Uma abordagem para predição de estudantes em risco utilizando algoritmos genéticos e mineração de dados: um estudo de caso com dados de um curso técnico a distância

Emanuel Marques Queiroga[1,2], **Cristian Cechinel**[1,3], **Marilton Sanchonete de Aguiar**[1]

[1]Programa de Pós-Graduação em Computação
Universidade Federal de Pelotas (UFPel) - Pelotas, RS, Brasil

[2]Campus Visconde da Graça (CaVG)
Instituto Federal Sul-rio-grandense (IFSul) - Pelotas, RS, Brasil

[3]Centro de Ciências, Tecnologias e Saúde (CTS)
Universidade Federal de Santa Catarina (UFSC) - Araranguá, SC, Brasil

{emanuel.queiroga,marilton}@inf.ufpel.edu.br

contato@cristiancechinel.pro.br

Abstract. *This paper demonstrates a non-traditional approach attempting to predict student dropout. For this, evolutionary system technics are used, which in this implementation seeks the optimization through the competition of six classifiers with initially random generated parameters among the possible ones for each classifier, the fitness function ranks the population at the end of each epoch. At last, the fittest individuals of each classifier are selected, and they compete with each other. The six classifiers, using the standard configuration, are compared and then contrasted with those obtained by the proposal. In this way, it was possible to obtain an improvement in the performance, with average gains ranging from 4% to 6%, in some cases up to 10%, depending on the metric.*

Resumo. *Este artigo apresenta uma abordagem não tradicional na tentativa de prever a evasão de estudantes. Para isso, são utilizadas técnicas de sistemas evolutivos buscando a otimização através da competição de seis classificadores com parâmetros inicialmente gerados aleatoriamente entre os possíveis para cada um. Por fim, os indivíduos mais aptos de cada classificador são selecionados e competem. Os mesmos seis classificadores, usando a configuração padrão, são utilizados para comparação com os resultados obtidos pela proposta. Essa abordagem resulta em uma melhora significativa no desempenho, com ganhos médios variando de 4% a 6%, em alguns casos até 10%, dependendo da métrica.*

1. Introdução

Entre as diferentes formações a distância oferecidas no Brasil destaca-se a Rede ETEC, ela é responsável pela oferta de cursos técnicos disponibilizados na modalidade à distância pelo governo federal em cidades do interior do país. Para acesso a esses cursos são utilizados os Ambientes Virtuais de Aprendizagem (AVA), sendo o Moodle uma das principais plataformas, pois apresenta uma série de recursos que visam auxiliar o processo de aprendizagem.

DOI: 10.5753/cbie.wcbie.2019.119

Periodicamente são realizados censos visando recolher informações sobre os cursos da Educação a Distância (EAD) ofertados no Brasil. No [Censo 2018], foram encontrados indicativos de até 50% de evasão em cursos totalmente a distância. Isso demonstra que essa modalidade de ensino sofre com altos índices de evasão, o que torna a pesquisa por métodos que auxiliem a diminuição desses um dos seus principais desafios.

Nesse cenário, a mineração de dados apresenta-se como uma alternativa, tanto para o tratamento quanto para a descoberta de conhecimento, nas bases geradas pelas informações dos estudantes nos AVA's. Atualmente a Mineração de Dados Educacionais (EDM) vem se estabelecendo como uma linha de pesquisa forte e consolidada, que possui grande potencial para melhora da qualidade do ensino a distância [Baker et al. 2011].

Contudo, maximizar os resultados obtidos pela EDM ainda é um desafio considerável, tendo em vista que os diferentes algoritmos que geralmente são utilizados apresentam uma grande variação de nas taxas de acerto a depender do conjunto de dados de entrada, a quantidade e qualidade dos mesmos. Devido a essa busca pela maximização de resultados, pode se tornar atraente a aplicação de métodos concorrentes, como a evolução populacional dos algoritmos genéticos (AG), se tornando uma alternativa as técnicas de predição usuais.

Desta forma, o presente trabalho busca apresentar uma abordagem para a detecção de alunos em risco de evasão em cursos técnicos a distância, aliando técnicas tradicionais de KDD com algoritmos genéticos na etapa de geração dos modelos de predição. Para isso, a metodologia utilizada considera apenas a contagem de interações dos estudantes dentro do AVA e atributos derivados dessas contagens. A premissa inicial é de que essa estratégia permita uma maior generalização em diferentes cursos, uma vez que não utiliza diferenciações entre os diferentes tipos de interações, nem informações de outra ordem encontradas fora do AVA (dados demográficos, etc).

O artigo está estruturado da seguinte maneira. A Seção 2 apresenta alguns trabalhos relacionados com o problema de predição de estudantes em risco e implementações de algoritmos genéticos voltados para EDM. Na Seção 3 é apresentada a hipótese de pesquisa deste trabalho. Na Seção 4 são descritos os dados e o método utilizado nos experimentos realizados. A Seção 5 discute os resultados alcançados, e a Seção 6 apresenta as conclusões do trabalho.

2. Trabalhos Relacionados

Nesta seção apresentaremos os trabalhos relacionados aos temas abordados nesse artigo, dando enfase nos utilizados para a mineração de dados educacionais e na utilização de algoritmos genéticos na otimização de classificadores.

2.1. Mineração de dados Educacionais

A mineração de dados educacionais é uma crescente área de pesquisa cientifica e que está intimamente ligada a Análise de Aprendizagem (Learning Analytics) [Siemens and d Baker 2012]. A EDM tem como uma de suas premissas à busca pela descoberta de conhecimento sobre as formas de aprendizagem, desempenho e evasão [Baker and Inventado 2014].

Na análise de métodos que auxiliem na predição de acadêmicos que apresentam um risco de evasão, podemos destacar a pesquisa de [Lykourentzou et al. 2009]. Nela são

utilizados dados demográficos como renda familiar, sexo, residência e atributos derivados, entre outros, combinados com informações extraídas dos cursos, como interações, notas e data de entrega dos trabalhos. Como método de predição é utilizada uma combinação de algoritmos, como Redes Neurais e maquina de vetor de suporte e lógica Fuzzy. Assim os acadêmicos são separados em conjuntos de acordo com quantos classificadores o apontaram como em risco e aplicada uma técnica de que utiliza Fuzzy para a classificação final do mesmo. Em seus experimentos o autor apresenta resultados que podem alcançar até 94% de acerto na situação do aluno.

A busca por métodos que possam ser generalizáveis, portanto, replicáveis a outros cursos, apresenta uma significativa parcela da área de pesquisa. Assim, [Whitehill et al. 2017] propõe uma arquitetura não dependente de dados únicos, trabalhando com o fluxo de cliques que os acadêmicos efetuam em um MOOC. Para isso ele captura dados de um curso e busca treinar modelos diferentes de predição e testar em outros cursos e ambientes. Nos experimentos são demonstradas taxas que entre 87% testando em cursos diferentes e 90% se testados no mesmo curso, assim não variando significativamente de acordo com o ambiente.

2.2. Algoritmos Genéticos aplicados a mineração de dados

Os algoritmos genéticos são amplamente usados na mineração de dados, podendo ser implementados como o próprio classificador ou como otimizador dos resultados [Minaei and Punch 2003] como proposto nessa abordagem.

Uma proposta de AG para otimização é apresentada por [Márquez-Vera et al. 2016], onde é utilizada uma variante do grammar-based genetic programming para melhorar a classificação de acadêmicos em risco de evasão. Essa técnica é aplicada sobre o algoritmo ICRM, proposto em [Cano et al. 2013], assim são ajustados os parâmetros do classificador até que se segue a um método que apresente maior aptidão. Os experimentos utilizaram dados de cursos de rápida duração (4 – 6 semanas). Em comparação com os classificadores usuais, o algoritmo proposto apresenta maior taxa de predição, se estabelecendo como uma alternativa a cursos que disponham das características utilizadas na proposta.

Na proposta de [Xing et al. 2015] é apresentada a utilização de AG para as etapas de seleção de variáveis e classificação de alunos quanto ao desempenho em uma disciplina de um curso. Para isso ele sugere uma abordagem quantifica as atividades dos alunos no MOOC em 6 variáveis pré-definidas, com o objetivo de diminuir a dimensionalidade dos dados. Como classificador é implementado o algoritmo ICRM proposto por [Cano et al. 2013]. Assim em seu estudo foi possível obter resultados superiores em até 6% se comparados as técnicas tradicionais, como Nave Bayes, Random Forest, MLP, entre outros, tanto na etapa de predição da situação final do aluno, quanto na interpretação dos modelos gerados.

3. Proposta

Este artigo traz como proposta avaliar a utilização de um algoritmo genético criado para otimização de classificadores, visando auxiliar na tarefa de predição da evasão em cursos EAD.

Para os testes e comparação foram selecionados os algoritmos J48 e Random Forest (RF), Naïve Bayes (NB), o Multilayer Perceptron (MLP), Regressão Logistica (RLL) e o meta-algoritmo AdaBoost. Os algoritmos selecionados são executados em suas configurações padrão e comparados com a implementação do algoritmo genético proposto. O conjunto de dados utilizado para os experimentos são o mesmo que anteriormente serviram como base para alguns artigos publicados e que demonstram a eficácia dos algoritmos selecionados para os testes [Queiroga et al. 2017].

Nos AG's o conjunto de soluções é definido por um espaço onde será efetuada a busca pela solução ótima, mas nem sempre esta será a global [Fonseca et al. 1993]. Esse é um fator que é diretamente dependente do problema, do tempo que pode ser gasto na busca, do resultado esperado e do conjunto de dados de entrada, entre outros, e que deverá ser considerado no momento que o algoritmo é projetado[Hartmann 1998].

Para este trabalho é proposta uma abordagem de busca limitada por épocas, assim o algoritmo criará um número N de gerações, onde N é pré-definido no momento de configuração do mesmo, e ao final trará como resultado um valor que pode ser a solução global ou local. Esse fluxo de informações é apresentando na figura 1, sendo no algoritmo proposto, utilizadas 50 épocas.

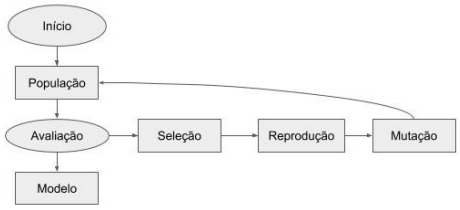

Figura 1. Diagrama de fluxo

Tendo como base a teoria de Koza [Koza 1990], o algoritmo proposto faz uso de um método de classificação onde diversos classificadores, com diferentes configurações, concorrem uns contra os outros. Ao final das épocas estipuladas, a solução é obtida através do classificador com a configuração que alcançou o maior valor na métrica estipulada, neste artigo, predição da situação acadêmica ao termino do curso, podendo esta ser ou não a global [Sebastiani 2002].

Em cada um dos algoritmos de classificação implementados no AG (J48, RF, NB, MLP, RLL e AdaBoost) foram selecionados parâmetros específicos de configuração, como, por exemplo o número de camadas ocultas do MLP e o tamanho da árvore nas árvores de decisão. Os parâmetros utilizados no MLP são apresentados na figura 2

Para inicialização do algoritmo proposto, é feita de forma randômica a geração de 100 indivíduos para cada um dos algoritmos selecionados. Sendo que cada um desses indivíduos é diferente entre si, o classificador é inicializado com a configuração gerada aleatoriamente e os resultados obtidos são salvos para comparação ao final da rodada. No final da rodada são comparados os resultados obtidos por cada um dos 100 indivíduos de cada um dos 6 algoritmos.

Nesse momento a concorrência dá-se somente entre esses membros de um mesmo classificador, ou seja, um indivíduo que seja da classe Adaboost concorre somente con-

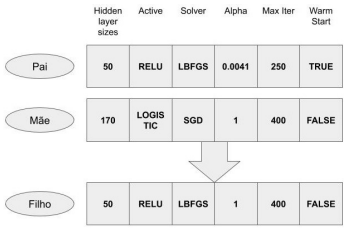

Figura 2. Reprodução

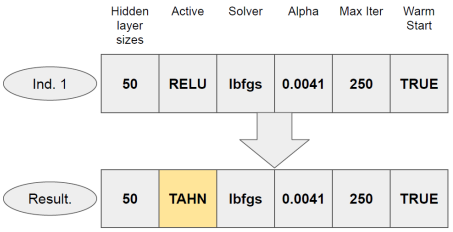

Figura 3. Mutação Genética

tra outros AdaBoost. Ao fim da época, é ativada a função de avaliação (Fitness) que é primordial nos algoritmos genéticos, pois, é a partir dos resultados obtidos nela que é dada a oportunidade do código genético (cromossomos) daquele membro do teste evoluir. Ela tem como objetivo avaliar a aptidão de um determinado indivíduo em resolver o problema proposto, é atribuído um conceito numérico para cada membro do teste a partir da sua capacidade em predizer a situação de um acadêmico ao final do curso. Assim, quanto maiores as taxas de acerto obtidas, maior será o valor atribuído aquele indivíduo.

Para geração das próximas épocas, são levadas 25 melhores configurações, conforme a função de avaliação, de cada um dos classificadores sem nenhuma alteração. Isso se faz necessário para que sempre se mantenha os indivíduos mais aptos e que obtiveram os melhores resultados anteriormente. Para a complementação dos indivíduos são aplicadas uma função de mutação (figura 3), um fator de cruzamento (figura 2) e a geração de novos integrantes aleatórios no grupo.

O cruzamento (crossover), é feito utilizando o conceito baseado na herança genética das reproduções sexuadas, onde cada filho recebe uma parte do código genético do pai e parte da mãe, como exemplificado na figura 2. Assim são combinadas as configurações dos indivíduos mais aptos da última geração, sendo um o pai e o outro a mãe. No algoritmo implementado, a escolha dos indivíduos que irão ceder parte de seu código genético para formar um novo membro é feita de forma randômica entre os 25 mais bem colocados daquele classificador na última geração.

Na mutação, os melhores indivíduos têm sua configuração, nessa abordagem o código genético, alterada de forma aleatória. Ou seja, uma determinada característica de um indivíduo selecionado na etapa anterior recebe uma configuração gerada aleatoriamente, conforme figura 3. Ainda na figura 3, é possível notar que um indivíduo da classe MLP selecionado tem o valor do parâmetro ”active” alterado. Essa taxa de mutação é estabelecida como alteração de apenas uma configuração dentre as possíveis do classificador.

O quarto e último fator de geração de novas populações é a aleatoriedade, sendo que para cada uma das gerações um quantitativo de indivíduos é gerado novamente de forma randômica mesmo que estes já podem ter sido gerados em épocas anteriores. Isso busca garantir a diversidade da população, tentando diminuir a hipótese de que a solução chegue em um máximo local e não tenha oportunidade de evoluir ao máximo global.

Os quantitativos de formação das populações da segunda rodada em diante são os seguintes, 25% são indivíduos novos gerados de forma aleatória, 25% são os selecionados da geração anterior a partir da função de aptidão, 25% são fruto de cruzamento e os

últimos 25% são de mutações dos mais bem classificados da etapa anterior.

Desta forma, o fluxo de geração final das populações para cada uma das épocas é apresentado na figura 1. Onde temos o início, que é a geração aleatória, a população que são os indivíduos daquela época, a função de avaliação que é quem atribui uma pontuação para cada membro, a seleção busca separa os 25 melhores membros, a reprodução que faz a combinação do código genético e a mutação que alterada um parâmetro aleatório.

O processo apresentado na figura 1 é repetido por 50 épocas, ao final para cada um dos seis classificadores é selecionado o indivíduo que apresentou maior aptidão. Com a seleção dos 6 mais aptos feita, eles concorrem entre si uma última vez para que ai seja efetuada a seleção final do classificador e configuração que se mostrou mais apto.

4. Metodologia Utilizada

Nessa seção é apresentada a implementação da abordagem proposta, sendo demonstrada a metodologia seguida no decorrer da experimentação, bem como as técnicas e ferramentas utilizadas no projeto. A metodologia seguida para o desenvolvimento desse trabalho utiliza a contagem de interações dos estudantes no AVA como a principal informação para a geração dos modelos de predição. As seções a seguir descrevem as características dos dados coletados, o pré-processamento realizado e a geração e avaliação dos modelos de predição.

4.1. Coleta

Foram coletados os logs de interações de cada uma das disciplinas do curso técnico em Administração ministrado no ocultado para revisão A Tabela 1 apresenta o volume de dados relativos a interações, a quantidade total de alunos desse curso, e as respectivas quantidades e percentuais de acadêmicos concluintes e evadidos.

Tabela 1. Dados utilizados

Quant. Logs	Nº de alunos	Evadidos (%)	Concluintes (%)
1.051.012	752	354 (47%)	398 (53%)

4.2. Pré-processamento dos dados

Na etapa de pré-processamento os dados passaram foram limpos, a anonimizados e agrupados quanto a aluno e semana. Também foram geradas variáveis derivadas da contagem das interações. Ao final, as interações foram contabilizadas ao longo de 103 semanas letivas que compunham os cursos. Além da contagem de interações semanais (75 semanas), foram contabilizadas também as contagens de interações diárias (525 dias), e a média, mediana e desvio padrão na semana. A Tabela 2 apresenta as variáveis utilizadas para a geração dos modelos de predição.

Tabela 2. Variáveis Utilizadas

Variável	Descrição
Interações diárias	Contagem de interações diárias (1 até 721 dias)
Interações Semanais	Contagem das interações na semana (1 até 103 semanas)
Média semanal	Média das contagens das interações na semana (1 até 103 semanas)
Mediana semanal	Mediana das contagens das interações na semana (1 até 103)
Desvio padrão semanal	Desvio padrão das contagens das interações na semana (1 até 103)
Situação final no curso	Situação final no curso (normal ou evadido)
Id	Id do estudante

Com os dados limpos e anonimizados foram separados em 3 bases diferentes, cada uma delas consistia nos dados referentes ao período de 25, 50 e 75 semanas de curso.

4.3. Geração dos modelos preditivos

A etapa de geração dos modelos preditivos consiste na aplicação de técnicas de mineração de dados, que buscam encontrar padrões que possam auxiliar na busca por um resultado aceitável na previsão de um determinado atributo, que no caso deste trabalho é a predição da variável situação.

A implementação das técnicas utilizadas nesse trabalho utilizou a linguagem de programação de alto nível Python[1], com a distribuição Anaconda[2]. Para desenvolvimento do algoritmo genético proposto nesse artigo, foi empregado a biblioteca de aprendizagem de máquina Scikit-learn (SKLearn)[3]. A escolha por essas tecnologias e ferramentas se deu pela ampla documentação disponível para ambas. Assim facilitando a aplicação das técnicas de mineração de dados e classificação, bem com a automatização dos processos necessários para a execução dos testes.

Para a etapa de geração dos modelos de predição e comparação com os resultados gerados pelo algoritmo genético, foram utilizados 6 diferentes algoritmos em sua configuração padrão do SKLearn: Nave Bayes (NB), Multilayer Perceptron (MLP), Random Forest (RF), J48, AdaBoost e Regressão Logica Linear (RLL). A escolha destes algoritmos se deve a estes serem alguns dos mais utilizados em pesquisas relacionadas ao tema e por já terem apresentado resultados satisfatórios em testes realizados anteriormente [Queiroga et al. 2017].

Já o algoritmo genético consistiu no estudo dos parâmetros que podem ser utilizados na implementação com SKLearn, assim foram definidos os que poderiam ser gerados de forma aleatória e executada a construção do mesmo em Python.

Os modelos de predição foram testados e avaliados em 3 momentos diferentes: 1) primeiras 25 semanas; 2) primeiras 50 semanas; e, 3) primeiras 75 semanas. Cada um dos cenários é referente aos dados do curso até o fim daquele semestre, sendo utilizada a técnica de validação cruzada (10-fold crossvalidation). Essa técnica se resume nos modelos serem gerados utilizando 9 subconjuntos diferentes e o teste é feito em 1 subconjunto, esse processo é repetido 10 vezes e a acurácia se dá pela média dos 10 testes.

A acurácia dos resultados é medida utilizando a taxa de Verdadeiros Positivos (TPR) (Acertos na predição de um estudante evadido sobre a quantidade de evadidos), na taxa de Verdadeiros Negativos (TNR) (Acertos na predição de um discente que irá finalizar o curso sobre a parcela de estudantes que finalizaram) e por último na Acurácia Geral que é a porcentagem de instâncias totais classificadas de forma correta.

5. Resultados

Esta Seção apresenta os resultados obtidos pelos modelos gerados por cada um dos algoritmos selecionados em comparação com a utilização do algoritmo genético. Nas figuras 4, 5, e 6 são apresentados os resultados obtidos nos experimentos realizados.

Para efeito de comparação, a melhor configuração para as primeiras 25 semanas de curso foi encontrada no individuo 37 da quarta época, gerado a partir da configuração

[1]https://www.python.org/
[2]https://www.anaconda.com/distribution/
[3]https://scikit-learn.org/

de um MLP com resultado 91,54%. Esse individuo foi fruto da reprodução de dois indivíduos da geração anterior, possuindo a configuração [hidden layer sizes=30, activation='logistic', solver='sgd', alpha='0.2855486101', max iter='353', warm start= "False"], enquanto a configuração default gerou um resultado de 84,9% tendo a seguinte configuração inicial, [hidden layer sizes=100, activation='relu', solver='adam', alpha='0.0001', max iter='200', warm start= "False"]. O ganho de 6,64% obtido com a otimização, demonstra que o AG proposto consegue customizar o classificador e resultar em uma predição mais exata, podendo aumentar o ganho em bases com maior quantitativo de dados.

Na figura 4, são demonstrados os resultados obtidos na métrica de Taxa de acertos Verdadeiros Positivos (TPR), que são os acadêmicos classificados como risco e que realmente se evadem do curso. Em uma comparação entre o classificador gerado pelo algoritmo genético e os classificadores com parametrização padrão é possível verificar que o AG tem uma taxa maior que os outros preditores no decorrer de todo o curso. Assim podemos entender que nesse experimento, o algoritmo proposto foi capaz de encontrar uma parametrização com taxas de acerto mais satisfatórias que as configurações usuais.

Na análise direta dos resultados, é possível notar que em todas as etapas do experimento o algoritmo genético encontrou uma parametrização que estabelece um resultado que percentualmente maior que os classificadores usuais. Assim, mesmo o percentual de 0.8% de melhora obtido na semana 50 em uma base de dados com 752 acadêmicos, sendo 226 utilizados somente no conjunto de teste, equivale a em torno de dois alunos de diferença. Entretanto, como as pesquisas identificam que a maior probabilidade de reversão da situação de evasão se da no inicio do curso, podemos analisar a semana 25, onde o algoritmo genético tem uma vantagem de 4% para o segundo colocado. Esse percentual representa uma melhora na predição de em torno de 8 alunos.

Na figura 5 ainda são apresentados os resultados obtidos na métrica de Taxa de acertos de Verdadeiros Negativos (TNR), que são os acadêmicos classificados como sem risco e que realmente concluem o curso. Nessa métrica é possível observar o crescimento linear da taxa de acertos do algoritmo genético. Em contraste com os classificadores usuais, podemos notar uma maior variação na classificação, assim ficando nítida a dificuldade de se encontrar um previsor que tenha uma taxa de acertos que seja uniforme. Assim, podendo ser eleito o classificador a ser aplicado durante o curso.

Desta forma, o algoritmo genético proposto demonstra que encontra uma configuração que pode trazer resultados satisfatórios na predição de acadêmicos tendem a concluir o curso, assim chegando próximo de encontrar a solução ótima para o caso.

O terceiro experimento, apresentado na figura 6 busca medir a taxa de acurácia total do algoritmo genético em comparação com os classificadores tradicionais. Assim nas primeiras 25 semanas de curso o AG apresenta resultados exatamente 6.6% superior aos obtidos pelos métodos tradicionais. Na predição de 50 semanas a diferença fica em 4.7%, com o algoritmo genético novamente com o melhor resultado. Já na predição de 75 semanas a diferença fica em 1.9% somente. No final o experimento 3 acaba por demonstrar que o AG obtém taxas de acertos maiores em praticamente todos os cenários do teste.

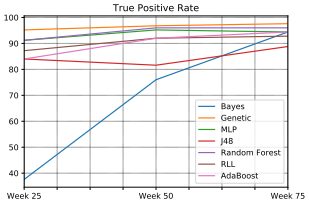

Figura 4. Verdadeiros Positivos

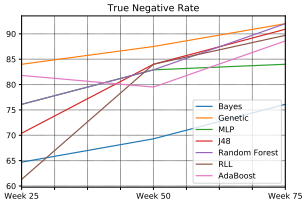

Figura 5. Verdadeiros Negativos

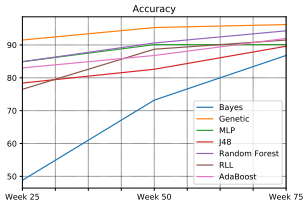

Figura 6. Acurácia Geral

6. Conclusões

O presente artigo apresenta uma abordagem diferente das usuais para a predição de estudantes em risco de evasão. Trazendo assim, como um de seus objetivos, testar a utilização de algoritmos genéticos na busca por modelos de predição customizados e otimizados para cada semana do curso.

Para a avaliação do método proposto, é necessário salientar que testar as quase infinitas combinações de configurações dentre um mesmo algoritmo, é uma tarefa complexa e exaustiva e que pode requerer uma infinidade de horas de trabalho. Sendo esse um problema recorrente da mineração de dados educacionais. Nos dados utilizados se formos testar apenas 6 algoritmos com parametrização pré-definidas teremos ao final 618 modelos para teste. Esse valor não consegue otimizar a busca por um modelo que seja customizado para cada semana, com uma configuração que seja a mais apta possível.

Ainda, a utilização de um determinado algoritmo e/ou configuração definida para que tenha um rendimento linear no decorrer do curso, pode ser evoluída para uma metodologia onde a cada semana tenhamos um modelo customizado. Assim, podendo ser possível a geração de modelos preditivos personalizados, principalmente em cursos com duração um pouco mais longa.

A abordagem proposta demonstrou resultados satisfatórios, principalmente no início dos cursos que é momento o mais propicio para a reversão da evasão. Na comparação geral, o AG manteve os melhores resultados em grande parte dos experimentos.Desta forma, conclui-se que a aplicação dos algoritmos genéticos pode ser uma alternativa viável para o andamento deste projeto, ficando como uma possibilidade a combinação com técnicas de teoria dos votos para uma otimização ainda maior.

7. Agradecimentos

Esse trabalho foi financiado pelo CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico) por meio do Edital Universal 01/2016, processo 404369/2016-2, e da Chamada Nº 17/2018, processo: 315445/2018-1.

Referências

Baker, R., Isotani, S., and Carvalho, A. (2011). Mineração de Dados Educacionais: Oportunidades para o Brasil. *Revista Brasileira de Informática na Educação*, 19(02):03.

Baker, R. S. and Inventado, P. S. (2014). Educational data mining and learning analytics. In *Learning analytics*, pages 61–75. Springer.

Cano, C. M.-v. A., Romero, C., and Ventura, S. (2013). Predicting School Failure and Dropout by Using Data Mining Techniques Predicting student failure at school using genetic programming and different data mining approaches with high dimensional and imbalanced data. (February).

Censo, E. (2018). Br 2016-relatório analítico da aprendizagem a distância no brasil. *Acesso em*, 16(08).

Fonseca, C. M., Fleming, P. J., et al. (1993). Genetic algorithms for multiobjective optimization: Formulationdiscussion and generalization. In *Icga*, volume 93, pages 416–423. Citeseer.

Hartmann, S. (1998). A competitive genetic algorithm for resource-constrained project scheduling. *Naval Research Logistics (NRL)*, 45(7):733–750.

Koza, J. R. (1990). *Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems*, volume 34. Stanford University, Department of Computer Science Stanford, CA.

Lykourentzou, I., Giannoukos, I., Nikolopoulos, V., Mpardis, G., and Loumos, V. (2009). Dropout prediction in e-learning courses through the combination of machine learning techniques. *Computers & Education*, 53(3):950–965.

Márquez-Vera, C., Cano, A., Romero, C., Noaman, A. Y. M., Mousa Fardoun, H., and Ventura, S. (2016). Early dropout prediction using data mining: a case study with high school students. *Expert Systems*, 33(1):107–124.

Minaei, B. and Punch, W. (2003). Using genetic algorithms for data mining optimization in an educational web-based system. volume 2724, pages 2252–2263.

Queiroga, E., Cechinel, C., and Araújo, R. (2017). Predição de estudantes com risco de evasão em cursos técnicos a distância. *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, 28(1):1547.

Sebastiani, F. (2002). Machine learning in automated text categorization. *ACM computing surveys (CSUR)*, 34(1):1–47.

Siemens, G. and d Baker, R. S. (2012). Learning analytics and educational data mining: towards communication and collaboration. In *Proceedings of the 2nd international conference on learning analytics and knowledge*, pages 252–254. ACM.

Whitehill, J., Mohan, K., Seaton, D., Rosen, Y., and Tingley, D. (2017). Delving deeper into mooc student dropout prediction. *arXiv preprint arXiv:1702.06404*.

Xing, W., Guo, R., Petakovic, E., and Goggins, S. (2015). Participation-based student final performance prediction model through interpretable genetic programming: Integrating learning analytics, educational data mining and theory. *Computers in Human Behavior*, 47:168–181.