Revista Brasileira de Computação Aplicada, Novembro, 2018

DOI: 10.5335/rbca.v10i3.8427
Vol. 10, Nº 3, pp. 11–20
Homepage: seer.upf.br/index.php/rbca/index

ISSN 2176-6649

ARTIGO ORIGINAL

Técnicas de mineração de dados: um estudo de caso da evasão no ensino superior do Instituto Federal do Maranhão

Tayná Costa Gonçalvez[1], Josenildo Costa da Silva[1] and Omar Andres Carmona Cortes[1]

[1]Instituto Federal do Maranhão // Departamento de Computação

*tayna.cgonçalves@gmail.com; {jcsilva, omar}@ifma.edu.br

Submetido: 01/08/2018. **Revisado:** 05/09/2018. **Aceito:** 21/09/2018.

Resumo

Este trabalho mostra que é possível extrair conhecimento útil de dados puros sobre os estudantes de graduação no IFMA, de modo a tentar entender os problemas de evasão do referido instituto. Neste artigo, o conhecimento foi modelado como um classificador capaz de identificar quais alunos são os mais propensos a abandonar o curso. Foram usado três algoritmos: Naive Bayes, Support Vector Machine e J48. Três abordagens de seleção de atributos foram também testadas: Manual, Seleção Baseada em Correlação e Ganho de Informação. Assim, baseados no entendimento do problema é possível tomar medidas na tentativa de reduzir essa evasão, como por exemplo, tentar auxiliar o possível aluno evasor antes que isso aconteça, aumentando assim o número de estudantes que se formam. Testes indicaram que os melhores resultados foram obtidos pelo J48, sendo em sua maoria através da seleção baseada em correlação.

Palavras-Chave: IFMA; J48; Mineração de Dados; Naive Bayes; SVM.

Abstract

This work shows that it is possible to extract new and potentially useful knowledge from raw data about IFMA's undergraduate students, in order to get a better understanding of the dropout issue in this institution. In this paper, the knowledge has been modeled as a classifier able to classify students as likely to drop out or not using three algorithms: Naive Bayes, Support Vector Machine and J48. Three attribute selection method were tested as well: Manual, Correlation-Based Feature Selection, and Information Gain. Based on this better understanding of the problem, it is possible to take measures in order to reduce the dropout rates, such as giving targeted support to those students who are more likely to drop out. Also, it is possible that such interventions can increase students retention, consequently increasing the number of undergraduate students. Tests indicated that J48 achieves the best results along with correlation-based feature selection.

Key words: Data Mining; IFMA; J48; Naive Bayes; SVM.

1 Introdução

Historicamente, um dos problemas educacionais mais preocupantes no Brasil é a evasão, tanto a escolar quanto a universitária. Trata-se de um problema difícil de combater, visto que suas causas são diversas e variam de um contexto educacional para outro (Lobo e Silva et al.; 2007). A evasão está presente em todos os níveis educacionais, sendo influenciada por múltiplos fatores, tais como: questões familiares, sociais e econômicas, necessitando assim de uma análise

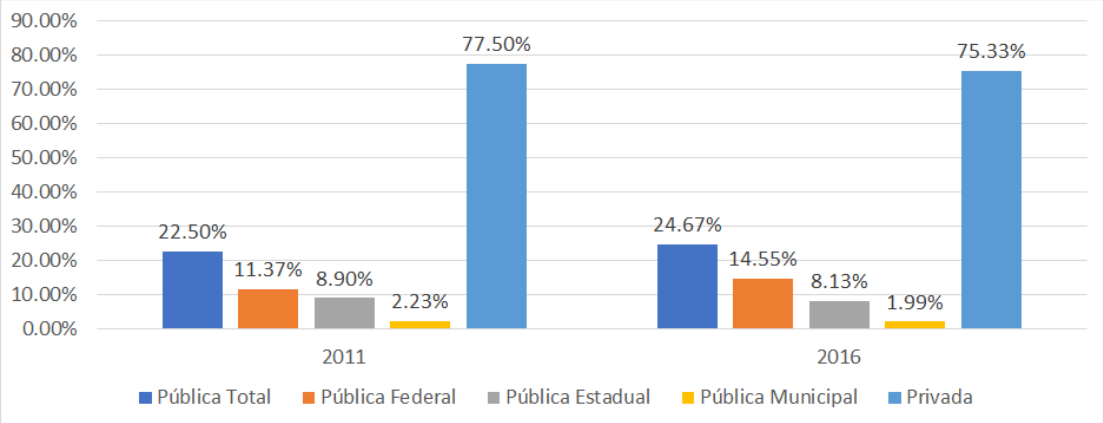

Figura 1: Concludentes em 2016 nas universidades públicas e privadas (INEP; 2018)

cuidadosa e abrangente.

De acordo com os dados do Censo da Educação Superior do Instituto Nacional de Estudos e Pesquisas Educacionais Anísio Teixeira (INEP; 2018), realizado em 2011, 5.257.898 candidatos inscreveram-se em processos seletivos para concorrer a 531.489 vagas em instituições de ensino superior públicas no Brasil. Estes números mostram que no cenário atual da educação superior pública no Brasil o número de vagas é reduzido em relação à demanda. Ainda de acordo com o censo de 2011 do INEP, no Brasil têm-se 2.365 Instituições de Ensino Superior (IES), das quais apenas 284 são públicas, sendo que 490.680 estudantes ingressaram nestas Instituições de Ensino Superior (IES) públicas em 2011, embora o número de concluintes no mesmo ano tenha sido de apenas 218.365 estudantes.

Em 2016 o número de inscritos cresceu para 6.155.369 em IES públicas, sendo a quantidade de concluintes foi de 231.572 em IES públicas como pode ser visto na Figura 1, ou seja, a proporção de não concluintes diminuiu em aproximadamente apenas dois pontos em 5 anos, sendo que o aumento de concludentes de maneira geral ocorreu nas IES federais, manteve-se mais ou menos estável nas estaduais e diminuiu nas IES municipais. E mesmo nas universidades privadas, uma evasão em torno de 25% dos alunos também é um índice alarmante para a educação superior.

Estes números mostram que os índices de evasão nas universidades públicas são alarmantes. Como consequência da evasão, têm-se o desperdício de recursos financeiros voltados para a educação, pois ao abandonar o curso o aluno continua representando, por algum tempo, custos para a IES.

À medida que as escolas e universidades passam a armazenar eletronicamente os dados acadêmicos e sócio-econômicos dos seus alunos, as técnicas de Mineração de Dados passam a constituir uma importante ferramenta na análise da evasão, considerando que estas bases de dados acadêmicos são uma potencial fonte de informação nova e útil para os gestores das instituições educacionais (Márquez-Vera et al.; 2013).

Considerando que o Instituto Federal de Educação, Ciência e Tecnologia do Maranhão (IFMA) é uma instituição que dispõe de dados eletronicamente armazenados sobre seus alunos, e que, assim como a maioria das instituições de nível superior do Brasil, sofre com o problema da evasão, busca-se neste trabalho tentar encontrar o perfil dos alunos evasores no IFMA utilizando técnicas de mineração de dados. Em outras palavras, espera-se descobrir como classificar alunos como potenciais evasores ou não.

Para esta tarefa, utilizam-se dados oriundos do Sistema Acadêmico do IFMA, correspondentes ao período de 2003 a 2013, como entrada para o processo de *Knowledge Discovery in Databases* (Descoberta de Conhecimento em Bases de Dados) (Fayyad; 1996; Makhabel; 2015), com o objetivo de investigar se esse processo é adequado para atingir um entendimento mais amplo e aprofundado sobre o fenômeno da evasão nesta Instituição. A hipótese deste estudo é que potenciais evasores, no contexto do IFMA, podem ser identificados precocemente e de forma automatizada, através de modelos computacionais construídos a partir de dados históricos oriundos do sistema acadêmico desta IES.

O restante deste trabalho é organizado da seguinte forma. Na Seção 2 são apresentados os trabalhos relacionados. Na Seção 3 é exposto o referencial teórico sobre o processo de Descoberta de Conhecimento em Bases de Dados (KDD). Na Seção 4 é feita a definição do estudo de caso de que trata este trabalho e é apresentada a base de dados utilizada, bem como é explicado de que forma foi realizado o pré-processamento dos dados. Na Seção 5 discute-se os resultados. Finalmente, na Seção 6 apresentam-se as conclusões e são sugeridos trabalhos futuros.

2 Trabalhos Correlatos

Existem na literatura muitos exemplos do uso de técnicas de Mineração de Dados com o objetivo de propor soluções para problemas recorrentes na educação superior. Obsivac et al. (2012) propõe um método para a criação de classificadores para prever a evasão estudantil, considerando tanto dados acadêmicos quanto dados relacionados aos hábitos sociais dos alunos. A conclusão foi que estudantes regulares que se comunicavam com frequência com os melhores estudantes tem uma probabilidade maior de concluir a graduação do que os que não se comunicavam com os estudantes de mais sucesso.

Romero and Ventura (2010) desenvolveram um levantamento abrangente do atual cenário da Mineração de Dados Educacionais, fazendo uma compilação dos mais relevantes estudos realizados na área até a data de sua pesquisa. Aspectos relevantes foram considerados, como os diferentes grupos de usuários e de ambientes educacionais, bem como os tipos de dados oriundos destes ambientes.

Tair and El-Halees (2012) utilizaram dados de estudantes universitários, referentes a um período de 15 anos, para criar modelos computacionais capazes de prover informações relevantes aos diretores de uma universidade, permitindo que fosse oferecido suporte aos alunos para superar problemas como notas baixas, melhorando assim o desempenho acadêmico dos estudantes.

Chuchra (2012) analisou uma base de dados contendo dados acadêmicos, residenciais e pessoais de estudantes do Departamento de MBA da Sri Sai University, com dados correspondentes ao período de um ano e meio. O objetivo era melhorar a performance acadêmica dos estudantes e diminuir o índice de reprovações.

No contexto nacional, vem crescendo o número de estudos buscando abordar o problema da evasão utilizando técnicas de mineração de dados. Marques (2014) realizou um estudo de caso utilizando dados oriundos da educação a distância do SENAI da Paraíba, com o objetivo de identificar padrões de acesso dos alunos que evadem dos cursos desta modalidade, buscando gerar regras que pudessem caracterizar o perfil de acesso desses alunos.

Modelos gerados a partir de dados oriundos de instituições educacionais carregam características que só podem ser corretamente interpretadas se analisadas dentro de um determinado contexto sócio econômico, de forma que os trabalhos existentes na literatura colaboram de maneira complementar para que seja compreendido o atual cenário da educação superior no Brasil. Este trabalho difere dos demais por analisar o fenômeno da evasão em uma instituição de nível superior de ênfase tecnológica na região nordeste do Brasil. Ainda, busca-se apresentar uma metodologia que visa oferecer à administração de instituições de ensino que se encontram no mesmo contexto a possibilidade de conduzir essa mesma avaliação, utilizando ferramentas atualmente gratuitas.

3 Evasão Universitária como Descoberta de Conhecimento

É importante ressaltar que a maior parte dos dados disponibilizados pela instituição para este estudo são dados acadêmicos, com uma pequena quantidade de dados relativos a aspectos socioeconômicos da vida dos estudantes. No entanto, estudos como o de Obsivac et al. (2012) mostram que o aspecto social da vida do estudante na universidade pode ser tão importante quanto a performance acadêmica para determinar se o estudante vai ou não concluir um curso superior.

A seguir são apresentados os conceitos básicos em termos mineração de dados e classificação, incluindo os algoritmos utilizados neste trabalho.

3.1 Descoberta de Conhecimento e Mineração de Dados

A metodologia utilizada neste trabalho é guiada pelo processo de Descoberta de conhecimento em bases de dados, do inglês *Knowledge Discovery in Databases* (KDD). De acordo com Fayyad (1996), KDD é um processo não trivial que objetiva identificar padrões válidos, novos (antes desconhecidos), potencialmente úteis e, essencialmente, compreensíveis em bases de dados. Conforme mostrado na Figura 2 (Fayyad; 1996), no processo de KDD os dados passam por várias etapas, a saber: seleção, pré-processamento, transformação e, mineração e geração de padrões, para depois estarem disponíveis como informação que pode ser visualizada e interpretada.

A mineração de dados é a análise de grandes conjuntos de dados a fim de encontrar relacionamentos inesperados e de resumir os dados de uma forma que eles sejam tanto úteis quanto compreensíveis às pessoas responsáveis pela análise dos dados (Hand et al.; 2001).

Devido a ampla área de aplicação da Mineração de Dados, seu estudo é frequentemente dividido em tarefas definidas com base no resultado esperado após o processo de mineração. Para cada tarefa há diversas técnicas e métodos a serem aplicados. Em termos de aprendizagem de máquina, estas técnicas e métodos tradicionalmente são divididas em:

- Aprendizado supervisionado (preditivo): os dados para treinamento dos modelos possuem rótulos, indicando a que classe pertencem as instâncias observadas. Novas classificações são feitas com base nas classes aprendidas na base de treinamento.
- Aprendizado não supervisionado (descritivo): não há rótulos nas instâncias da base de treinamento, portanto o objetivo é identificar classes, relações ou agrupamentos nos dados.

Neste trabalho, o objetivo é prever a situação do aluno, portanto um exemplo de aprendizagem supervisionada. Mais especificamente, este problema é definido como classificação, que será detalhado a seguir.

3.2 Problema de Classificação

Classificação consiste na tentativa de aprender a generalizar um conceito com o objetivo de identificar a classe de novos dados. A base de dados para a tarefa de classificação deve conter um ou mais atributos preditivos e um atributo classe, este último é o objetivo da classificação e deve ser do tipo discreto.

Neste tipo de problema utilizam-se técnicas de aprendizado supervisionado, pois a classe a ser identificada é conhecida. De acordo com Witten et al. (2011), a tarefa de classificação pode ser dividida em duas etapas. A primeira etapa é a construção do modelo com base no conjunto de dados de treinamento, no qual assume-se que cada tupla pertence a uma classe definida pelo rótulo do atributo classe. O modelo resultante desta etapa é representado através de regras de classificação, árvores de decisão, fórmulas matemáticas, etc., dependendo da técnica usada para construir o modelo.

Na segunda etapa, um conjunto de dados de teste

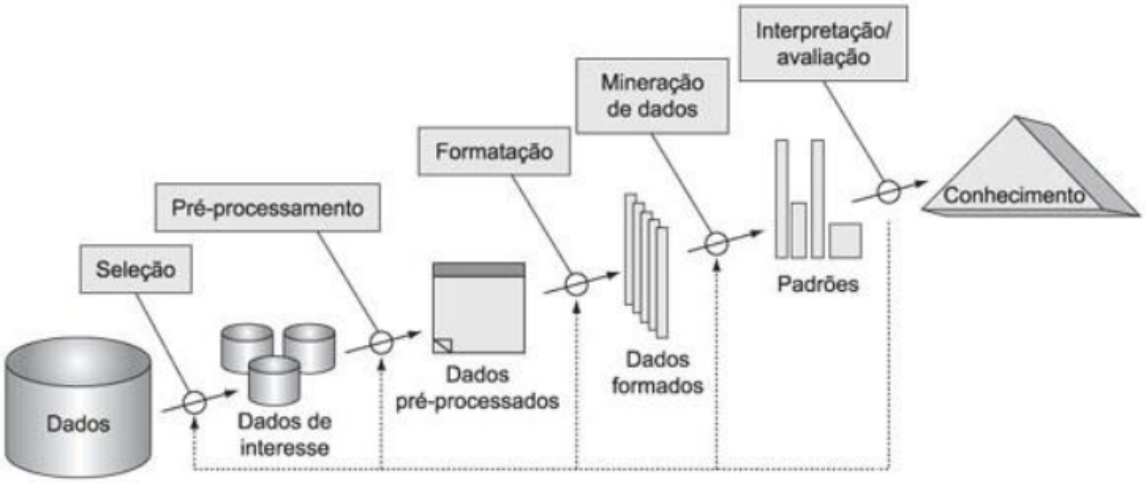

Figura 2: O processo de KDD

é submetido ao classificador anteriormente modelado com o intuito de avaliar sua qualidade. O rótulo indicado pelo classificador é comparado com o rótulo conhecido para cada instância do conjunto de teste. O modelo é avaliado segundo métricas específicas, e caso alcance um determinado limiar, pode ser usado para classificar dados novos cujos rótulos da classe são desconhecidos.

Neste trabalho, são utilizados três dos algoritmos mais conhecidos para classificação: Naive Bayes, J48 (que é uma implementação do C4.5) e*Support Vector Machines* (SVM). A escolha se baseou na larga aplicação dos mesmos, disponibilidade para uso e pelo fato de que cada algoritmo representa uma abordagem teórica diferente. A seguir apresentam-se os três algoritmos. Detalhes sobre o funcionamento de cada um deles podem ser encontrados em Faceli et al. (2011), Dangeti (2017) e Goldschmidt et al. (2015).

3.2.1 *Naive Bayes*

Naive Bayes é um classificador que ganhou popularidade nos anos 90 sendo utilizado como filtro de *spam*. A ideia do algoritmo é utilizar o teorema de Bayes, apresentada na Equação 1, para determinar a qual classe pertence uma observação, tupla ou registro.

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)} \quad (1)$$

como $P(x_1, x_2, \ldots, x_m|c_k) = P(x_1|c_k) \times P(x_2|c_k) \times \ldots \times P(x_n|c_k)$, pode-se generalizar que $P(c_k|x_1, x_2, \ldots, x_n|) \cong P(x_1|c_k) \times P(x_2|c_k) \times \ldots \times P(x_n|c_k) \times P(c_k)$, na qual x_i é um preditor (campo da base de dados) e c_k representa a classe sendo avaliada. No final das contas a saída do classificar é dada pela Equação 2, também conhecida como *Maximum a Posteriori*, a qual significa que uma observação ou tupla irá pertencer à classe que possuir a maior probabilidade.

$$y = argmax_k\ P(c_k) \times \prod P(x_i|c_k) \quad (2)$$

Segundo Li and Li (2015), o Naive Bayes apresenta as seguintes vantagens: (i) possui uma base matemática sólida; (ii) é rápido; e (iii) apesar de simples, apresenta de modo geral um bom desempenho em tarefas de classificação. Por esse motivo, este algoritmo é utilizado muitas vezes como base de comparação com outros algoritmos de classificação.

3.2.2 *SVM*

Maquinas de Vetor de Suporte, do inglês *Support Vector Machine*, é um conjunto de métodos de aprendizado supervisionado utilizados para tanto para classificação quanto para regressão. Regressão é utilizada para prever valores quantitativos (James et al.; 2013), por esse motivo, esta fora do escopo deste trabalho.

Um modelo SVM é uma representação de exemplos como pontos no espaço, mapeados de maneira que os exemplos de cada classe sejam divididos por um espaço claro que seja tão amplo quanto possível, como mostrado na Figura 3, na qual se observa a linha que divide as duas classes e as linhas pontilhadas denominadas de vetor de suporte. Essa divisão é denominada de margem (m), sendo que o objetivo da SVM é maximizar m, ou seja, maximizar a distância entre os hiperplanos do vetor de suporte.

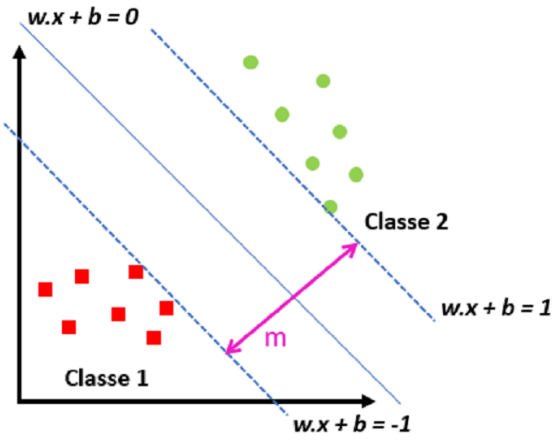

Figura 3: Classificador linear SVM

Como observado na Figura 3, a ideia no SVM é construir um hiperplano $h(x) = w \,.\, x + b$, no qual $w \,.\, x$ é o produto escalar entre os vetores w e x e $w \in X$ é o vetor normal ao hiperplano. Em duas dimensões w representa o coeficiente angular da reta. Ainda na figura, a reta principal é $w \,.\, x + b = 0$ e as linhas, ou marges, são $w \,.\, x + b = -1$ e $w \,.\, x + b = 1$, para as linhas inferior e superior, respectivamente, sendo que b é computado a partir da média de todos os vetores de suporte possíveis. Detalhes sobre como obter b podem ser vistos no trabalho de Lorena and de Carvalho (2007). Nesse contexto, um classificador pode ser construído usando a Equação 3.

$$g(x) = sgn(h(x)) = \begin{cases} +1, \text{se } w \,.\, x + b > 0 \\ -1, \text{se } w \,.\, x + b < 0 \end{cases} \quad (3)$$

Segundo Campbell (2000), usando manipulações algébricas, a margem m de separação entre objetos pode ser obtida pela minimização de $||w||$. Como não é permitido que haja dados de treinamento entre as margens, essa SVM é dita com margens rígidas. Caso o hiperplano não seja suficiente para separar as classes, então pode-se usar técnicas de aumento de dimensionalidade, assim o que não pode ser separado em 2 dimensões por uma reta, pode dependendo dos dados ser separável em 3 dimensões, e assim por diante.

Segundo Lorena and de Carvalho (2007), as SVMs são robustas diante de dados de grandes dimensões. sobre os quais outras técnicas de aprendizado obtêm classificadores super ou sub ajustados. Outra vantagem é que existe somente uma configuração ótima para a SVM em seu treinamento. Essa característica é interessante frente a técnicas como as Redes Neurais Artificiais (RNAs) (Braga et al.; 2000) que são multimodais (apresentam muitos ótimos locais, dificultando a busca) em seu treinamento. Além disso, o uso de funções Kernel na não-linearização das SVMs torna o algoritmo eficiente, pois permite a construção de simples hiperplanos em um espaço de alta dimensão de forma tratável do ponto de vista computacional (Burges; 1998). Por outro lado, a principal limitação das SVMs está na sensibilidade a escolha de seus parâmetros, pois pode levar a modelos imprecisos.

3.2.3 J48

Como já mencionado, trata-se de uma implementação do popular algoritmo C4.5 para geração de árvores de decisão, criado por Quinlan (1993) como uma extensão do seu algoritmo anterior, o ID3. Neste algoritmo o conhecimento é representado em forma de árvore de decisão, que é uma estrutura que consiste em nós rotulados com nomes de preditores ou atributos, arcos rotulados com os possíveis valores para os atributos preditivos e folhas rotuladas com as diferentes categorias de classe.

Novas instâncias são classificadas percorrendo um caminho da árvore, o que corresponde a execução de uma sequência de testes. O papel do algoritmo é escolher as regras mais importantes para compor a árvore e descartar regras que são menos adequadas. Seu funcionamento é dado por:

Dada uma base de treinamento D, o algoritmo gera uma árvore inicial usando a abordagem de dividir e conquistar, utilizando as seguintes regras: (i) Se todas as instâncias em D pertencem a mesma classe ou se a base D é muito pequena, a árvore pode ser representada como uma folha rotulada com a classe mais frequente em D; (ii) Caso contrario, é escolhido um atributo com dois ou mais valores (ou faixas de valores) para ser usado como teste. Esse atributo será o nó raiz da árvore, com uma ramificação para cada possível saída do nó. A base de dados D é então dividida em subconjuntos, um para cada possível saída do nó raiz. O mesmo procedimento é aplicado recursivamente em cada um dos subconjuntos.

Normalmente o algoritmo usa a entropia ($H(D)$) e o ganho de informação para decidir como construir a árvore. A entropia determina a impureza de um determinado conjunto de dados, sendo computada pela Equação 4, na qual p_i é a proporção do atributo ser da classe i. Já o ganho de informação, dado o atributo A, que tem um domínio v é dado pela Equação 5, na qual D_j é o subconjunto do domínio de A e $|D_j|$ é a quantidade de registros em D_j.

$$H(D) = -\sum_{i}^{k} p_i \times log_2 p_i \quad (4)$$

$$GI(D,A) = H(D) - \sum_{j=1}^{v} \frac{|D_j|}{|D|} \times H(D_j) \quad (5)$$

Assim, o atributo que possuir maior ganho de informação será utilizado em um nó da árvore. O processo continua recursivamente até que a árvore esteja construída.

A principal vantagem das árvores de decisão é ser compreensível à leitura humana, ou seja, é fácil de ser entendida pelo usuário. Além disso, pode lidar com uma variedade grande de dados (categórica, numérica e textual), não sendo sensível à dados faltantes (Bhargava et al.; 2013). Por outro lado podem apresentar as seguintes desvantagens: (i) podem gerar árvores complexas, sendo sua criação computacionalmente intensa; e, (ii) dados desbalanceados podem levar a árvores grandes que são de difícil interpretação.

4 Estudo de caso no IFMA

Este trabalho foi desenvolvido em formato de estudo de caso, considerando-se que o estudo deste caso em particular pode ser útil na investigação de aspectos mais abrangentes da evasão universitária nos institutos federais. A sequência de etapas seguidas neste estudo foram as seguintes:

i. **Identificação do Problema**

Inicialmente foi feito um levantamento sobre a evasão nas instituições de ensino superior, bem como sobre a utilização de técnicas de mineração de dados no contexto educacional. A partir do entendimento destes dois aspectos do trabalho, o problema de como prever quais alunos são mais propensos a evasão foi mapeado para a tarefa de classificação.

ii. **Adequação dos dados**

Para que a execução das tarefas de mineração fosse possível, inicialmente os dados são pré-processados, para que se tornem adequados para servir como entrada para os algoritmos de mineração.

iii. **Escolha das técnicas e algoritmos**

Na etapa de pré-processamento foram utilizadas tanto técnicas automatizadas quanto análise manual para seleção de atributos. As técnicas automatizadas utilizadas foram *Information Gain* e *Correlation Based Feature Selection*. Desta forma, as saídas da etapa de pré-processamento são três diferentes *datasets*: um oriundo de seleção manual e os demais oriundos das técnicas automatizadas de seleção. Após análise das características dos *datasets* resultantes, os seguintes algoritmos de classificação foram selecionados para serem utilizados neste trabalho: *Naive Bayes*, J48 e *Support Vector Machine*.

iv. **Escolha dos parâmetros para validação**

Para que o modelo seja considerado útil para ser aplicado ao problema real torna-se necessário avaliar a sua qualidade. Dentre os diversos critérios possíveis para avaliar a qualidade dos modelos criados por algoritmos de aprendizagem de máquina, torna-se necessário decidir quais são mais relevantes dentro do contexto desse trabalho. Após a seleção das métricas a serem utilizadas, foi realizada a comparação entre os resultados obtidos pelas diversas combinações de *datasets* e algoritmos. Esta comparação foi feita através da interface *Experimenter* do Weka (of Waikato; 2015).

v. **Interpretação dos resultados:**

É feita uma análise dos modelos obtidos, com o objetivo de avaliar até que ponto os mesmos são utilizáveis e podem ser compreendidos, para que sejam disponibilizados aos gestores do IFMA, ou seja, nesta etapa interpretam-se os padrões descobertos e, eventualmente, volta-se a qualquer um dos passos anteriores caso seja necessário.

A seguir, discutem-se em detalhes as etapas de adequação dos dados, que são etapas intrinsecamente mais trabalhosas do processo de KDD, englobando a integração, limpeza, transformação e redução dos dados.

Base de dados

Os dados utilizados neste trabalho foram extraídos do Sistema Acadêmico do IFMA. Foram fornecidos três arquivos em formato CSV (*Comma Separated Values*), contendo atributos relacionados aos mais diversos aspectos da vida dos estudantes, porém sem atributos que pudessem identificar os alunos, isto é, os dados são anônimos. Além disso, cabe-se destacar que foram utilizados apenas dados acadêmicos compostos por 3 tabelas básicas: *Pautas*, *Histórico* e *Matriculas*. A tabela Pautas possui informações sobre disciplinas oferecidas em um determinado semestre. O conceito de pauta e diferente do de disciplina, pois a pauta contém informações específicas sobre o modo como a disciplina foi oferecida em determinado semestre, como por exemplo, o professor que a ministrou e o horário O Histórico representa a relação entre matrículas e pautas, ou seja, representa cada pauta em que um aluno se matriculou ao longo da sua vida acadêmica, com as respectivas notas. Finalmente, a tabela Matriculas possui todos os dados pessoais e socioeconômicos informados pelo aluno no momento da matrícula na instituição. Apesar de contar com muitos campos relacionados ao aspecto socioeconômicos da vida do estudante, grande parte dessas informações não estava preenchida, inviabilizando seu uso. Dessa forma, os resultados aqui contidos baseiam-sem apenas nos dados acadêmicos.

Limpeza dos dados

Na limpeza busca-se eliminar inconsistências e valores errados ou incompletos, para que não influenciem no resultado dos algoritmos, bem como para reduzir a dimensão dos dados. A etapa de limpeza é especialmente importante neste trabalho, visto que a maior parte dos dados que originalmente compõem as tabelas disponibilizadas pelo IFMA possuíam valores nulos ou em branco. A limpeza foi iniciada com a remoção dos atributos duplicados nas tabelas. Somadas as três tabelas, existiam 111 atributos, dos quais 10 estavam duplicados e foram removidos, restando 101 atributos. Foram também removidos registros de alunos para os quais não haviam matriculas e os registros correspondentes aos cursos de educação a distância de Licenciatura em Química e Licenciatura em Informática, considerando que a EaD possui especificidades que não estão englobadas no escopo deste trabalho.

Originalmente existiam cinco valores de status para os alunos: *matriculado*, *evasão*, *formado*, *concluído* e *concludente* [1]. Considerando o objetivo deste trabalho, foram excluídos os registros cujo rótulo é igual a *matriculado*, mantendo-se os demais, que definem casos em que o aluno ou evadiu ou concluiu o curso. A Tabela 1 mostra as características das tabelas do banco de dados antes e depois da limpeza.

Utilizando as ferramentas de visualização do Weka para análise manual, constatou-se a necessidade de remover 61 atributos devido a: (i) conterem alto índice de valores nulos (acima de 60%); (ii) conterem informação que não contribuiriam com os algoritmos, como por exemplo *codigoTurma*; (iii) conterem informação sem nenhuma diversidade, como por exemplo, *nacionalidade*. Portanto, ao fim da limpeza a tabela que dará origem ao *dataset* principal conta com 40 atributos como apresentado na Tabela 2. Além da seleção manual de atributos, outras visões dos dados foram geradas através de seleção automatizada de atributos.

Integração

Foi criado um banco de dados através da importação dos arquivos CSV disponibilizados pelo IFMA, criando-se tabelas com seus respectivos relacionamentos. Embora existam outras formas de realizar a integração dos dados, neste trabalho optou-se pela criação de um banco de dados para gerar um repositório único, que serve de ponto de partida para as

[1]Concludente é aquele que esta em processo de conclusão, faltando por exemplo, apenas a defesa de monografia

Tabela 1: Descrição das tabelas

Tabela	Atributos antes da limpeza	Instâncias antes da limpeza	Atributos após a limpeza	Instâncias após a limpeza
Matrículas	61	8099	58	3454
Histórico	19	348961	15	97447
Pautas	31	14381	28	10039

Tabela 2: Atributos Utilizados

mediaFaltasSemestre
mediaNotas
percentualPresenca
disciplinasSemestre
tempoDeCurso
sexo
CR
turno
descricaoCurso
formaIngresso
codEscolaGraduacao
cor
anoConclusaoGraduacao
descCursoGraduacao
anoConclusao1grau
codEscola1grau
anoConclusao2grau
codEscola2grau
estadoCivilPais
grauInstrucaoPai
grauInstrucaoMae
paiFalecido
maeFalecida
rendaFamiliar
tipoEscolaOrigem
necessidadeFisica
necessidadeVisual
superdotado
condutasTipicas
rendaPerCapita
profissao
codGrauInstrucao
estadoCivil
codCidade
numeroFilhos
periodoLetivoIni
periodoAtual
siglaCurso
periodoLetivoAtual
situacaoMatricula

demais etapas do pré-processamento, devido às possibilidade de usar as consultas SQL na realização das etapas posteriores.

Após esta etapa inicial de limpeza, foi gerada a partir de uma consulta SQL uma tabela única que representa a integração das três tabelas iniciais, cujo resultado foi exportado para um arquivo CSV. Algumas limitações foram encontradas com o uso do formato CSV no Weka, como por exemplo a presença de caracteres que pudessem ser confundidos com separadores. Para solucionar esse problema, foi necessário previamente editar e excluir os caracteres vírgula(,), ponto-e-vírgula(;) e aspas(“) dos arquivos CSV. Após serem carregados pela primeira vez com o Weka, os arquivos foram exportados para o formato nativo do Weka, o ARFF.

Transformação

Inicialmente, o conjunto total de dados continha informações sobre cada disciplina cursada por um aluno ao longo de sua vida acadêmica. Como o objetivo da tarefa de classificação é criar um modelo capaz de classificar novos estudantes como evasores ou não, foi necessário transformar as diversas tuplas existentes para cada aluno em uma única tupla para cada aluno. Para tanto, alguns atributos foram manipulados matematicamente, de forma a refletirem a variação ao longo dos semestres. Esses atributos e as manipulações realizadas são descritos a seguir.

O atributo *mediaFaltasSemestre* possui a média de faltas do aluno, obtida através da soma de todas as faltas do aluno ao longo de toda a sua vida acadêmica dividida pelo número de semestres em que o aluno se matriculou. Da mesma forma, o atributo *disciplinasSemestre* é obtido a partir da soma de todas as disciplinas nas quais o aluno se matriculou dividido pelo número de semestres cursados. Os atributos *mediaNotas* e *percentualPresenca* são a média simples dos respectivos valores ao longo da vida acadêmica dos estudantes. O atributo *tempoDeCurso* foi criado subtraindo-se o ano letivo atual do aluno pelo ano letivo inicial.

Para finalizar a etapa de transformação, foi utilizado o filtro *Spread Subsample* do Weka para alterar a proporção dos registros com base no atributo classe, ou seja, define o balanceamento de classes do *dataset*. Neste trabalho foi usado o valor 1 como parâmetro, de forma que a proporção entre registros de alunos evasores e alunos que concluíram os cursos fosse de 1 para 1. Antes da aplicação deste filtro o *dataset* contava com 993 instâncias, sendo 287 alunos que concluíram os cursos e 706 que evadiram. Após a aplicação, o *dataset* contava com 287 instâncias de cada tipo.

Ao final da etapa de transformação, o *dataset* principal deste trabalho foi definido, com 40 atributos e 574 instâncias. É importante ressaltar que o número reduzido de instâncias, se comparado ao volume inicial de dados disponíveis, se justifica pelo fato que as informações foram condensadas em uma instância por aluno, enquanto que inicialmente haviam várias por aluno. Ainda, considera-se aqui apenas os alunos que concluíram o curso ou evadiram. Finalmente, na etapa em que os rótulos da classe foram balanceados, houve redução no número de instâncias.

Seleção de atributos

Devido à importância da seleção apropriada de atributos foram testadas duas abordagens de seleção automatizada de atributos, com o objetivo de identificar dentre elas qual a mais apropriada para a base de dados e o problema em questão. Assim, lidam-se neste trabalho com três conjuntos de dados selecionados de forma distinta: o primeiro é obtido através da seleção manual, conforme apresentado no início desta seção; os demais são resultado da aplicação dos

algoritmos de seleção descritos a seguir no *dataset* obtido manualmente (40 atributos).

- *Correlation-based Feature Selection*

Nesta técnica considera-se que um bom conjunto de atributos contém os que são altamente correlacionados à classe, mesmo que não sejam correlacionados uns com os outros (Hall; 1999). CFS é um algoritmo de filtro simples, que cria um ranking de subconjuntos de atributos, de acordo com uma função heurística de avaliação. A função prioriza subconjuntos que contém atributos altamente correlacionados ao atributos classe, e não-correlacionados entre si. Dessa forma atributos irrelevantes são eliminados pois possuem baixa correlação com a classe, e atributos redundantes são eliminados pois são altamente correlacionados com um ou mais dos demais atributos. Os seguintes atributos foram selecionados pelo algoritmo de CFS, implementado no Weka: *tempoDeCurso*, *CR*, *turno*, *necessidadeVisual*, *condutasTipicas*.

- *Information gain*

O valor de um atributo é medido a partir do ganho de informação que ele proporciona em relação ao atributo classe, de modo semelhante ao explicado na Seção 3.2.3. É um método de seleção de atributos amplamente utilizado, porém possui a desvantagem de não levar em consideração a relação entre os atributos, apenas a relação entre cada atributo e a classe. O cálculo leva em consideração a razão de ganho de informação, que vai de 0 a 1, entre o ganho de informação e o valor intrínseco do atributo (entropia). Em geral, atributos com um maior número de valores distintos são selecionados por terem uma alta razão de ganho de informação. Esta propriedade explica a seleção do atributo *codEscola2grau*, que possui um grande número de valores diferentes. Os seguintes atributos foram selecionados pelo algoritmo de *information gain*, implementado no Weka: *tempoDeCurso*, *codEscola2grau*, *percentPresenca*, *CR*, *mediaFaltasSemestre*, *periodoAtual*, *mediaNotas*, *FormaIngresso*, *disciplinasSemestre*, *codEscola1grau*. O atributo a ser predito é o status do aluno: matriculado ou evadido, pois o interesse é saber se o aluno continua no curso ou se já se evadiu.

5 Resultados e Discussão

Todos os experimentos relatados neste estudo foram realizados com validação cruzada por 10 vezes (*10 fold cross validation*) (Hastie et al.; 2001). Os valores da Tabela 3 mostram a porcentagem de instâncias classificadas corretamente (acurácia) para cada um dos algoritmos, considerando diferentes métodos de seleção de atributos.

De forma geral, o algoritmo J48 obteve os melhor resultados, tendo a maior taxa de acerto de 98.08% sendo obtida pela combinação com os atributos selecionados via *CFS*. Entretanto, pode-se observar que para todas as configurações testadas a acurácia foi acima de 93%, e similar entre os algoritmos.

Os valores da Tabela 4 mostram a sensibilidade (*recall*), ou taxa de verdadeiros positivos, para cada um dos algoritmos. Para todas as estratégias de seleção de atributos a sensibilidade (*recall*) dos algoritmos SVM e J48 se mostrou superior em relação ao algoritmo Naive Bayes; a diferença entre os dois primeiros, entretanto, é de apenas 0,01. O melhor resultado (0.96) foi encontrado com a combinação do algoritmo J48 e a estratégia de seleção de atributos CFS.

Tabela 3: Acurácia

Datasets	Algoritmos		
	NB	SVM	J48
Seleção Manual	0,94	0,96	**0,97**
CSF	0,94	0,97	**0,98**
InfoGain	0,93	0,97	**0,97**

Tabela 4: Sensibilidade (Recall)

Datasets	Algoritmos		
	NB	SVM	J48
Seleção Manual	0,89	0,94	**0,95**
CSF	0,89	0,95	**0,96**
InfoGain	0,88	**0,95**	**0,95**

Os valores da Tabela 5 mostram a taxa de verdadeiros negativos, ou especificidade, para cada um dos algoritmos. Observa-se que todos os algoritmos obtiveram um desempenho maior ou igual a 99% em relação à especificidade, ou seja, foram capazes de classificar alunos **não evasores** de forma correta. Isso se deve ao fato de que os dados relativos a esses alunos são mais completos e variam pouco, ou seja, a variabilidade dos atributos dentro do grupo de não evasores é pequena, permitindo que o modelo aprenda melhor como classificá-los.

Tabela 5: Especificidade

Datasets	Algoritmos		
	NB	SVM	J48
Seleção Manual	**0,99**	0,98	**0,99**
CSF	**1,00**	**1,00**	**1,00**
InfoGain	**0,99**	**0,99**	**0,99**

A Tabela 6 mostra a eficiência para cada um dos algoritmos. A eficiência de um modelo é calculada como a média entre sensibilidade e especificidade. Modelos com eficiência próxima de 1 são mais assertivos em prever a condição nos casos em que ela realmente existe. Neste trabalho essa métrica é relevante pois significa que a maioria dos alunos classificados como evasores realmente o são, de forma que as ações a serem tomadas no combate à evasão podem ser direcionadas ao público correto. O algoritmo com melhor desempenho com relação a essa métrica foi o J48.

Tabela 6: Eficiência

Datasets	Algoritmos		
	NB	SVM	J48
Seleção Manual	0,94	0,96	**0,97**
CSF	0,94	0,97	**0,98**
InfoGain	0,93	**0,97**	**0,97**

Em termos de descoberta de conhecimento, ou seja, daquilo que esta além dos resultados numéricos, cabe ressaltar que os atributos mais relevantes para a classificação nos algoritmos utilizados foram *tempoDeCurso* e também o *coeficiente de rendimento*(CR) do aluno. No melhor modelo gerado pelo J48, por exemplo, a maior probabilidade de evasão está associado a tempo de curso inferior a 3 semestres. O segundo atributo mais relevante para determinar a evasão é o coeficiente de rendimento (CR). Neste caso, estudantes com coeficiente de rendimento menor ou igual a 5.0 evadem seus cursos com alta probabilidade, a menos que permaneçam matriculados por mais do que 9 semestres.

6 Conclusões e Trabalhos Futuros

Este trabalho expôs a aplicação do processo de KDD em dados acadêmicos do IFMA, de forma a mostrar que é possível conduzir um estudo destes dados com o objetivo de fornecer à administração da instituição conhecimento acerca da evasão; mais especificamente, o processo visou induzir uma classificador capaz de classificar novos alunos como potenciais evasores ou não.

Os modelos de classificação criados neste trabalho possuem bom desempenho em todas as métricas analisadas, sendo capazes de classificar alunos **evasores** e **não evasores** com alta acurácia. Entretanto, espera-se estender este trabalho de forma a confirmar a qualidade desses classificadores realizando testes com dados novos, de alunos ingressantes, e acompanhá-los ao longo dos semestres.

Este trabalho pode ser estendido em trabalhos futuros de diversas formas. Dentre elas, é possível extrair um grande número de diferentes visões dos dados a partir do banco de dados criado neste trabalho. Estas visões constituem novos *datasets* para investigação. Nestes *datasets* as etapas documentadas neste trabalho podem ser replicadas. Por exemplo, pode-se explorar o caráter temporal dos dados, extraindo *datasets* contendo o número de reprovações de cada aluno por semestre, entre outros.

Uma outra extensão possível é criar um módulo que automatize ao máximo as etapas realizadas manualmente neste trabalho, para que os gestores do IFMA possam, periodicamente, conduzir a análise acerca da evasão na instituição. Os desafios envolvem a criação de uma interface simples e intuitiva para os gestores, bem como a investigação acerca da possibilidade da etapa de pré-processamento ser completamente automatizada, e de que forma isto pode ser feito.

Agradecimentos

Os autores gostariam de agradecer ao Instituto Federal do Maranhão por fornecer o banco de dados para este estudo.

Referências

Bhargava, N., Sharma, G., Bhargava, R. and Mathuria, M. (2013). Decision tree analysis on j48 algorithm for data mining, *International Journal of Advanced Research inComputer Science and Software Engineering* **3**(6): 1114–1119.

Braga, A., de Carvalho, A. C. P. L. F. and Ludermir, T. B. (2000). *Redes Neurais Artificiais: Teoria e Aplicações*, LTC, Rio de Janeiro.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition, *Knowledge Discovery and Data Mining* **2**(2): 1–43.

Campbell, C. (2000). An introduction to kernel methods, *in* R. J. Howlett and L. C. Jain (eds), *Radial Basis Function Networks: Design and Applications*, Springer Verlag, Berlin.

Chuchra, R. (2012). Use of data mining techniques for the evaluation of performance: A case study, *International Journal of Computer Student Science and Management Research* .

Dangeti, P. (2017). *Statistics for Machine Learning*, Packt Publishing.

Faceli, K., Lorena, A. C., Gama, J. and de Carvalho, A. C. P. L. F. (2011). *Inteligência Artificial: Uma Abordagem de Aprendizagem de Máquina*, LTC/Grupo Gen.

Fayyad, U. (1996). From data mining to knowledge discovery in databases, *AI magazine* **17**(3): 37–54.

Goldschmidt, R., Passos, E. and Bezerra, E. (2015). *Data Mining: Conceitos, técnicas, algoritmos, orientações e aplicações*, 2 edn, Elsevier.

Hall, M. A. (1999). *Correlation-based Feature Selection for Machine Learning*, PhD thesis, University of Waikato.

Hand, D., Mannila, H. and Smyth, P. (2001). *Principles of Data Mining*, The MIT Press.

Hastie, T., Tibshirani, R. and Friedman, J. (2001). *The Elements of Statistical Learning*, Springer Series in Statistics, Springer.

INEP (2018). Sinopses estatísticas da educação superior - graduação, visitado em julho/2018, http://portal.inep.gov.br/superior-censosuperior-sinopse.

James, G., Witlen, D., Hastie, T. and Tidshirani, R. (2013). *An Introduction to Statistical Learning: with Applications in R*, Springer.

Li, L. and Li, C. (2015). Research and improvement of a spam filter based on naive bayes, *2015 7th International Conference on Intelligent Human-Machine Systems and Cybernetics*, Vol. 2, pp. 361–364.

Lobo e Silva, R., Montejunas, P., Hipólito, O. and Lobo, M. (2007). Evasão no ensino superior brasileiro, *Cadernos de Pesquisa* **37**(132).

Lorena, A. C. and de Carvalho, A. C. P. L. F. (2007). Uma introdução às support vector machines, *RITA* **XIV**(2).

Makhabel, B. (2015). *Learning Data Mining with R*, Packt Publishing.

Marques, J. L. Q. (2014). Mineração de dados educacionais: um estudo de caso utilizando o ambiente virtual do senai.

Márquez-Vera, C., Morales, C. and Soto, S. (2013). Predicting school failure and dropout by using data mining techniques, *IEEE Journal of latin-american learning technologies* **8**(1).

Obsivac, T., Popelinsky, L., Bayer, J., Geryk, J. and Bydzovska, H. (2012). Predicting drop-out from social behaviour of students, *Proceedings of the 5th International Conference on Educational Data Mining*.

of Waikato, T. U. (2015). Weka: Data mining software in Java, https://www.cs.waikato.ac.nz/ml/weka/. Accessed: 2018-09-05.

Quinlan, J. R. (1993). *C4.5: programs for machine learning*, Morgan Kaufmann Publishers.

Romero, C. and Ventura, S. (2010). Educational data mining: A review of the state of the art, *IEEE Transactions on systems, man, and cybernetics – part c: applications and reviews* **40**(6).

Tair, M. M. A. and El-Halees, A. M. (2012). Mining educational data to improve students performance, *International Journal of Information e Communication Technology Research* **2**(2).

Witten, I. H., Frank, E. and Hall, M. A. (2011). *Data Mining: Practical Machine Learning Tools and Techniques*, 3 edn, Morgan Kaufmann Publishers.