descobre regras para classificar os objetos com o menor número de amostras na base dados. Para a geração de regras é utilizada a técnica chamada de cobertura sequencial.

O objetivo da cobertura sequencial, é a partir de uma regra abranger o maior número de objetos possível. Os objetos que foram atingidos pela regra são removidos da base de dados de treinamento e a regra que foi encontrada é inserida em uma lista de regras. Esses dois passos são repetidos até que não haja mais nenhum objeto na base de dados de treinamento, ou seja, a abordagem utiliza o conceito de "separar e conquistar" para

induzir regras.

Para a lista de regras não ficar extensa, o algoritmo analisa o tamanho da lista. Caso ultrapasse um tamanho definido, faz-se a poda de algumas regras. Para realizar a poda, o algoritmo utiliza um conjunto de dados para realizar as validações de quais regras serão realmente descartadas (OLIVEIRA, 2016). Ao final tem-se uma lista de regras, que serão utilizadas para predizer a classe de cada novo objeto.

A Figura 5 apresenta um exemplo da primeira iteração do algoritmo. O algoritmo funciona do seguinte modo:

- 1° passo: Encontrar uma regra e cobrir os objetos que atendem a regra;
- 2° passo: Remover os objetos cobertos e colocar a regra encontrada em uma lista de regras;
- 3° passo: Retorne ao 1° passo, enquanto houver objetos no plano.

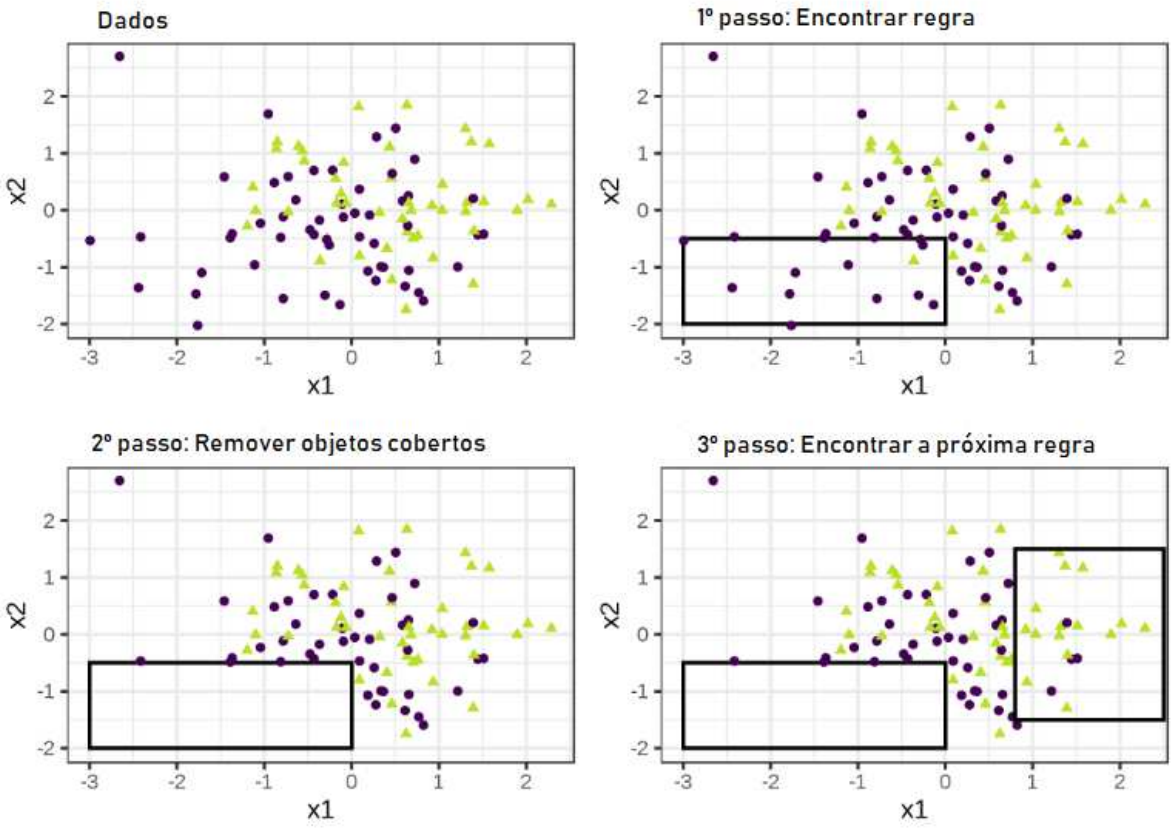

Figura 5 – Modelo de classificação JRIP

Fonte: ADAPTADA (MOLNAR, 2018)

Devido a divisão da base de dados de treinamento, o JRIP tem boa performance em base dados com classes desbalanceadas ou com presença de ruído (BENZ , 2017).

2.3 Método para a avaliação

Para avaliar se um algoritmo obteve um bom desempenho preditivo na classificação existem métricas, dentre elas acurácia e medida F1. A acurácia é a porcentagem de objetos que foram corretamente classificados pelo algoritmo. Considerada uma medida simples, mas pode haver equívoco quanto ao resultado gerado. Considerando uma base de dados desbalanceada com 90% pertencente a classe X e 10% a classe Y, se o algoritmo classificar todos objetos pertencente a classe X, a acurácia será 90%, porém não haverá qualquer indício de que este modelo possui desempenho satisfatório, ou seja, aprendeu algo relevante de fato. Por esse motivo essa métrica de classificação é mais utilizada em bases balanceadas (CASTRO; BRAGA, 2011).

Por outro lado, a medida F1 é mais utilizada em bases de dados desbalanceadas, pois o seu cálculo combina duas métricas: cobertura e precisão. A cobertura é calculada a partir do número de objetos que foram classificados como pertencente a classe positiva - verdadeiros positivos (VP), dentre todos os objetos da base de dados que realmente pertence a classe positiva P.

$$cobertura = \frac{VP}{P} \tag{3}$$

Já a precisão é o número de objetos que foram classificados como pertencente a classe positiva, dentre todos os objetos que foram classificados como classe positiva - VP e falsos positivos (FP).

$$precisão = \frac{VP}{VP + FP} \tag{4}$$

Com estas duas métricas de análise de resultados, a medida F1 é calculada a partir da fórmula:

$$F1 = \frac{2 * precisão * cobertura}{precisão + cobertura} \tag{5}$$

2.4 Ferramenta Weka

O software Waikato *Environment for Knowledge Analysis* (WEKA) é uma ferramenta de código aberto desenvolvida na Universidade de Waikato - Nova Zelândia. A primeira versão pública foi lançada em outubro de 1996, posteriormente, atualizada com todo o sistema implementado na linguagem Java. *Weka* oferece grande variedade de bibliotecas facilitando aos interessados o uso de técnicas de pré-processamento e algoritmos de MD. Com uma interface gráfica de fácil entendimento vem sendo uma das ferramentas mais utilizadas por pesquisadores de várias áreas da ciência (HALL et al., 2009).

Para a entrada dos dados possui o próprio formato de arquivos chamado Attribute relation file format *(ARFF)*, feito exclusivamente para ser utilizado no *Weka*. Todavia, podem ser utilizados outros tipos de arquivo como CSV e DAT (ALCÂNTARA, 2012).

A Figura 6 exibe a tela inicial da ferramenta, que fornece cinco opções para o usuário escolher: *Explorer*, *Experimenter*, *KnowledgeFlow*, *Workbench* e *Simple CLI*.

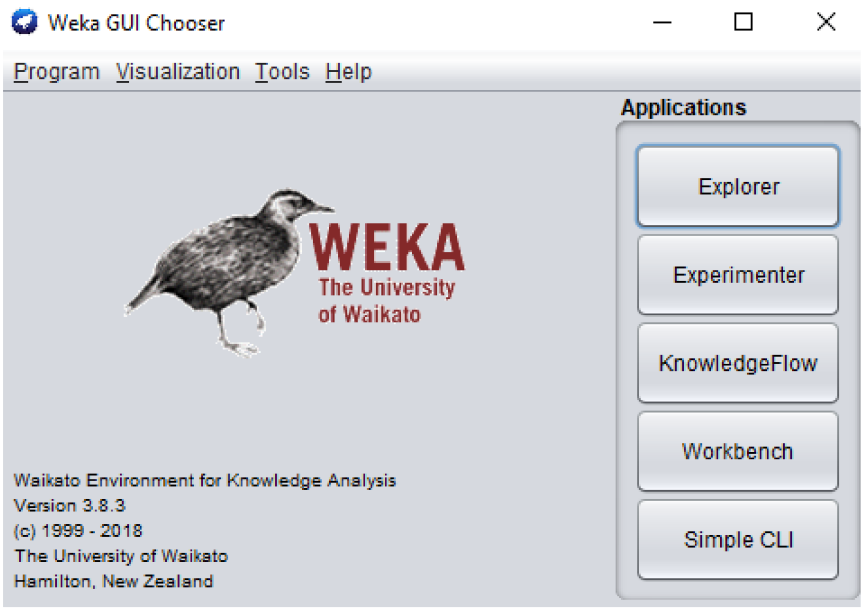

Figura 6 – Tela inicial *Weka* versão 3.6.4

Fonte: O autor

Dentre as cinco opções somente *Simple CLI* não oferece uma interface gráfica ao usuário. A aba *Explorer* é a mais utilizada dentre todas as opções, seja pelo fato de ser considerada simples de compreender e também por possuir os principais pacotes para a classificação e visualização dos dados. Além disso, também apresenta informações sobre a base de dados fornecida na entrada (FRANK et al., 2004).

2.5 Mineração de dados no combate à evasão no ensino superior

As técnicas da MD vem sendo amplamente utilizadas na educação, com o intuito de descobrir informações úteis auxiliando em diversas tarefas, tais como: na verificação da eficácia da metodologia de ensino, encontrar erros frequentes dos alunos ou atividades que são mais eficientes para o aprendizado, entre outros. Para os gestores, a MD contribui para se estabelecer medidas a fim de combater a evasão. Para os estudantes, auxilia na recomendação de atividades que podem ajudar no aprendizado. A área que trabalha

com extração de conhecimento a partir de dados educacionais é mineração de dados educacionais (MDE) (ROMERO; VENTURA, 2007).

Para Romero e Ventura (2010) existe pontos importantes que diferenciam a MD tradicional e a MDE, tais como:

- Objetivo: Cada área que a MD é utilizada possui objetivos diferentes. Utilizando a MD na educação, o objetivo é melhorar a metodologia de ensino, conduzir e auxiliar na aprendizagem do aluno, bem como o entendimento dos fatores associados ao desempenho;

- Dados: A coleta de dados no ambiente educacional pode considerar diferente variedade de dados, pois há registros de todos os discentes, e normalmente os atributos possuem relações e níveis de hierarquia entre si;

- Técnicas: As técnicas de mineração de dados podem ser utilizadas em dados educacionais, porém algumas delas precisam ser adaptadas, a fim de explorar com êxito os dados educacionais que tem características específicas, um exemplo é a não independência dos dados.

A MDE é uma área da pesquisa com crescimento considerável nos últimos anos, tanto no Brasil quanto em outros países (BAKER; ISOTANI; CARVALHO, 2011). Na literatura há diversos trabalhos onde os pesquisadores buscam encontrar fatores que influenciam na evasão no ensino superior como forma de predizer quais alunos tem mais probabilidade de evadir. Os principais trabalhos relacionados ao assunto são apresentados na próxima seção.

2.6 Trabalhos relacionados

Em Zhang et al. (2010) buscaram desenvolver um sistema que auxilia alunos e gestores de instituições, analisando se os alunos estão com risco de evasão e se a resposta for afirmativa, mostrar quais fatores que contribuíram para tal fato. Para a realização do trabalho foi utilizado o *Oracle Data* para determinar quais atributos possuem maior relevância para ser analisado, onde foi possível perceber que o aluno abandonar o curso não está relacionado com a idade ou gênero, mas sim a frequência com que ele utiliza o sistema on-line de aprendizado e a biblioteca.

Em Devasia, Vinushree e Hegde (2016), os autores trabalharam em um sistema que tem como objetivo prever o desempenho de um novo aluno, comparando os dados desse aluno com atributos de alunos já cadastrados no sistema, utilizando a técnica *Naive bayes*. Para a descoberta da informação, foram selecionados os atributos que os autores consideravam importantes, como o número de registro do estudante, as notas obtidas em cada semestre, o gênero, nível de escolaridade dos pais, renda familiar, entre outros. De

acordo com a técnica foi descoberto que os fatores como qualificação e renda da mãe estão bastantes relacionados com o desempenho dos alunos.

Utilizando uma metodologia chamada CRISP-DM, e aplicando técnicas de regressão logística e indução de regras, Calixto, Segundo e Gusmão (2017) realizaram comparações entre os estados de Ceará e Sergipe. A partir dos resultados foi possível classificar quais fatores possuem maior influência para a evasão nos dois estados, dentre elas: a idade do aluno, se a escola dispõe de laboratórios, localização das escolas e se possui transporte público. Vale ressaltar que, ao analisar os resultados individualmente, foi possível identificar que a quantidade de alunos e quantidade de computadores por turma, influenciam na evasão no estado do Ceará, diferentemente do estado de Sergipe.

Aulck et al. (2016) realizou três experimentos distintos, com os algoritmos regressão logística, k-vizinhos mais próximos, e florestas aleatórias em uma base de dados da Universidade de Washington, de alunos que estiveram matriculados entre os de 1998 e 2006, com o objetivo de encontrar os indicadores que contribuem para o aluno evadir da instituição. A partir da curva ROC [1], foi possível visualizar que os resultados dos algoritmos foram próximos, com a regressão logística obtendo resultados um pouco melhores. Dentre as variáveis que se destacaram para a predição destacam-se a média em matemática, inglês, química e psicologia, bem como o ano de matrícula e ano de nascimento.

Solomon (2018) destaca a dificuldade em prever precocemente as variáveis que interferem no desempenho do aluno, e relata a importância da MDE, para este problema. No trabalho são apresentados diferentes tipos de técnicas utilizadas para predizer os fatores que influenciam no abandono escolar, dentre eles floresta aleatória, rede neural artificial e regressão logística. A partir da pesquisa o autor concluiu que tanto o fracasso acadêmico quanto o sucesso é baseado em dados históricos, educacionais e não acadêmicos. Vale destacar, que o autor considerou a média da nota obtida pelo aluno, como principal atributo para determinar o desempenho.

Paz e Cazella (2017) investigaram perfis de alunos que tem probabilidade de evadir de uma universidade comunitária do Rio Grande do Sul no ano de 2016, utilizando a mineração de dados educacionais. Os autores utilizaram por 3 hipóteses: 1) Alunos que estão no primeiro semestre tem maior tendência de evadir, 2) Alunos que moram em cidades diferentes onde se está situada o campus tendem a reprovar nas matérias e 3) Alunos que não conseguem bolsas estudantis são mais propícios a evadir. Para verificar se as hipóteses são verídicas, os autores utilizaram a ferramenta *Weka* e o algoritmo J48, devido a possibilidade de melhor visualização dos resultados. A partir da árvore gerada foi possível constatar que alunos que estão nos primeiros semestres são mais propensos a evadir. No entanto a segunda e terceira hipóteses não foram confirmadas. Do ponto de vista dos autores os experimentos foram válidos, pois a taxa de acerto atingiu valor superior a 90%.

[1] Representação gráfica que permite avaliar o desempenho do classificador (PRATI et al., 2008).

A partir de dados do Instituto Federal de Educação, Ciência e Tecnologia do Maranhão (IFMA), Gonçalves, Silva e Cortes (2018) buscaram encontrar o perfil de estudantes evadidos utilizando técnicas de mineração de dados. No trabalho foi utilizado três algoritmos: *Naive bayes*, J48 e Máquina de vetores de suporte (SVM). Os dados analisados tinham informações do histórico, dados pessoais e socioeconômicos dos alunos. No desenvolvimento, os dados foram pré-processados, de modo que dados duplicados ou nulos fossem removidos da base de dados. Ao final a base de dados continha 40 atributos. A partir dos experimentos realizados verificou-se que os algoritmos obtiveram bons resultados. Além disso foi possível visualizar que o tempo de curso e o coeficiente de rendimento do aluno foram atributos importantes para a classificação, no entanto o J48 também considerou importante o tempo que o aluno está no curso, havendo maior probabilidade de evasão se o tempo for inferior a 3 semestres.

Alban e Mauricio (2019) fizeram uma revisão sistemática da literatura sobre os estudos de evasão na mineração de dados publicados entre os anos de 2006 a 2018. Os autores analisaram 67 documentos e buscaram identificar quais técnicas de pré- processamento, atributos, algoritmos e ferramentas estão sendo mais utilizados. As técnicas que se destacaram para pré-processamento foi a discretização, normalização e limpeza. Em relação aos atributos os fatores pessoais são os principais motivos do abandono a qual se destaca a idade, sexo, nota no vestibular, renda familiar e profissão dos pais. O algoritmo mais utilizado é Árvore de Decisão (78%), seguida por Redes Neurais e SVM. As técnicas mais precisas são os classificadores de árvore de decisão como o C4.5. Finalmente, a ferramenta mais utilizada para a predição é o *Weka*.

A abordagem utilizada nos artigos são similares, diferenciando-se principalmente os algoritmos aplicados e atributos analisados. Neste trabalho pretende-se identificar as variáveis que mais contribuem para o aluno reprovar/evadir no curso Bacharelado em Sistemas de Informação do Campus Monte Carmelo. Além disso tais variáveis serão analisadas na classificação do status (aprovado/reprovado) de novos alunos, de modo que, se comprovada a eficiência, poderão auxiliar os gestores a proporem ações para mitigar tal problema.

Capítulo 3

Materiais e Métodos

Este capítulo apresenta o conjunto de dados trabalhado, o tratamento de valores ausentes realizado na base de dados, os algoritmos utilizados no desenvolvimento bem como seleção de parâmetros configurados.

3.1 Base de dados

A base de dados é composta por dados de alunos do curso de Bacharelado em Sistemas de Informação da Universidade Federal de Uberlândia - Campus Monte Carmelo, compreende de 98 amostras e 16 atributos. A classe é a nota de uma disciplina do 1° período do curso chamada Introdução a programação de computadores (IPC), se o aluno obteve nota inferior a 60 então o mesmo pertence a classe 0 (reprovado), senão pertence a classe 1 (aprovado).

Vale ressaltar que 69 amostras pertencem a classe 0 “reprovado” e 29 a classe 1 “aprovado”. Cerca de 69.69% da base de dados pertence somente a uma classe, ou seja, há muito mais amostras de uma classe na base de dados o que torna a base desbalanceada.

Na Tabela 1 é apresentado a descrição dos atributos que compõe o conjunto de dados que será estudado neste projeto, na qual a primeira coluna exibe os atributos e a segunda coluna o tipo dos mesmos.

Tabela 1 – Descrição dos atributos da base de dados

Atributo	Tipo
Gênero	Nominal
Idade	Numérico discreto
Etnia	Nominal
Situação de trabalho	Nominal
Estado civil	Nominal
Tipo escola no ensino médio	Nominal
Média matemática no ensino médio	Numérico contínuo
Com quem mora	Nominal
Distância do bairro até o campus	Numérico contínuo
Distância da cidade dos pais até Monte Carmelo	Numérico contínuo
Situação dos pais	Nominal
Família Beneficiária de programa social	Nominal
Renda per capita	Numérico contínuo
Tipo ingresso	Nominal
Modo de transporte para UFU	Nominal
Nota na disciplina IPC	Nominal

3.2 Tratamento de valores ausentes

No conjunto de dados trabalhado havia 59 valores ausentes, total de 3.90% de valores dos atributos da base de dados, composta por 1511 valores. Vale destacar que 47 objetos possuem valores ausentes. Além disso dentre os 15 atributos, 8 possuem pelo menos um valor ausente, ressaltando que apenas a soma dos atributos nota média em matemática no ensino médio e a distância do local onde o aluno mora até o campus em que estuda, atingiu total de 40 valores ausentes.

Para o tratamento destes atributos, o estado da arte apresenta vários tipos de abordagens que podem ser empregados a fim de tratar estes dados faltantes. O método empregado neste trabalho foi a substituição dos dados ausentes pela média dos atributos, um dos métodos mais utilizados na literatura (GARCÍA-PEÑA; ARCINIEGAS-ALARCÓN; BARBIN, 2014). No qual é calculado o valor médio de cada atributo. Ao verificar que a amostra obtém valor nulo, é atribuído o valor médio do respectivo atributo.

3.3 Desenho experimental

Para os experimentos foram utilizadas duas bases de dados, a base de dados com valores ausentes (Base Original) e outra pré-processada, em que foi feito o tratamento dos valores ausentes (Base Transformada). Optou-se por avaliar a base de dados com valores ausentes, pois os mesmos podem ter alguma relação quanto ao aluno reprovar na disciplina.

Para a execução dos experimentos foi utilizada a interface gráfica denominada *Explorer* no *Weka*, a qual foi explicada no Capítulo 2. A ferramenta *Weka* foi escolhida para a realização dos experimentos especialmente pela facilidade de lidar com base de dados que possuem atributos ausentes. Para avaliar o desempenho dos algoritmos foram analisadas duas métricas: acurácia e medida F1.

Nos experimentos, os dados foram divididos em dois conjuntos: treinamento e teste. Para a divisão foi utilizada a validação cruzada, a qual divide o conjunto de dados em K partes de tamanhos iguais. Destes, K-1 conjuntos são utilizados para treinamento e um é utilizado para teste. Assim, o algoritmo executa K vezes, de forma que cada partição seja usada para teste uma vez, o resultado final é o desempenho médio dos K testes. Neste trabalho K recebeu o valor 10.

A partir de estudos do estado da arte foi selecionado um conjunto de algoritmos de classificação mais utilizados. A fim de obter resultados satisfatórios houve estudo para a seleção dos parâmetros das técnicas, pois isso pode influenciar diretamente no resultado do modelo. Desse modo a escolha dos parâmetros foi baseado no trabalho de Carneiro e Gabriel (2018). Assim, os algoritmos e respectivos parâmetros utilizados neste trabalho são:

- KNN, tem o parâmetro número de vizinhos mais próximos K $\in \{1, 2, \ldots, 20\}$;
- J48, tem um parâmetro fator de confiança $\in \{0.05, 0.15, 0.25, 0.50, 0.75, 0.90, 0.95\}$;
- Floresta Aleatória, tem o parâmetro número de árvores $\in \{2^1, 2^2, \ldots, 2^{10}\}$.
- *Naive Bayes* nenhum parâmetro;
- *Multilayer Perceptron*, tem dois parâmetros a taxa de aprendizado inicial $\alpha \in \{0.01, 0.05, 0.1, 0.2, 0.3\}$ e o número de neurônios em cada camada oculta n $\in \{10, 20, 50, 100, 500, 1000\}$;
- JRIP tem o parâmetro número de passos de otimização $\in \{2^0, 2^1, 2^2, \ldots, 2^{11}\}$.

Na próxima seção são apresentados e discutidos os resultados dos experimentos realizados com os algoritmos e os respectivos parâmetros citados acima.

Capítulo 4

Resultados Experimentais

Neste capítulo apresentamos como foi realizado os experimentos, bem como os resultados obtidos pelos algoritmos aplicados na base de dados original e base de dados pré-processada (base transformada). Também é apresentado um sumário dos melhores resultados por algoritmo e análises estatísticas, e aplicação do algoritmo com melhor desempenho na predição de novas amostras.

4.1 Resultados por algoritmo

As seções abaixo apresentam os resultados dos experimentos com as diferentes configurações de parâmetros para cada algoritmo, aplicados às bases de dados Original e Transformada.

4.1.1 K-vizinhos mais próximos

A Tabela 2 mostra o resultado do experimento realizado com o algoritmo KNN, com K variando de 1 até 20. Verifica-se que o parâmetro é um coeficiente importante para a classificação, pois para cada valor que K recebe, os valores da acurácia e medida F1 se alteram. Observa-se que na maioria dos testes a Base Original obteve melhores resultados comparada a Base Transformada. Vale ressaltar que K recebendo o valor 8, obteve melhor resultado na Base Original com acurácia e medida F1 obtendo valores 73.46% e 72.10%, respectivamente. Já na Base Transformada, o maior valor de acurácia ocorreu com K igual a 15 atingindo cerca de 70.40%, a medida F1 obteve melhor resultado com K igual a 1 com 60.60%. Vale ressaltar neste experimento, que para alguns casos a medida F1 obteve o valor zero (0), fato que ocorreu pois o algoritmo classificou todas as amostras como pertencente somente a uma classe. Ora, esta situação decorre do fato da base de dados ser bastante desbalanceada.

Tabela 2 – Acurácia média do algoritmo KNN com diferentes variações de parâmetros

	Base Original		Base Transformada	
K	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
1	65.30	63.50	62.24	**60.60**
2	56.12	58.00	44.89	46.80
3	66.32	63.60	62.24	57.50
4	65.30	64.90	53.06	53.10
5	68.36	65.20	61.22	56.80
6	67.34	66.60	61.22	58.50
7	68.36	63.50	62.24	55.30
8	**73.46**	**72.10**	63.26	59.10
9	65.30	60.50	65.30	55.60
10	71.42	69.40	65.30	58.40
11	69.38	65.10	68.36	58.80
12	67.34	63.70	68.36	58.80
13	68.36	61.50	68.36	57.20
14	68.36	62.60	69.38	59.40
15	68.36	57.20	**70.40**	0
16	70.40	61.50	70.40	0
17	69.38	59.40	70.40	0
18	67.34	59.60	70.40	0
19	70.40	59.90	70.40	0
20	66.32	57.70	70.40	0

4.1.2 J48

A Tabela 3 exibe os resultados dos experimentos realizados com o algoritmo J48, em que foi alterado o parâmetro taxa de confiança, de modo que quanto menor o seu valor, mais podas o algoritmo irá realizar. Analisando as métricas da Base Original verifica-se que quanto menor a taxa de confiança maior é o valor da acurácia e menor é o valor da medida F1. Todavia, na Base Transformada, quanto menor a taxa de confiança maior é o valor da acurácia e da Medida F1. Portanto, visualiza-se que com a imputação de valores na base de dados e realizando mais podas, o algoritmo obteve melhor desempenho. Sendo assim, na Base Original o algoritmo obteve 70.40% de acurácia e 63.60% de medida F1, e na Base Transformada chegou a 72.44% e 62.70 % de acurácia e medida F1 respectivamente.

Tabela 3 – Acurácia média do algoritmo J48 com diferentes variações de parâmetros

	Base Original		Base Transformada	
#Confiança	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
0.05	**70.40**	59.90	**72.44**	**62.70**
0.15	67.34	60.80	72.44	62.70
0.25	64.28	62.70	66.32	59.00
0.50	63.26	62.40	56.12	55.40
0.75	63.26	**63.60**	54.08	54.30
0.90	63.26	63.60	54.08	54.30
0.95	63.26	63.60	54.08	54.30

4.1.3 Floresta aleatória

A Tabela 4 mostra os resultados dos experimentos realizados nas duas bases de dados com o algoritmo Floresta Aleatória, com alteração do parâmetro número de árvores. Nota-se que o número de árvores é um parâmetro importante para a classificação, pois com a alteração do valor do mesmo, os valores das métricas de classificação mudam, seja para um valor maior ou menor. De modo geral a Base Original obteve melhores resultados comparado a Base Transformada. Nota-se que para o melhor resultado da Base Transformada foram necessárias apenas 16 árvores, chegando a 71.42% de acurácia e 68.20% da medida F1, enquanto que na Base Original o melhor resultado foi obtido com o número de árvores igual a 1024, atingindo 74.48% de acurácia e 68.90% de medida F1. Possivelmente a imputação dos dados ausentes, foi a razão do modelo atingir resultados satisfatórios com menor número de árvores na Base Transformada, portanto, com dados ausentes o modelo precisou de um número maior de árvores para atingir melhores resultados.

Tabela 4 – Acurácia média do algoritmo FA com diferentes variações de parâmetros

	Base Original		Base Transformada	
#Árvores	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
2	60.20	61.30	56.12	57.9
4	66.32	63.60	58.16	58.70
8	67.34	63.70	65.30	63.50
16	64.28	59.80	**71.42**	**68.20**
32	69.38	63.30	67.34	60.80
64	71.42	65.70	70.40	65.00
128	71.42	65.70	69.38	62.10
256	73.46	68.20	69.38	63.30
512	73.46	67.20	70.40	64.00
1024	**74.48**	**68.90**	69.38	63.30
2048	74.48	68.90	67.34	61.50
4096	74.48	68.90	67.34	63.30
8192	74.48	68.90	67.34	63.30

4.1.4 Naive bayes

A Tabela 5 ilustra o experimento com o algoritmo *Naive Bayes* nas duas bases de dados. Na Base Original, os melhores valores em função das métricas de classificação, foram 69.38% de acurácia e 68.70% de medida F1, cerca de 12% a mais em comparação com resultado da Base Transformada. Analisando o desempenho do algoritmo na Base Transformada, é possível afirmar que foi insatisfatório obtendo valores abaixo de 60% nas duas medidas avaliadas, fato que pode ter ocorrido pela influência do mecanismo de imputação adotado. De modo geral, o algoritmo não obteve bom desempenho, talvez pelo fato de supor a independência entre os atributos no momento de predizer um novo objeto, aliado à pouca quantidade de dados disponíveis na base de dados.

Tabela 5 – Acurácia média do algoritmo NB com diferentes variações de parâmetros

Base Original		Base Transformada	
Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
69.38	68.70	57.14	54.90

4.1.5 Multilayer perceptron

A Tabela 6 mostra os testes realizados com o algoritmo MLP, no qual foi realizado vários experimentos variando a taxa de aprendizado e o número de neurônios. De modo geral o algoritmo obteve resultados satisfatórios de acurácia e medida F1 na Base Original se comparado à Base Transformada, atingindo maiores resultados em todos os testes realizados. Na Base Original, o melhor valor de acurácia e medida F1 foram obtidos com a taxa de aprendizado igual a 0.2 e número de neurônios igual a 500, atingindo 75.51% e 74.60% respectivamente. Em contrapartida o melhor valor de acurácia da Base Transformada foi 62.24% e medida F1 67.50%, valor consideravelmente menor se comparado à Base Original.

Tabela 6 – Acurácia média do algoritmo MLP com diferentes variações de parâmetros

		Base Original		Base Transformada	
#Aprendizado	#Neurônios	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
0.01	10	68.36	67.40	58.16	54.80
	20	66.32	65.80	58.16	54.80
	50	68.36	67.80	57.14	54.10
	100	68.36	67.80	57.14	54.10
	500	68.36	67.80	55.10	51.90
	1000	67.34	66.60	58.16	56.30
0.05	10	67.34	67.00	57.14	54.90
	20	66.32	66.50	54.08	51.30
	50	66.32	66.20	54.08	52.00
	100	68.36	68.20	52.04	50.60
	500	68.36	69.00	**62.24**	50.60
	1000	64.28	64.50	61.22	58.50
0.1	10	67.34	67.00	60.20	60.00
	20	67.34	67.00	56.12	54.20
	50	67.34	67.00	56.12	55.40
	100	66.32	65.80	53.06	51.30
	500	65.30	65.90	62.24	58.40
	1000	72.44	71.20	60.20	55.20
0.2	10	70.40	70.30	62.24	62.40
	20	68.36	67.80	58.16	57.50
	50	68.36	67.80	52.04	52.70
	100	68.36	67.80	55.10	54.60
	500	**75.51**	**74.60**	58.16	54.80
	1000	64.28	63.70	60.20	56.20
0.3	10	67.34	66.60	58.16	**67.50**
	20	68.36	67.80	56.12	55.90
	50	67.34	67.00	54.08	54.70
	100	66.32	66.20	57.14	55.60
	500	69.38	69.10	60.20	56.20
	1000	54.08	55.60	60.20	58.40

4.1.6 JRIP

A Tabela 7 exibe os resultados do experimento com o algoritmo de indução de regras JRIP, variando o parâmetro número de otimizações. Esse parâmetro tem como objetivo melhorar as regras geradas, assim, após a geração da lista de regras, o modelo acrescenta mais condições nas regras já existentes a fim de obter regras mais precisas. Posteriormente realiza-se a poda a partir da taxa do erro. Em geral o algoritmo fez boas classificações variando a acurácia entre 74.48% e 81.63% nas duas bases de dados, enquanto que a medida F1 variou de 73.40% a 80.70%. Destaca-se que apenas com o número de otimizações

igual a 1 a Base Transformada obteve resultados superiores em comparação com a Base Original. A Base Original obteve melhores resultados, com número de otimizações igual a 32, obtendo 81.63% de acurácia e 80.70% de medida F1. Em contrapartida, na Base Transformada o melhor desempenho alcançou 80.61% e 79.40% de acurácia e medida F1, respectivamente. Comparado aos outros algoritmos, o JRIP foi o que obteve melhores resultados, registrando taxa de acerto sempre acima de 73%, feito que nenhum algoritmo avaliado anteriormente conseguiu atingir.

Tabela 7 – Acurácia média do algoritmo JRIP com diferentes variações de parâmetros

	Base Original		Base Transformada	
#Otimização	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
1	74.48	72.90	76.53	74.70
2	76.53	75.10	74.48	73.40
4	77.55	76.00	77.55	76.00
8	79.59	78.80	77.55	76.40
16	79.59	78.80	76.53	75.10
32	**81.63**	**80.70**	77.55	76.40
64	81.63	80.70	**80.61**	**79.40**
128	80.61	79.40	79.59	78.20
256	80.61	79.40	79.59	78.20
512	80.61	79.40	77.55	76.00
1024	81.63	80.70	79.59	78.50
2048	81.63	80.70	79.59	78.50

4.2 Sumário dos resultados e análises estatísticas

A Tabela 8 sumariza os melhores resultados obtidos para cada um dos algoritmos considerados na seção anterior. De modo geral comparando os resultados da Base Original e Transformada, a maioria dos modelos obtiveram melhor desempenho na Base Original, possivelmente pela estratégia adotada na imputação dos valores ausentes o que gerou ruídos na base de dados.

Tabela 8 – Melhores resultados dos algoritmos em termos de acurácia e medida F1

	Base Original		Base Transformada	
Algoritmo	Acurácia(%)	F1(%)	Acurácia(%)	F1(%)
KNN	73.46 ± 1.78	72.10 ± 5.52	70.40 ± 0.71	60.60 ± 4.75
J48	70.40 ± 1.60	63.60 ± 7.37	72.44 ± 1.58	62.70 ± 5.60
FA	74.48 ± 3.40	68.90 ± 9.87	71.42 ± 3.53	68.20 ± 6.20
NB	69.38 ± 1.72	68.70 ± 5.00	57.14 ± 3.12	54.90 ± 5.39
MLP	75.51 ± 4.30	74.60 ± 3.87	62.24 ± 3.22	67.50 ± 3.62
JRIP	**81.63 ± 2.79**	**80.70 ± 5.47**	**80.61 ± 5.09**	**79.40 ± 1.31**

O algoritmo KNN obteve resultado satisfatório, ressaltando a diferença da acurácia e medida F1 da Base Transformada no qual a diferença foi aproximadamente 10%. O J48 também se destaca pela diferença entre a acurácia e medida F1 nas duas bases de dados. Analisando o resultado em termos de acurácia nas duas bases de dados o algoritmo J48 obteve bom desempenho, porém analisando a medida F1 ocorre o contrário. Tal fato ocorreu pois nesses casos os algoritmos classificaram praticamente todas as amostras como pertencente a mesma classe (majoritária), obtendo resultado relativamente alto de acurácia. Por outro lado a medida F1 leva em consideração a precisão e cobertura. Assim, esse tipo de predição é duramente penalizada, ocasionando em um desempenho menor.

O algoritmo Floresta Aleatória, utiliza como método de classificação um esquema de votação entre um comitê de árvores de decisão, contudo, o fato da base ser desbalanceada e obter poucos objetos comprometeu o desempenho da técnica em relação às classes menos representadas (MORAIS, 2017). Em suma, analisando os resultados gerados, o algoritmo obteve resultados intermediários. De modo geral o *Naive Bayes* obteve os piores resultados, com todas as métricas abaixo de 70%, provavelmente pelo fato deste algoritmo não explorar qualquer correlação entre os atributos dos objetos, circunstância que pode ter interferido no resultado.

Considerando o desempenho do *Multilayer Perceptron*, verifica-se que o mesmo foi mais eficiente na Base Original comparado aos resultados que o mesmo obteve na Base Transformada. Um fator que pode ter relação com o desempenho do algoritmo, é a classe majoritária ter controlado o ajuste no treinamento, o que interferiu diretamente nos resultados (FU et al., 2002).

O JRIP destacou-se entre as demais técnicas, e a razão para isso é devido a sua capacidade de lidar com dados desbalanceados. Além disso tem bom desempenho com ruídos, devido ao método empregado na execução de dividir a base de dados para a predição (CERRI; CARVALHO; COSTA, 2008). Vale destacar nos resultados a diferença pequena entre acurácia e medida F1 nas duas bases de dados, fato que poucos algoritmos conseguiram alcançar.

A fim de aprofundarmos sobre o JRIP, apresenta-se abaixo as três regras geradas pelo algoritmo para predizer se um aluno seria aprovado ou reprovado:

- Se a distância da cidade em que os pais moram for inferior a 41 km, a renda per capita familiar for menor que 766 reais e o tipo de escola no ensino médio for pública, o aluno seria reprovado;

- Se a nota média em matemática no ensino médio for menor que 83 pontos, o aluno seria reprovado;

- Se a distância da cidade em que os pais do aluno residem for inferior a 219 km e a distância do local em que o aluno mora até o campus for maior que 2.9 km, o aluno seria reprovado.

A partir das regras geradas pelo modelo, verifica-se que a distância da cidade onde moram os pais até a localidade da universidade, a renda per capita, a nota média de matemática no ensino médio, distância da casa em que o aluno mora até o campus e o tipo de escola que cursou no ensino médio, foram os atributos importantes na classificação.

A fim de examinar uma possível diferença estatística entre o desempenho dos algoritmos, foi adotado o teste estatístico de *McNemar* (DIETTERICH, 1998). Basicamente, o teste cria uma tabela de contingência a partir das predições realizadas por dois classificadores ($\hat{f}_A$ e $\hat{f}_B$) tal como representado na Tabela 9, onde n_{11} denota o número de objetos que ambos os classificadores acertaram, n_{00} número de objetos que ambos erraram, n_{01} o número de exemplos errados pelo classificador $\hat{f}_A$ mas acertados por $\hat{f}_B$, e n_{10} o número de exemplos errados pelo classificador $\hat{f}_B$ mas acertados por $\hat{f}_A$. A hipótese nula desse teste afirma que dois algoritmos possuem a mesma taxa de erro. Assim, sob a hipótese nula, o teste de *McNemar* compara a distribuição de contagens de erros esperadas (n_{01} e n_{10}) em relação às observadas baseado na distribuição χ^2.

Tabela 9 – Tabela de contingência construída pelo teste de *McNemar*.

		$\hat{f}_A$	
		correto	errado
$\hat{f}_B$	correto	n_{11}	n_{01}
	errado	n_{10}	n_{00}

Os testes estatísticos foram aplicados considerando a base de dados Original (pelo melhor desempenho preditivo observado) e a medida de desempenho F1 (por levar em consideração o desbalanceamento dos dados). A Tabela 10 apresenta o p-value obtido pelo teste de *McNemar* considerando todos os pares de algoritmos. Considerando o nível de confiança de 95% ($\alpha = 0.05$), as seguintes afirmações podem ser derivadas do teste: MLP possui taxa de erro estatisticamente menor do que J48; e JRIP possui taxa de erro estatisticamente menor do que J48 e NB. Como a base de dados é pequena, também realizamos análises com nível de significância $\alpha = 0.1$, onde pode-se afirmar, com nível de confiança de 90%, que FA possui taxa de erro estatisticamente menor do que J48 e que JRIP possui taxa de erro estatisticamente menor do que FA.

Tabela 10 – Resultados p-value do teste estatístico *McNemar*

	KNN	J48	FA	NB	MLP
J48	0.1003				
FA	1.0000	**0.0543**			
NB	0.5562	0.3268	0.3827		
MLP	0.8312	**0.0310**	1.0000	0.3447	
JRIP	0.1530	**0.0005**	**0.0961**	**0.0190**	0.3613

Para uma análise mais detalhada dos testes estatísticos, a Tabela 11 apresenta as tabelas de contingência geradas pelos testes de *McNemar* entre todos os pares de algoritmos. Pela tabela é possível observar a relação das taxas de erros de todos os pares de algoritmos. Pela tabela é possível observar o desempenho ruim do J48 frente aos outros algoritmos, bem como o bom desempenho do JRIP. Mais interessante ainda, é observar que há um potencial para combinação entre alguns algoritmos, tais como o JRIP e a MLP, que juntos erraram apenas seis objetos da base de dados. Nesse sentido, a investigação de comitês (*ensembles*) capazes de combinar eficientemente as predições desses algoritmos pode resultar em uma melhora significativa de desempenho.

Tabela 11 – Representação das tabelas de contingência entre todos os pares de algoritmos de classificação.

	KNN	J48	FA	NB	MLP
J48	52 | 10 20 | 16				
FA	65 | 08 07 | 18	54 | 19 08 | 17			
NB	57 | 11 15 | 15	52 | 16 10 | 20	60 | 08 13 | 17		
MLP	62 | 12 10 | 14	55 | 19 07 | 17	62 | 12 11 | 13	57 | 17 11 | 13	
JRIP	64 | 16 08 | 10	59 | 21 03 | 15	70 | 10 03 | 15	63 | 17 05 | 13	62 | 18 12 | 06

Por fim, é importante mencionar que como as bases de dados possuem pouca quantidade de dados, pode haver uma maior variabilidade neles o que pode ser uma limitação inclusive para os testes estatísticos. Nesse sentido, eles foram adotados aqui mais como uma forma cautelosa e prudente de conduzir as nossas análises, do que propriamente um resultado definitivo. Acreditamos que o caminho para uma análise mais coerente e precisa é realmente o acréscimo de mais objetos na base de dados, o que tem sido alvo de vários esforços do orientador deste trabalho junto à comunidade acadêmica, incluindo os próprios discentes do curso.

4.3 Aplicação em um novo conjunto de dados

O objetivo deste experimento é identificar a eficiência de um algoritmo da MD em prever se um novo aluno será reprovado ou aprovado a partir de seus atributos, considerando como treinamento a base de dados original utilizada nos experimentos acima e explicada na seção 3.1

Para realizar este experimento foi utilizado o algoritmo JRIP, pois foi o modelo que obteve melhor desempenho dentre as técnicas utilizadas neste projeto. Para a experi-

mentação foi utilizado um novo conjunto de teste contendo 10 amostras compostas pelos atributos que estão listados na Tabela 1 do Capítulo 3. Tais amostras representam dados de alunos do curso BSI-MC de outro semestre letivo, sendo 7 pertencentes à classe reprovado e 3 à classe aprovado.

Durante a execução dos testes foram alterados parâmetros do algoritmo a fim de permitir uma análise mais abrangente de seu desempenho, com o parâmetro recebendo o valor 2^0, 2^5 e 2^{10}. Para ilustrar os acertos e erros do algoritmo foi utilizada a matriz de confusão, a qual permite a visualização do desempenho do modelo. Na matriz de confusão as colunas exibem a classe que o algoritmo classificou e as linhas denotam as classes as quais as amostras realmente pertencem. A diagonal principal indica os acertos do modelo, enquanto que a diagonal secundária exibe os erros.

- JRIP com parâmetro 2^0

Tabela 12 – Matriz de confusão com número de otimizações igual 2^0

		Predito	
		Aprovado	Reprovado
Verdadeiro	Aprovado	2	1
	Reprovado	0	7

Com o parâmetro recebendo valor 1, o resultado do experimento atingiu 90% de acurácia e 89.3% de medida F1, classificando 8 alunos como reprovado e 2 como aprovados. Nesse sentido o algoritmo acertou todos alunos que foram reprovados, e classificou corretamente dois alunos aprovados. Vale destacar que o algoritmo classificou incorretamente somente uma amostra classificando-a como reprovado, porém seria aprovado. O algoritmo gerou as seguintes regras neste experimento:

  - Se a nota media da matéria matemática no ensino médio for menor que 77 pontos, o aluno seria reprovado;
  - Se a renda per capita familiar for menor que 761 reais e se o aluno não morar em república, ele seria reprovado;
  - Se a distância da cidade em que os pais moram até o campus for inferior a 219 km e distância do local em que o aluno mora até o campus for maior que 2.9 km, ele seria reprovado.

A partir das regras geradas, verifica-se que o desempenho do aluno na disciplina está fortemente relacionado ao seu desempenho em matemática no ensino médio. Ademais, verifica-se que a renda possui papel importante no desempenho do aluno, especialmente para aqueles que não moram em república. Esta é uma regra que

pode ser melhor entendida considerando as particularidades do curso. Diferente dos grandes centros, a maioria das repúblicas em Monte Carmelo-MG possuem um perfil diferente, com muito menos badalação devido às características dos alunos e da própria cidade. Além disso, esse tipo de ambiente favorece a interação e ajuda mútua entre os alunos no estudo e entendimento do conteúdo das disciplinas. Por outro lado, os alunos que não moram em república normalmente são aqueles em que a família mora na própria cidade ou em cidades vizinhas. Estes alunos costumam ter uma renda menor e uma formação básica fraca, normalmente obtida na rede pública de ensino. A terceira regra relaciona o desempenho de alunos cuja família vive na região de Monte Carmelo-MG (até 219 km) e o local de moradia desses alunos, de modo que alunos que residam a mais de 2.9 km do campus, correm grandes riscos de reprovação. De fato, estimular os alunos a morar próximo da universidade pode contribuir para a criação de uma rotina mais adequada para o estudo, com menos interrupções relacionadas à transporte e deslocamento, por exemplo. Além disso, a própria gestão da assistência estudantil poderia promover isso, incentivando bolsistas na modalidade moradia a residirem próximos à universidade.

- JRIP com parâmetro 2^5

Tabela 13 – Matriz de confusão com número de otimizações igual 2^5

		Predito	
		Aprovado	Reprovado
Verdadeiro	Aprovado	2	1
	Reprovado	2	5

- JRIP com parâmetro 2^{10}

Tabela 14 – Matriz de confusão com número de otimizações igual 2^{10}

		Predito	
		Aprovado	Reprovado
Verdadeiro	Aprovado	2	1
	Reprovado	2	5

Tanto no parâmetro recebendo o valor 32 (Tabela 13) quanto recebendo o valor 1024 (Tabela 14), o algoritmo acertou o mesmo número de amostras. Visualiza-se na matriz de confusão, que o algoritmo classificou corretamente 7 amostras, sendo 5 da classe reprovado e 2 da classe aprovado, no entanto classificou incorretamente 3 amostras, 2 pertencente a classe reprovado e 1 da classe aprovado. Nesse sentido, o modelo atingiu cerca de 70% e 71% de acurácia e medida F1, respectivamente.

No experimento realizado com parâmetro igual a 2^5, as regras geradas foram iguais àquelas discutidas na Seção 4.2. Já as regras geradas no experimento com o parâmetro igual a 2^{10}, alterou somente a primeira regra comparada às regras da Seção 4.2. A nova regra encontrada foi: se a renda per capita familiar for menor que 761 reais e a distância da cidade dos pais do aluno residem for menor que 34 Km e tipo da escola do ensino médio for pública, o aluno seria reprovado.

Analisando os resultados, verifica-se que o experimento com o parâmetro de menor valor alcançou melhor resultado. Isso aconteceu provavelmente pelo fato de que os outros dois modelos ajustaram demais as regras ao conjunto de dados treinado, e ao prever novos objetos não obtiveram bom desempenho, fenômeno denominado *overfitting*. Vale ressaltar que nos três experimentos o modelo classificou incorretamente somente uma amostra da classe aprovado, situação que ocorreu pois a amostra não seguia os padrões existentes na base de dados de treinamento.

A partir das regras geradas pelo modelo nos três experimentos, verifica-se que houve poucas alterações quanto à regra do experimento que está descrita na seção 4.2, mantendo-se a maioria dos atributos para a classificação: a distância da cidade onde moram os pais até a cidade do campus, a renda per capita familiar, a nota média em matemática no ensino médio, a distância da casa em que o aluno mora até o campus, o tipo de escola no ensino médio. E houve o acréscimo de um atributo: com quem o aluno reside.

Desse modo, considerando os valores das métricas dos experimentos em classificar se o aluno irá reprovar, o mesmo obteve bom desempenho especialmente se levarmos em conta as dificuldades inerentes aos nossos dados, tais como o tamanho da base de dados e o desbalanceamento.

Capítulo 5

Conclusão

Este trabalho apresentou um estudo relacionado à aplicação de algoritmos de classificação de dados em um conjunto de dados do curso de Sistemas de Informação do Campus Monte Carmelo com objetivo de identificar os fatores que contribuem para o aluno reprovar na disciplina de introdução à programação de computadores, uma das disciplinas mais relacionadas à evasão de discentes do curso.

Foram aplicadas seis técnicas de classificação na base de dados, a qual foi observado que o algoritmo que obteve melhor desempenho preditivo foi o JRIP. Tal fato foi constatado a partir dos resultados obtidos em termos de acurácia e medida F1 mostrados na Tabela 8, e a partir dos testes estatísticos realizados, nos quais a diferença de desempenho do algoritmo para os outros foi significativa do ponto de vista estatístico, superando três deles. Acredita-se que as outras técnicas não obtiveram a mesma eficiência, devido ao fato da base de dados ser desbalanceada e a pequena quantidade de amostras disponíveis impactou diretamente nos resultados da maioria delas.

Além disso, realizou-se a aplicação da técnica JRIP para prever novo conjunto de dados. A técnica manteve bom desempenho, chegando a alcançar 90% de medida F1, atestando assim que é promissora a utilização de algoritmos de mineração de dados para identificação precoce de alunos com maior probabilidade de reprovação/evasão.

De modo geral nos experimentos verificou-se que os fatores que mais contribuem para o aluno reprovar foram a renda per capita familiar, nota média em matemática no ensino médio, distância da cidade dos pais até a cidade do campus, distância da localidade em que o aluno mora até o campus, o tipo de escola que cursou o ensino médio e com quem reside na atualidade.

Portanto, conclui-se que a utilização de algoritmos de classificação demonstra-se propícia para caracterizar com boa precisão os alunos com maiores chances de reprovar, permitindo aos gestores de instituições identificá-los precocemente e possibilitando o planejamento de ações mais eficazes.

Para trabalhos futuros sugere-se investigar métodos de comitês (*ensemble*) a fim de obter melhores resultados, e também aplicar outras técnicas da mineração de dados, a

fim de obter novas analises. Além disso, sugere-se também a aplicação da metodologia em outros cursos da universidade, para analisar tanto os fatores específicos de cada um deles quanto àqueles em comum, de modo a contribuir decisivamente para o avanço no combate à evasão nas mais diversas esferas da Universidade Federal de Uberlândia.

Referências

ALBAN, M.; MAURICIO, D. Predicting university dropout through data mining: A systematic literature. **Indian Journal of Science and Technology**, v. 12, p. 4, 2019. Citado na página 30.

ALCÂNTARA, A. R. D. **Estudo de ferramentas baseadas no weka para mineração de dados distribuída em ambiente de grid**. 2012. Disponível em: <http://repositorio.ufla.br/jspui/handle/1/31310>. Citado na página 27.

AMO, S. D. Técnicas de mineração de dados. **Jornada de Atualização em Informática**, 2004. Citado 2 vezes nas páginas 18 e 21.

AULCK, L. et al. Predicting student dropout in higher education. **arXiv preprint arXiv:1606.06364**, 2016. Citado na página 29.

BAKER, R.; ISOTANI, S.; CARVALHO, A. Mineraçao de dados educacionais: Oportunidades para o brasil. **Brazilian Journal of Computers in Education**, v. 19, n. 02, p. 03, 2011. Citado 2 vezes nas páginas 14 e 28.

BENZ . **Sistema de apoio à detecção de fraudes em e-commerce**. 2017. Disponível em: <http://propecaut.sr.ifes.edu.br/images/stories/Gabriel_Benz.pdf>. Acesso em: 02 junho 2019. Citado na página 25.

CALIXTO, K.; SEGUNDO, C.; GUSMÃO, R. P. de. Mineração de dados aplicada a educação: um estudo comparativo acerca das características que influenciam a evasão escolar. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2017. v. 28, n. 1, p. 1447. Citado na página 29.

CARNEIRO, M. G. **Redes complexas para classificação de dados via conformidade de padrão, caracterização de importância e otimização estrutural**. Tese (Doutorado) — Instituto de Ciências da Computação (ICMC/USP), São Carlos - SP, 2016. Citado na página 19.

CARNEIRO, M. G.; GABRIEL, A. What's the next move? learning player strategies in zoom poker games. In: IEEE. **2018 IEEE Congress on Evolutionary Computation (CEC)**. [S.l.], 2018. p. 1–8. Citado na página 33.

CASTRO, C. d.; BRAGA, A. P. Aprendizado supervisionado com conjuntos de dados desbalanceados. **Rev. Controle Autom**, v. 22, n. 5, p. 441–466, 2011. Citado na página 26.

CERRI, R.; CARVALHO, A. C.; COSTA, E. de P. Classificação hierárquica de proteínas utilizando técnicas de aprendizado de máquina. In: **II Workshop on Computational Intelligence**. [S.l.: s.n.], 2008. p. 1–6. Citado na página 40.

COHEN, W. W. Fast effective rule induction. In: **Machine learning proceedings 1995**. [S.l.]: Elsevier, 1995. p. 115–123. Citado na página 24.

COSTA, M. A. M. da; MARTINS, H. C. Análise quantitativa da formação do estoque de mão de obra qualificada de profissionais na área de ti. **Revista Eletrônica de Sistemas de Informação**, v. 15, n. 1, 2016. Citado na página 17.

DAMASCENO, I.; CARNEIRO, M. Panorama da evasao no curso de sistemas de informacao da universidade federal de uberlandia: Um estudo preliminar. In: **Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)**. [S.l.: s.n.], 2018. v. 29, n. 1, p. 1766. Citado na página 14.

DEVASIA, T.; VINUSHREE, T.; HEGDE, V. Prediction of students performance using educational data mining. In: IEEE. **2016 International Conference on Data Mining and Advanced Computing (SAPIENCE)**. [S.l.], 2016. p. 91–95. Citado na página 28.

DIETTERICH, T. G. Approximate statistical tests for comparing supervised classification learning algorithms. **Neural computation**, MIT Press, v. 10, n. 7, p. 1895–1923, 1998. Citado na página 41.

FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. **AI magazine**, v. 17, n. 3, p. 37–37, 1996. Citado na página 18.

FILHO, R. B. S.; ARAÚJO, R. M. de L. Evasão e abandono escolar na educação básica no brasil: fatores, causas e possíveis consequências. **Educação Por Escrito**, v. 8, n. 1, p. 35–48, 2017. Citado na página 13.

FILHO, R. L. L. S. et al. A evasão no ensino superior brasileiro. **Cadernos de pesquisa**, SciELO Brasil, v. 37, n. 132, p. 641–659, 2007. Citado 3 vezes nas páginas 13, 16 e 17.

FRANK, E. et al. Data mining in bioinformatics using weka. **Bioinformatics**, Oxford University Press, v. 20, n. 15, p. 2479–2481, 2004. Citado na página 27.

FU, X. et al. Training rbf neural networks on unbalanced data. In: IEEE. **Proceedings of the 9th International Conference on Neural Information Processing, 2002. ICONIP'02.** [S.l.], 2002. v. 2, p. 1016–1020. Citado na página 40.

GAIOSO, N. d. L. O fenômeno da evasão escolar na educação superior no brasil. **Brasília, DF: Universidade Católica de Brasília**, 2005. Citado na página 16.

GARCÍA-PEÑA, M.; ARCINIEGAS-ALARCÓN, S.; BARBIN, D. Climate data imputation using the singular value decomposition: an empirical comparison. **Revista Brasileira de Meteorologia**, SciELO Brasil, v. 29, n. 4, p. 527–536, 2014. Citado na página 32.

GIASSON, E. et al. Avaliação de cinco algoritmos de árvores de decisão e três tipos de modelos digitais de elevação para mapeamento digital de solos a nível semidetalhado na bacia do lageado grande, rs, brasil. **Ciência Rural**, Universidade Federal de Santa Maria, v. 43, n. 11, p. 1967–1973, 2013. Citado na página 21.

GOLLAPUDI, S. **Practical machine learning**. [S.l.]: Packt Publishing Ltd, 2016. Citado na página 24.

GONÇALVES, T. C.; SILVA, J. C. da; CORTES, O. A. C. Técnicas de mineração de dados: um estudo de caso da evasão no ensino superior do instituto federal do maranhão. **Revista Brasileira de Computação Aplicada**, v. 10, n. 3, p. 11–20, 2018. Citado na página 30.

HALL, M. et al. The weka data mining software: an update. **ACM SIGKDD explorations newsletter**, ACM, v. 11, n. 1, p. 10–18, 2009. Citado na página 26.

HAN, J.; PEI, J.; KAMBER, M. **Data mining: concepts and techniques**. [S.l.]: Elsevier, 2011. Citado 5 vezes nas páginas 18, 19, 20, 22 e 23.

HAYKIN, S. **Redes neurais: princípios e prática**. [S.l.]: Bookman Editora, 2007. Citado 2 vezes nas páginas 23 e 24.

HOED, R. M. Análise da evasão em cursos superiores: o caso da evasão em cursos superiores da área de computação. **Brasília, DF: Universidade de Brasília**, 2016. Citado na página 13.

KIRA, L. F. **A evasão no ensino superior: o caso do Curso de Pedagogia da Universidade Estadual de Maringá (1992-1996)**. Tese (Doutorado) — Dissertação de Mestrado] Universidade Metodista de Piracicaba—Pós-Graduação ..., 1998. Citado na página 16.

KOTSIANTIS, S. B. Decision trees: a recent overview. **Artificial Intelligence Review**, Springer, v. 39, n. 4, p. 261–283, 2013. Citado na página 21.

KOVÁCS, Z. L. **Redes neurais artificiais**. [S.l.]: Editora Livraria da Fisica, 2002. Citado na página 23.

LIBRELOTTO, S. R.; MOZZAQUATRO, P. M. Análise dos algoritmos de mineração j48 e apriori aplicados na detecção de indicadores da qualidade de vida e saúde. **Revista Interdisciplinar de Ensino, Pesquisa e Extensão**, v. 1, n. 1, 2014. Citado na página 22.

LOBO, M. Panorama da evasão no ensino superior brasileiro: aspectos gerais das causas e soluções. **Associação Brasileira de Mantenedoras de Ensino Superior. Cadernos**, v. 25, 2012. Citado 2 vezes nas páginas 16 e 17.

LORENZETT, C. D. C.; TELÖCKEN, A. V. Estudo comparativo entre os algoritmos de mineração de dados random forest e j48 na tomada de decisão. **Simpósio de Pesquisa e Desenvolvimento em Computação (SPDC)**, v. 2, n. 1, 2016. Citado na página 22.

MEC. **Altos índices de desistência na graduação revelam fragilidade do ensino médio, avalia ministro**. 2016. Disponível em: <http://portal.mec.gov.br/component/tags/tag/32044-censo-da-educacao-superior>. Acesso em: 02/12/2019. Citado na página 13.

MOLNAR, C. Interpretable machine learning: A guide for making black box models explainable. **E-book at< https://christophm. github. io/interpretable-ml-book/>, version dated**, v. 10, 2018. Acesso em: 08/10/2019. Citado na página 25.

MORAIS, R. L. d. Uso de árvores aleatórias para classificação sensorial de arroz cozido. 2017. Citado na página 40.

MOROSINI, M. C. et al. A evasão na educação superior no brasil: uma análise da produção de conhecimento nos periódicos qualis entre 2000-2011. In: **Congresos CLABES**. [S.l.: s.n.], 2011. Citado na página 16.

OLIVEIRA, P. H. M. A. **Detecção de fraudes em cartões: um classificador baseado em regras de associação e regressão logística**. Tese (Doutorado) — Universidade de São Paulo, 2016. Citado na página 25.

PARMEZAN, A. R. S. **Predição de séries temporais por similaridade**. Tese (Doutorado) — Universidade de São Paulo, 2014. Citado na página 20.

PAZ, F.; CAZELLA, S. Identificando o perfil de evasão de alunos de graduação através da mineração de dados educacionais: um estudo de caso de uma universidade comunitária. In: **Anais dos Workshops do Congresso Brasileiro de Informática na Educação**. [S.l.: s.n.], 2017. v. 6, n. 1, p. 624. Citado na página 29.

PETERMANN, R. J. et al. Modelo de mineração de dados para classificação de clientes em telecomunicação. Pontifícia Universidade Católica do Rio Grande do Sul, 2006. Citado na página 19.

PETERSON, L. E. K-nearest neighbor. **Scholarpedia**, v. 4, n. 2, p. 1883, 2009. Citado na página 19.

PRATI, R. C. et al. Curvas roc para avaliação de classificadores. **Revista IEEE América Latina**, v. 6, n. 2, p. 215–222, 2008. Citado na página 29.

RISTOFF, D. A universidade brasileira contemporânea: tendências e perspectivas. **A universidade no Brasil: concepções e modelos**, INEP/MEC Brasília, p. 37–citation_lastpage, 2006. Citado na página 13.

ROMERO, C.; VENTURA, S. Educational data mining: A survey from 1995 to 2005. **Expert systems with applications**, Elsevier, v. 33, n. 1, p. 135–146, 2007. Citado na página 28.

_____. Educational data mining: a review of the state of the art. **IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)**, Ieee, v. 40, n. 6, p. 601–618, 2010. Citado na página 28.

RUELA, A. S. Redes neurais feedforward e backpropagation. **UFOP (http://www. decom. ufop. br/imobilis/wpcontent/uploads/2012/06/03_Feedforward-e-Backpropagation. pdf)**, 2012. Citado na página 23.

SANTOS, P. K. d.; GIRAFFA, L. Evasão na educação superior: um estudo sobre o censo da educação superior no brasil. **CLABES, III. Anais. Disponível em: http://www. alfaguia. org/www. alfa/images/ponencias/clabesIII/LT_1/ponencia_completa_200. pdf Acesso em**, v. 5, n. 03, 2015. Citado na página 13.

SOLOMON, D. Predicting performance and potential difficulties of university student using classification: Survey paper. **International Journal of Pure and Applied Mathematics**, v. 118, n. 18, p. 2703–2707, 2018. Citado na página 29.

SOUZA, C. T.; SILVA, C. da; GESSINGER, R. M. Um estudo sobre evasão no ensino superior do brasil nos últimos dez anos. In: **Congresos CLABES**. [S.l.: s.n.], 2012. Citado na página 17.

TINTO, V. Dropout from higher education: A theoretical synthesis of recent research. **Review of educational research**, Sage Publications Sage CA: Thousand Oaks, CA, v. 45, n. 1, p. 89–125, 1975. Citado na página 17.

WITTEN, I. H. et al. **Data Mining: Practical machine learning tools and techniques**. [S.l.]: Morgan Kaufmann, 2016. Citado 2 vezes nas páginas 14 e 18.

ZHANG, Y. et al. Using data mining to improve student retention in higher education: a case study. In: CITESEER. **In International Conerence on Enterprise Information Systems**. [S.l.], 2010. Citado na página 28.