UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

CURSO DE BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

RICARDO DANTAS DE OLIVEIRA

OTIMIZAÇÃO DE ALGORITMOS PARA PREVISÃO DE DESEMPENHO ACADÊMICO DE ESTUDANTES EM AMBIENTES EDUCACIONAIS

RECIFE

2017

UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO

RICARDO DANTAS DE OLIVEIRA

OTIMIZAÇÃO DE ALGORITMOS PARA PREDIÇÃO DE DESEMPENHO ACADÊMICO DE ESTUDANTES EM AMBIENTES EDUCACIONAIS

Monografia apresentada ao curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco como requisito parcial para obtenção do título de Bacharel em Ciência da Computação

Orientador: Prof. Dr. Rafael Ferreira Leite de Mello

Coorientador: Prof. Dr. Péricles B. C. de Miranda

RECIFE

2017

MINISTÉRIO DA EDUCAÇÃO E DO DESPORTO
UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO (UFRPE)
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO

http://www.bcc.ufrpe.br

FICHA DE APROVAÇÃO DO TRABALHO DE CONCLUSÃO DE CURSO

Trabalho defendido por Ricardo Dantas de Oliveira às 11 horas do dia 01 de setembro de 2017, no laboratório 39 do CEAGRI-02, como requisito para conclusão do curso de Bacharelado em Ciência da Computação da Universidade Federal Rural de Pernambuco, intitulado **Otimização de Algoritmos para Análise Automática de Desempenho de Estudantes em Ambientes Educacionais**, orientado por Rafael Ferreira Leite de Mello e aprovado pela seguinte banca examinadora:

Rafael Ferreira Leite de Mello
DEINFO/UFRPE

Pablo Azevedo Sampaio
DEINFO/UFRPE

Valmir Macário Filho
DEINFO/UFRPE

AGRADECIMENTOS

Agradeço primeiramente ao meu pai Wilton de Oliveira Dantas e minha mãe Maria de Lourdes Dantas da Silva, por terem me proporcionado uma boa educação e apoio incondicional durante toda minha vida.

Agradeço ao meu orientador Rafael Ferreira e ao meu Coorientador Péricles Miranda, pelos ensinamentos e orientações fornecidas durante todo o curso, além da motivação e paciência.

Agradeço a todo o corpo docente do Departamento de Estatística e Informática da Universidade Federal Rural de Pernambuco, pela imensa colaboração na minha formação acadêmica.

Agradeço também aos meus amigos da universidade, em especial a turma BCC2010.2, e a todos meus amigos do trabalho, que colaboraram direta ou indiretamente com a conclusão deste trabalho.

RESUMO

Devido ao aumento de cursos na modalidade de ensino à distância no Brasil, cresce a cada ano o uso de Ambientes Virtuais de Aprendizagem (AVA). Com isso é produzida uma grande quantidade de dados e a possibilidade de inferir diversas informações sobre os usuários. Uma informação comumente verificada, é a avaliação escolar, cujo objetivo principal é verificar o aproveitamento do curso pelos alunos. Com o crescimento da educação à distância, a aprendizagem de máquina, ramo da ciência da computação, passou a ser utilizada para auxiliar a área educacional. Uma dessas utilizações é antecipar a informação de alunos com baixo desempenho acadêmico, e consequentemente, possibilitar ao corpo docente atuar nesses alunos, evitando dois grandes problemas do ensino à distância: os altos índices de reprovação e evasão. Este trabalho tem como objetivo a otimização de algoritmos de aprendizado de máquina, utilizando técnicas de mineração de dados para otimizar parâmetros desses algoritmos. Com esses algoritmos otimizados, é possível a identificação antecipada de alunos com baixo desempenho acadêmico, com risco de reprovação, ou ainda a evasão desse aluno, que é um dos maiores problemas do ensino à distância. Para alcançar esse objetivo foram realizadas as seguintes etapas: identificação dos melhores algoritmos e otimização dos parâmetros desses algoritmos. Na etapa de identificação, os algoritmos que mostraram melhores média de resultados para todas as bases foram *Random Forest* e *Decision Stump*. Na etapa de otimização dos parâmetros, foi utilizado uma implementação do algoritmo PSO. Com esse algoritmo, foi possível aumentar a taxa de acerto dos algoritmos, otimizando seus parâmetros.

Palavras-chave: inteligência artificial, aprendizagem de máquina, ambientes virtuais de aprendizagem, otimização de parâmetros, otimização de enxame de partículas.

ABSTRACT

Due to increase of distance learning in Brazil, Virtual Learning Environments (VLE) grows every year. This produces a large data amount and possibility to inferring various informations about users. The school evaluation, commonly verified information, has the main objective to verify the achievements of the students in the class. Within the growth of the distance education, machine learning, which is a computer science branch, started to been used for educacional area. One of the uses is anticipates the information of students with low performance and, consequently, enables the faculty to act on these students avoiding two major problems of distance learning: High School Failure Rate and Dropout. This study aims to optimize machine learning algorithms, using data mining techniques to optimize algorithm parameters. With those algorithms optimized, it is possible to anticipate the low academic performance students and those who have high probability to school failure rate and dropout. To achieve this goal, the following steps were made: Identification of the best algorithms and optimization of the parameters found. In the identification step, the algorithms that showed the best average results for all bases were Random Forest and Decision Stump. In the step of optimizing algorithm parameters, the PSO algorithm was implemented. With this algorithm, it was possible to increase the rate of correctness of the algorithms, optimizing their parameters.

Keywords: artificial intelligence, machine learning, virtual learning environment, parameters optimization, particle swarm optimization.

LISTA DE FIGURAS

Figura 1 - Estrutura de uma árvore de decisão 15
Figura 2 - Eficácia na base de ensino Presencial 27
Figura 3 - Eficácia na base de ensino à Distância 28
Figura 4 - Eficácia na base de ensino Presencial sem Notas 28
Figura 5 - Eficácia na base de ensino à Distância sem Notas 29

LISTA DE TABELAS

Tabela 1 - Atributos selecionados na modalidade ensino Presencial 20
Tabela 2 - Atributos selecionados na modalidade ensino Presencial sem as Notas 21
Tabela 3 - Atributos Selecionados na modalidade de ensino à Distância 22
Tabela 4 - Atributos Selecionados na modalidade de ensino à Distância sem Notas..... 23
Tabela 5 - Resultados das bases completas, contendo os atributos de notas 25
Tabela 6 - Resultados das bases que não contém as notas**Error! Bookmark not defined.**
Tabela 7 - Descrição dos parâmetros do Random Forest 26
Tabela 8 - Descrição dos parâmetros do Decision Stump 26

LISTA DE ABREVIATURAS, SIGLAS E SÍMBOLOS

AVA	AMBIENTE VIRTUAL DE APRENDIZAGEM
PSO	*PARTICLE SWARM OPTIMIZATION*
UFAL	UNIVERSIDADE FEDERAL DE ALAGOAS
UFRPE	UNIVERSIDADE FEDERAL RURAL DE PERNAMBUCO
WEKA	*WAIKATO ENVIRONMENT FOR KNOWLEDGE ANALYSIS*

SUMÁRIO

1. INTRODUÇÃO 11
1.1. JUSTIFICATIVA 12
1.2. PROPOSTA 12
1.3. OBJETIVOS 13
1.3.1. OBJETIVO GERAL 13
1.3.2. OBJETIVOS ESPECÍFICOS 13
1.4. ESTRUTURA DO TRABALHO 13
2. FUNDAMENTAÇÃO TEÓRICA 14
2.1. APRENDIZAGEM DE MÁQUINA 14
2.2. ÁRVORE DE DECISÃO 14
2.3. PARTICLE SWARM OPTIMIZATION 16
3. TRABALHOS RELACIONADOS 18
4. METODOLOGIA DE DESENVOLVIMENTO 20
4.1. BASES DE DADOS 20
4.3. OTIMIZAÇÃO DOS PARÂMETROS 26
4.4. RESULTADOS ALCANÇADOS COM A OTIMIZAÇÃO 27
5. CONCLUSÃO 31
REFERÊNCIAS 32

1. INTRODUÇÃO

A crescente necessidade de capacitação e atualização profissional de qualidade, tornou o ensino à distância popular, usufruindo das facilidades e recursos da informática, além de flexibilidade no horário de estudo. A necessidade de um sistema de apoio para estes cursos à distância foi fundamental para o surgimento de Ambientes Virtuais de Aprendizagem (AVA) (POOJA, 2015) e outras tecnologias como chats e videoconferência, apoiando docentes e discentes em diversas modalidades de ensino.

AVA é um sistema *online* que disponibiliza uma grande quantidade de ferramentas, para melhorar a interação entre os participantes de um curso, e é muito utilizado, principalmente, nos cursos de Educação à Distância (EAD). Para que estes ambientes tenham sucesso, é necessário compromisso e envolvimento não somente do aluno, como dos professores e tutores.

Com o crescimento da utilização dos AVA promovido pelo aumento do ensino à distância, surgem alguns desafios para os profissionais de TI (Tecnologia da Informação) na área de educação. Um exemplo destes desafios é o tratamento e manipulação de grande volume de informações sobre discentes e docentes nesses sistemas (ROMERO, CRISTOBAL e VENTURA, SEBASTIAN, 2017). Na maioria das vezes, as bases de dados possuem muitas informações e é de difícil interpretação humana.

É importante destacar que a avaliação escolar tem como objetivo principal verificar se o aluno adquiriu o conteúdo passado. A avaliação do processo ensino-aprendizagem apresenta três tipos de funções: diagnóstica (analítica), formativa (controladora) e somativa (classificatória). Todas elas têm como objetivo verificar se o aluno adquiriu o conteúdo passado (BLOOM, B.S., HASTINGS, J. T. e MADAUS, G. F., 1993).

Na avaliação diagnóstica, leva-se em consideração o conhecimento prévio do aluno, visando constatar os pré-requisitos necessários de conhecimentos e habilidades. A avaliação somativa, é feita no final de um curso ou módulo de curso, e classifica o aluno de acordo com seu aproveitamento, enquanto a avaliação formativa, tem função controladora e é realizada no decorrer do período letivo, verificando se os alunos estão atingindo os objetivos previstos em cada etapa.

Com o auxílio da ciência da computação também na área educacional, algoritmos de aprendizado de máquina passaram a ser utilizados na avaliação formativa, com base nos dados gerados a partir das interações dos alunos com o AVA. O principal resultado de algoritmos para esse propósito, é predizer grupos de alunos que estão com dificuldade de atingir os objetivos da disciplina, detectar baixo desempenho e baixa interação. Com essa informação, o docente pode atuar para evitar dois grandes problemas da EAD, que são os grandes índices de evasão e reprovação. (MARI, M. M., OPRIME, P. C. *et all* 2011).

1.1. JUSTIFICATIVA

Devido à grande quantidade de dados gerados pelos AVAs, trabalhos já vêm utilizando algoritmos e técnicas de aprendizagem de máquina para inferir conhecimento e detectar alunos com baixo desempenho acadêmico, e consequentemente, possibilitar ao corpo docente atuar nesses alunos, evitando dois grandes problemas do ensino à distância: Alto índice de reprovação e evasão.

Por exemplo, Gottardo *et al.* (GOTTARDO, KAESTENER, NORONHA, 2012) atingiram 76% de índice de acerto na previsão de desempenho acadêmico dos estudantes, com o algoritmo *Random Forest* e MLP (*Multilayer Perceptron*: Perceptron MultiCamadas). Zaidah e Daliela (ZAIDAH IBRAHIM e DALIELA RUSLI, 2007) obtiveram 80% em todos os três modelos analisados: redes neurais artificiais, árvore de decisão, regressão linear, tendo como melhor resultado redes neurais artificiais. Por fim, Santana, Marcelo A. (2015) obteve os valores 79% e 82% de Medida-F, nos cursos de modalidade presencial e à distância, respectivamente, utilizando árvore de decisão.

A proposta deste trabalho é a avaliação de diferentes algoritmos de mineração de dados para auxiliar docentes na avaliação formativa dos alunos. Além disso, os algoritmos com maiores eficácias terão seus parâmetros otimizados, através de algoritmo de otimização.

1.2. PROPOSTA

Este trabalho propõe a otimizar parâmetros de diferentes algoritmos de aprendizagem de máquina para auxiliar professores no acompanhamento da avaliação formativa dos estudantes em ambientes educacionais.

A eficácia dos algoritmos é medida utilizando a Medida-F. Foram testados diversos algoritmos e os dois melhores resultados foram selecionados para terem seus parâmetros de entradas otimizados.

Uma das principais diferenças deste trabalho para os trabalhos relacionados é a utilização de algoritmos de otimização de parâmetros para melhorar os resultados dos classificadores utilizados.

Por fim, espera-se obter os algoritmos de predição otimizados para que possa facilitar a avaliação formativa dos alunos de ensino presencial e à distância, predizendo possíveis grupos de alunos com dificuldades em atingir os objetivos da disciplina ou com baixo desempenho acadêmico, para que os docentes possam atuar, diminuindo ou eliminando as reprovações e evasão escolar.

1.3. OBJETIVOS

1.3.1. OBJETIVO GERAL

Este trabalho tem como objetivo geral a avaliação de diferentes algoritmos de aprendizagem de máquina, fazendo o uso de algoritmos de otimização de parâmetros, para melhorar a eficácia de técnicas de predição de desempenho acadêmico de alunos, detectando possíveis candidatos ao insucesso.

1.3.2. OBJETIVOS ESPECÍFICOS

Dos objetivos específicos deste trabalho, destacam-se:

- Seleção de Conjunto de dados: foram utilizados um total de 4 conjunto de dados, sendo dois independentes e os outros dois com os atributos de notas removidos.
- Escolha dos melhores algoritmos de aprendizagem de máquina, para os conjuntos de dados.
- Executar o algoritmo de Otimização por Enxame de Partículas para otimização dos parâmetros de entrada dos classificadores.
- Escolha dos melhores métodos de atualização de partícula do algoritmo de otimização, para aumento da taxa de acerto.

1.4. ESTRUTURA DO TRABALHO

No Capítulo 1 é apresentado uma introdução sobre o trabalho, descrevendo o problema a ser tratado, a proposta, os objetivos gerais e específicos do documento. O Capítulo 2 apresenta os conceitos básicos relevantes ao tema como aprendizagem de máquina e algoritmos utilizados. O Capítulo 3 foi feita uma revisão da literatura sobre o tema. No Capítulo 4 foi exposto o experimento, a otimização e os resultados obtidos. Por fim, o Capítulo 5 destina-se às considerações finais.

2. FUNDAMENTAÇÃO TEÓRICA

Este capítulo apresenta conceitos para melhor compreensão dos tópicos abordados neste trabalho.

2.1. APRENDIZAGEM DE MÁQUINA

Aprendizagem de máquina é uma subárea da inteligência artificial que tenta obter conhecimento a partir de um conjunto particular de entrada de informações. Um algoritmo de aprendizado de máquina contém instruções de tomadas de decisões baseadas no conhecimento adquirido em exemplos resolvidos com sucesso (WEISS, S. M. e KULIKOWSKI, C. A. 1991).

Dentro das categorias do aprendizado de máquina, tem-se o aprendizado supervisionado e não-supervisionado. O aprendizado supervisionado tem como objetivo classificar um determinado dado, podendo ser uma imagem, um documento, um texto, através de parâmetros fornecidos como entrada (ROLIM, V.B. 2016).

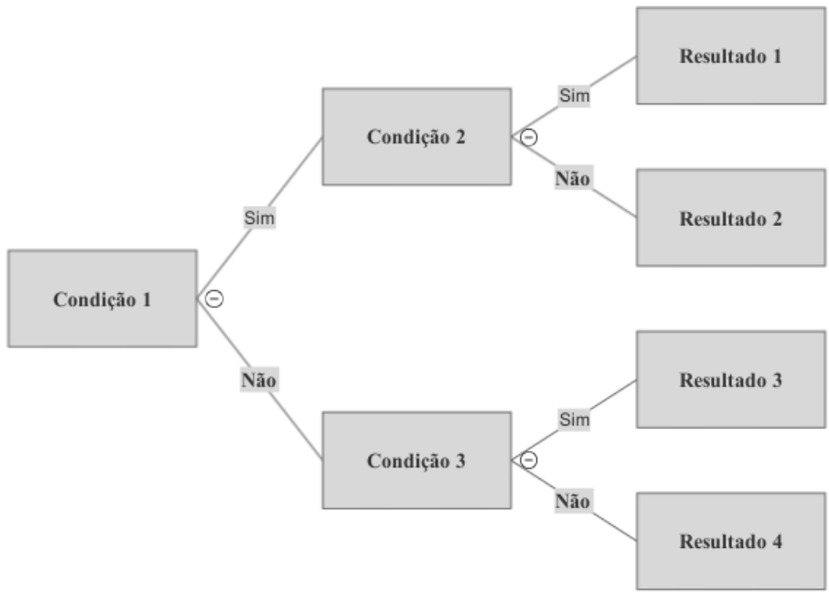
Um dos métodos de Aprendizagem de máquina é Árvore de Decisão. Esse método possui uma estrutura de nós e conexões e é equivalente a uma árvore invertida, iniciando da raiz e indo até as folhas para classificar uma instância. A cada nível de aprofundamento da árvore, é feita uma tomada de decisão até atingir um nó folha, onde é dada a classificação da instância (BARANAUSKAS, J. A. 2000).

2.2. ÁRVORE DE DECISÃO

Uma árvore de decisão é um algoritmo de aprendizado de máquina que possui uma estrutura semelhante a uma árvore, para avaliar as instâncias de entrada e classifica-las. Essa árvore é composta por vários nós, que são testes de características ou condições para as instâncias de entrada. Os nós da árvore são ligados através de ramos, que são possíveis resultados para as características do teste do nó. Cada ramo pode, a depender da análise de decisão do nó, fazer conexão com outro nó, ou levar a uma folha, indicando um resultado final da classificação e atribuído a essa folha.

A Figura 1 ilustra a estrutura de uma árvore de decisão do tipo binária, onde cada nó possui exatamente dois ramos e dois nós descendentes. Porém, as árvores de decisão utilizadas neste trabalho não se restringem à esse modelo binário. Elas podem possui mais ramos e nós descendentes.

Figura 1 - Estrutura de uma árvore de decisão



A forma mais adequada de testar as características das instâncias, é utilizando duas métricas: entropia e ganho. A entropia de um conjunto pode ser definida como sendo o grau de pureza desse conjunto, enquanto o ganho é a redução esperada na entropia de um conjunto, causada pela partição dos exemplos de acordo com um dado atributo A. Dado um conjunto de instâncias S, as equações da entropia e ganho desse conjunto são, respectivamente:

$$E(S) = -p_+ \log_2 p_+ - p_- \log_2 p_- \quad (1)$$

$$G(S,A) = E(S) - \sum_{v \in Valores(A)} \frac{S_v}{S} E(S_v) \quad (2)$$

A equação da entropia é definida pela diferença entre a negação da proporção de exemplos positivos, multiplicados pelo logaritmo na base dois dessa proporção positiva e a proporção de exemplos negativos, multiplicados pelo logaritmo na base dois dessa proporção negativa.

A equação do ganho é definida a partir do cálculo da entropia de um conjunto S. O atributo escolhido será o de maior ganho.

Abaixo estão descritos os classificadores presentes no Weka que foram utilizados neste trabalho:

C4.5 (QUINLAN, 1993) é um algoritmo cujo objetivo principal é gerar um modelo utilizando a estrutura de uma árvore de decisão. Algumas características desse algoritmo é que permite atributos desconhecidos, bem como atributos categóricos (ordinais e não-ordinais) e contínuos.

J48 baseia-se no C4.5 e gera uma árvore de decisão podada ou não. Uma poda na árvore pode aumentar a capacidade de generalização da árvore.

PART é um algoritmo que utiliza uma combinação de dois outros classificadores: C4.5 e RIPPER. O C4.5 utiliza o método de árvore de decisão. O PART cria regras de árvore de decisão e utiliza a técnica dividir para conquistar.

O algoritmo *Decision Table* é um algoritmo que gera tabela de decisão. Tabelas de decisão possuem uma representação simples, são facilmente lidas e interpretadas por humanos.

O *Decision Stump* (Árvore de Decisão Firme), apesar de ser um algoritmo bastante simples, pode ser útil para comparação com outros algoritmos de aprendizagem de máquina. As principais características do *Decision Stump* é o tempo de aprendizagem, que é equivalente à quantidade de instâncias de treinamento, e a memória ocupada, equivalente ao produto dos valores pela quantidade de classes. (STEINER *et al.,* 2004)

Random Forest tem sido bastante utilizado em diversos trabalhos dos mais diferentes temas, visto que é um bom algoritmo para conjunto de dados com muitos atributos e poucas instâncias (OSHIRO, 2013). Ele é um algoritmo que utiliza a combinação de predições de algoritmos de árvores de decisão e gera várias árvores. Uma das vantagens do *Random Forest*, é ser muito preciso e eficiente, mesmo para grandes bases de dados e apontar atributos importantes para a classificação.

2.3. PARTICLE SWARM OPTIMIZATION

Um dos algoritmos utilizados neste trabalho, é o PSO (do inglês, *Particle Swarm Optimization*) (KENNEDY, J., EBERHART, R., 1995) versão 2.0.8, que é um Algoritmo de inteligência artificial de Otimização baseado em Enxames de Partículas.

Esse algoritmo surgiu a partir de trabalhos observando o comportamento social dos pássaros e peixes e a dinâmica de movimentação em busca de alimentos. Cientistas observaram a sincronia desses comportamentos, e as rápidas mudanças que poderiam acontecer, e simularam esses modelos sociais, criando o algoritmo PSO (MIRANDA, PÉRICLES B.C, 2016). Como o PSO é um algoritmo inspirado na natureza, alguns consideram como Algoritmo Evolutivo.

O funcionamento do algoritmo PSO segue os seguintes passos: Um enxame de partículas é inicializado com seus atributos de posição e velocidade de forma aleatória. É feita uma avaliação de suas posições e velocidade de cada partícula e calculado o desempenho dessas partículas. Após isso, é calculada a melhor posição encontrada pelo enxame e pela partícula. Em seguida atualiza-se as velocidades e posições. E isso permanece num laço até que uma solução seja alcançada.

Segue abaixo o passo a passo do algoritmo PSO de forma mais detalhada:

Passo 1 - Inicializar um enxame de partículas

Passo 2 - Inicializar aleatoriamente as posições e velocidades do enxame

Passo 3 - Atualizar o pbest (melhor posição encontrada por partícula)

Passo 4 - Atualizar o gbest (melhor posição encontrada por enxame)

Passo 5 - Avaliar o enxame (Aplicar a função Fitness)

Passo 6 - Atualizar velocidade e velocidade de todas as partículas

Passo 7 - Se se as novas posições encontradas são melhores, Voltar ao Passo 3.

Passo 8 - Caso o critério de parada seja satisfeito, a solução do algoritmo é o gbest.

O PSO baseia-se em duas equações, sendo uma para posição da partícula e a outra para a velocidade da mesma:

Posição da partícula: $x_{k+1}^i = x_k^i + v_{k+1}^i$ (3)

Velocidade da partícula: $v_{k+1}^i = w_k v_k^i + c_1 r_1 \left(p_k^i - x_k^i\right) + c_2 r_2 \left(p_k^g - x_k^i\right)$ (4)

Onde:

$x_k^i = Posição\ da\ partícula\ i\ no\ instante\ k$

$v_k^i = Velocidade\ da\ partícula\ i\ no\ instante\ k$

$p_k^i = Melhor\ posição\ individual\ da\ partícula\ i\ no\ instante\ k$

$p_k^g = Melhor\ posição\ global\ de\ todas\ as\ posições\ individuais\ no\ instante\ k$

$w_k = Constante\ de\ inércia$

$c_1, c_2 = Peso\ do\ comportamento\ cognitivo\ e\ social\ da\ partícula, respectivamente$

$r_1, r_2 = Números\ aleatórios\ entre\ 0\ e\ 1$

Em outras palavras, a equação de posicionamento da partícula nos diz que a posição da partícula no próximo instante, é calculada através da posição da mesma no instante atual, acrescido da velocidade da mesma.

E a velocidade da mesma no instante seguinte é definida pela velocidade da partícula no instante atual multiplicado pela constante de inércia (chamada de fator de inércia), somado ao mínimo local (melhor posição da partícula) e ao mínimo global (melhor posição de todas as partículas). As variáveis aleatórias $r_1\ e\ r_2$ são variáveis de perturbação para evitar que o algoritmo pare em mínimos locais. O fator de inércia é importante, pois se a partícula se move para uma direção, essa inércia deve ser considerada, não podendo ser ignorada.

3. TRABALHOS RELACIONADOS

EDM significa mineração de dados educacionais (do inglês *Educational Data Mining*) e permite compreender de forma mais eficaz e adequada os alunos, e sua forma de aprendizagem, o papel do contexto na qual a aprendizagem ocorre, além de outros fatores. É uma área de pesquisa recente que surgiu com grande potencial para melhorar a qualidade do ensino (BAKER, ISOTANI E CARVALHO, 2011).

Ernani Gottardo et. al. (ERNANI GOTTARDO, CELSO KAESTENER, ROBINSON VIDA NORONHA, 2012) atinge índices de acerto de 76% na previsão do desempenho acadêmico dos estudantes, definindo um conjunto amplo de atributos, generalizável. Ele utilizou os classificadores *Random Forest* e MLP (*Multilayer Perceptron*), por obterem melhores resultados em outros trabalhos. Os alunos foram agrupados em 3 dimensões: perfil geral de uso do AVA, interação estudante-estudante, interação professor-professor. Como resultado da sua pesquisa, tem-se atributos definidos com grande chance de acerto de previsão, possibilitando a criação de sistemas inteligentes a serem inseridos nos AVA. Esses sistemas podem fazer o monitoramento dos estudantes, visando identificar aqueles em que o desempenho acadêmico não está satisfatório para que sejam tomadas providências por parte dos docentes.

Zaidah e Daliela Rusli (ZAIDAH IBRAHIM e DALIELA RUSLI, 2007) utilizam o CGPA (Média de pontos cumulativos) do aluno, após este se graduar e desenvolveu três modelos preditivos: Rede Neural Artificial, Árvore de Decisão e Regressão Linear, conseguindo mais de 80% de acerto na previsão do desempenho acadêmico dos alunos em todos os três modelos. Ele ainda mostrou que RNA (Rede Neural Artifical) é o melhor entre os três modelos.

Kotsiantis (KOTSIANTIS, 2012) desenvolve na linguagem Java, o protótipo de uma ferramenta chamado *Regression-tool*, a qual prevê o desempenho acadêmico dos alunos, utilizando a técnica de regressão. Foram utilizadas as seis técnicas de regressão mais comuns para comparação: *Model Trees*, Redes Neurais, Regressão Linear, Regressão Linear Ponderada e SVM (*Support Vector Machines*).

Da mesma forma que as técnicas de mineração de dados servem para detectar baixo desempenho acadêmico, também detecta possíveis candidatos com excelente rendimento acadêmico. Foi isso que Hamidah Jantan et. al. (HAMIDAH JANTAN, ABDUL RAZAK HAMDAN et al., 2010) propuseram, para instituições de nível superior, já que essas pessoas são bastantes valiosas. Para isso, ele determinou potenciais técnicas de classificação que fossem comuns em mineração de dados e pudessem ser utilizadas. A primeira técnica de classificação usada foi Redes Neurais, bastante popular em mineração de dados, com os algoritmos MLP e RBFC (*Radial Basic Function Network*). Foram utilizadas também Árvore de decisão, conhecida como dividir e conquistar, com os algoritmos C4.5 e *Random Forest*, além de vizinho mais próximo, baseada na distância entre os nós, com o algoritmo *K-Star*. O classificador com maior acurácia foi o C4.5. Como trabalhos futuros, foi sugerida a utilização de outros algoritmos de árvore de decisão para comparação dos resultados.

Marcelo (SANTANA, MARCELO A., 2015) após fazer o pré-processamento da base de dados, teve o algoritmo máquina de vetor de suporte (SVM: *Support Vector Machine*) com maior taxa de acerto, alcançou com a Medida-F uma taxa de 83% na base de dados de ensino presencial e 92% na base de dados de ensino à distância. Essas taxas foram possíveis pois houve ajuste fino no núcleo (*kernel*) do algoritmo SVM e posteriormente um algoritmo de *Grid-Search* para otimização dos parâmetros. No geral, dos trabalhos estudados, foi o que obteve a maior taxa de acerto.

Este trabalho propõe a comparação de diferentes algoritmos de aprendizagem de máquina que auxiliam a avaliação formativa dos estudantes, e a diferença da proposta para os trabalhos relacionados é a otimização dos seus parâmetros, através de algoritmo de aprendizagem de máquina voltado para otimização dos parâmetros.

4. METODOLOGIA DE DESENVOLVIMENTO

Este capítulo aborda o processo de desenvolvimento deste trabalho, fornecendo as informações necessárias para compreensão de todo o processo, desde o planejamento, até a execução e análise do mesmo.

A forma utilizada neste trabalho para melhorar a eficácia dos algoritmos foi otimizando os parâmetros de entrada, fazendo o uso de algoritmo de otimização de parâmetro, especificamente o PSO.

4.1. BASES DE DADOS

Este trabalho tem parceria entre a UFRPE (Universidade Federal Rural de Pernambuco) e a UFAL (Universidade Federal de Alagoas). Com este trabalho, espera-se uma troca de conhecimentos mútuos e experiência, além de possibilitar a avaliação e melhoria do ensino à distância nessas instituições.

Neste trabalho foram utilizadas duas bases de dados independentes: uma base é referente à alunos da modalidade de Ensino Presencial, contendo 83 instâncias, e a outra é referente à modalidade de ensino à Distância, contendo 177 instâncias. Essas bases foram obtidas através de dados acadêmicos fornecidos pelos sistemas utilizados na Universidade Federal de Alagoas (UFAL) e cedidas para este trabalho. Outras duas bases foram criadas a partir dessas, removendo os atributos de notas de forma a testar a eficácia dos algoritmos sem elas. Nas tabelas abaixo estão listados todos os atributos utilizados em cada base de dados:

Tabela 1 - Atributos selecionados na modalidade ensino Presencial

Atributos	Descrição
Problemas	Quantidade de Exercícios
1a Avaliação	Nota da Primeira Avaliação
Corretos	Total de Exercícios Corretos
Submissões	Total de submissões
1a Semana	Nota da primeira semana
2a Semana	Nota da segunda semana
3a Semana	Nota da terceira semana
Estado Civil	Estado Civil do Estudante
Sexo	Sexo do Estudante
Idade	Idade do Estudante
Status	Status (Aprovado ou Reprovado)

Fonte: Santana, Marcelo A. (2015)

Tabela 2 - Atributos selecionados na modalidade ensino Presencial sem as Notas

Atributos	Descrição
Problemas	Quantidade de Exercícios
Corretos	Total de Exercícios Corretos
Submissões	Total de submissões
Estado Civil	Estado Civil do Estudante
Sexo	Sexo do Estudante
Idade	Idade do Estudante
Status	Status (Aprovado ou Reprovado)

Fonte: O Autor

Tabela 3 - Atributos Selecionados na modalidade de ensino à Distância

Atributos	Descrição
1a Avaliação	Nota da primeira Avaliação
2a Semana	Nota da segunda semana
3a Semana	Nota da terceira semana
4a Semana	Nota da quarta semana
5a Semana	Nota da quinta semana
Blog	Quantidade de postagens e visualização no blog
Forum	Quantidade de postagens e visualizações no fórum
Acessos	Quantidade de acessos ao AVA
Assign	Quantidade de arquivos enviados e baixados
Cidade	Cidade do Estudante
Message	Quantidade de mensagens enviadas
Wiki	Quantidade de mensagens enviadas
Sexo	Sexo do estudante
Estado	Estado Civil do estudante
Idade	Idade do estudante
Status	Status da disciplina (Aprovado ou Reprovado)

Fonte: Santana, Marcelo A (2015)

Tabela 4 - Atributos Selecionados na modalidade de ensino à Distância sem Notas

Atributos	Descrição
Blog	Quantidade de postagens e visualização no blog
Forum	Quantidade de postagem e visualização no fórum
Acessos	Quantidade de acessos ao AVA
Assign	Quantidade de arquivos enviados e baixados
Cidade	Cidade
Message	Quantidade de mensagens enviadas
Wiki	Quantidade de mensagens enviadas
Sexo	Sexo do estudante
Estado	Estado Civil do estudante
Idade	Idade do estudante
Status	Status da disciplina (Aprovado ou Reprovado)

Fonte: O Autor

4.2. ALGORITMOS DE APRENDIZAGEM DE MÁQUINA

Neste trabalho foi utilizado o Weka (Waikato Environment for Knowledge Analysis) (HALL *et al.,* 2009) (FRANK *et al.,* 2010), versão 3.8.1., que possui um conjunto de algoritmos de aprendizado de máquina. Utilizando a linguagem Java, com o auxílio do Weka, que já possui implementados os algoritmos necessários, foi calculada a eficácia dos seguintes classificadores: J48, PART*, Decision Table, Decision Stump* e *Random Forest*. Esses algoritmos foram submetidos aos testes, utilizando seus parâmetros padrões.

Para uma melhor validação, em todos os conjuntos de dados, foi utilizada a técnica validação cruzada. A técnica de validação cruzada é a criação de vários subconjuntos distintos a partir de uma base de dados, que auxiliarão na estimativa de parâmetros do modelo (conjunto de treinamento) e os outros subconjuntos (conjunto de teste ou validação) que serão utilizados na validação do modelo. (P.B.C. De Miranda, 2013). A forma escolhida para realizar o particionamento dos dados foi o *K-Fold*.

O particionamento do tipo *K-Fold* consiste na divisão da base de treinamento em K subconjuntos. Dos K subconjuntos, um é selecionado para teste ou validação do modelo e o restante é utilizado no treinamento. Efetua-se a validação cruzada, utilizando cada um dos subconjuntos como conjunto de treinamento para o modelo uma e somente uma vez. (HAN J., KAMBER, M. PEI, J., 2011)

Neste trabalho, a métrica para comparar a eficácia do algoritmo foi feita através da Medida-F, o qual é explicada nas equações abaixo. Foram testados diversos algoritmos, e

selecionados os dois melhores para terem seus parâmetros de entradas otimizados, na próxima etapa.

Termos utilizados nas equações abaixo:

True positive (TP): Amostras positivas corretamente classificadas.

True negative (TN): Amostras negativas corretamente classificadas.

False positive (FP): Amostras negativas erroneamente classificadas como positivas.

False negative (FN): Amostras positivas erroneamente classificadas como negativas.

Precisão é a porcentagem de amostras positivas corretamente classificadas sobre o total de amostras classificadas como positivas.

$$Precisão = \frac{TP}{TP + FP} \quad (5)$$

Recall é a porcentagem de amostras positivas corretamente classificadas sobre o total de amostras positivas.

$$Cobertura = \frac{TP}{TP + FN} \quad (6)$$

Medida-F é a média harmônica entre precisão e cobertura.

$$Medida - F = 2 \frac{Precisão * Cobertura}{Precisão + Cobertura} \quad (7)$$

Acurácia é a porcentagem de amostras positivas e negativas classificadas corretamente sobre todas as amostras

$$Acurácia = \frac{TP + TN}{TP + TN + FP + FN}$$

É de grande importância pedagógica a execução dos testes em ambos os cenários: bases de dados presencial e à distância, incluindo as notas dos alunos e outros cenários com bases derivadas a partir delas, porém sem os atributos que contém notas de exercícios e avaliações dos alunos. Esses resultados em ambos os cenários podem auxiliar o corpo docente e gestão acadêmica das instituições.

As tabelas 5 e 6 mostram a Medida-F dos algoritmos, a fim de serem selecionados os dois melhores para serem otimizados, enquanto a tabela 7 apresenta a média da Medida-F dos algoritmos.

Tabela 5 - Resultados das bases completas, contendo os atributos de notas

Algoritmo	Presencial	Distância
J48	82,00%	88,70%
PART	83,30%	90,40%
Decision Table	81,50%	90,90%
Decision Stump	84,10%	91.50%
Random Forest	90,40%	92,10%

Fonte: O Autor

Tabela 6 - Resultados das bases que não contém as notas

Algoritmo	Presencial	Distância
J48	72,70%	74,60%
PART	71,50%	72,90%
Decision Table	70,50%	74,60%
Decision Stump	70,50%	75,20%
Random Forest	72,70%	83,00%

Fonte: O Autor

Tabela 7 - Média aritmética das tabelas 5 e 6

Algoritmo	Média
J48	79,50%
PART	79,53%
Decision Table	79,38%
Decision Stump	80,33%
Random Forest	84,55%

Fonte: O Autor

Na tabela 5, os algoritmos que obtiveram o maior resultado foram *Random Forest,* com 86,90% de acerto na base de ensino presencial e o *Decision Stump* um pouco acima na base de ensino à distância, com 91,53% em relação ao segundo colocado *Random Forest*, com 90,40%.

Em contrapartida, na Tabela 6, o *Random Forest* obteve o maior resultado nos dois cenários de ensino presencial e à distância, atingindo 73,81% e 80,23% respectivamente. Enquanto isso, o *Decision Stump* obteve o pior resultado de todos algoritmos testados: 71,19%.

O algoritmo *Random Forest* se mostrou melhor no cenário de ensino Presencial, mesmo após ter os atributos de notas removidos. De acordo com a tabela 7, o algoritmo *Decision Stump*,

mesmo com o baixo desempenho acadêmico em um dos cenários de ensino à distância, foi o algoritmo que obteve melhor média de Medida-F, depois do *Random Forest.* Por esse motivo, a otimização de parâmetros se restringiu ao escopo desses dois melhores algoritmos.

4.3. OTIMIZAÇÃO DOS PARÂMETROS

Nessa etapa do trabalho, para maximizar o ganho dos algoritmos *Random Forest* e *Decision Stump*, foram otimizados os parâmetros listados abaixo:

Tabela 8 - Descrição dos parâmetros do Random Forest

Parâmetro	Descrição
BagSizePercent	Porcentagem do conjunto de treinamento
BatchSize	Tamanho do lote
MaxDepth	Profundidade da árvore
NumDecimalPlaces	Número de casas decimais
NumExecutionSlots	Número de threads utilizadas para construção do modelo
NumIterations	Número de iterações
NumFeatures	Número de recursos na seleção aleatória
Seed	Semente utilizada para aumentar a aleatoriedade da floresta

Fonte: O Autor

Tabela 9 - Descrição dos parâmetros do Decision Stump

Parâmetro	Descrição
BatchSize	Tamanho do lote
NumDecimalPlaces	Número de casas decimais

Fonte: O Autor

Esses parâmetros possibilitam o aumento da eficácia de um modelo, bem como a melhoria no treinamento do mesmo.

Como a inicialização do algoritmo PSO é feita de forma aleatória, de forma a garantir os resultados obtidos, cada modelo do algoritmo foi executado dez vezes e foi calculada a média aritmética da Medida-F entre as dez execuções. Um modelo de execução de algoritmo é composto de: Base de Dados, Algoritmo de classificação e Método de Atualização de Partícula.

4.4. RESULTADOS ALCANÇADOS COM A OTIMIZAÇÃO

A seguir são apresentados os resultados para cada modelo de algoritmo. Ou seja, foi obtida a eficácia para cada base de dados, cada algoritmo de classificação, cada método de atualização de partícula. A métrica utilizada para testar a eficácia dos algoritmos foi Medida-F.

Os gráficos a seguir mostram a Medida-F sendo utilizada como eficácia, em relação aos métodos de atualização de partículas do algoritmo de otimização PSO. A implementação do algoritmo PSO utilizada neste trabalho disponibilizava três métodos de atualização de partículas: *Simple Particle Update, RandomBy Particle Update* e *Fully Particle Update*. Cada uma dessas três variações de métodos de atualização de partículas foi executada dez vezes a fim de ser obtido o melhor método para cada base de dados.

Nestes gráficos, cada barra representa, respectivamente:

RF máximo: Representa o maior valor da Medida-F atingido pelo algoritmo *Random Forest* após otimização dos parâmetros, nas dez execuções do PSO.

RF otimizado: Representa a média da Medida-F nas dez execuções do algoritmo *Random Forest* após otimização dos parâmetros.

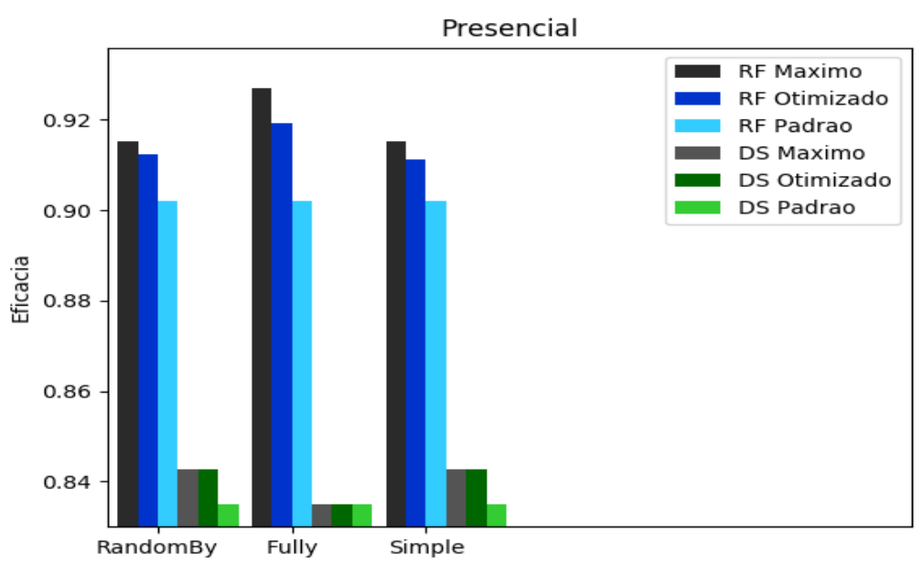
RF padrão: Representa a Medida-F do classificador *Random Forest* antes de ser otimizado, sem ajustes de parâmetros.

DS máximo: Representa o maior valor da Medida-F atingido pelo algoritmo *Decision Stump* após otimização dos parâmetros, nas dez execuções do PSO.

DS otimizado: Representa a média da Medida-F nas dez execuções do algoritmo *Decision Stump* após otimização dos parâmetros.

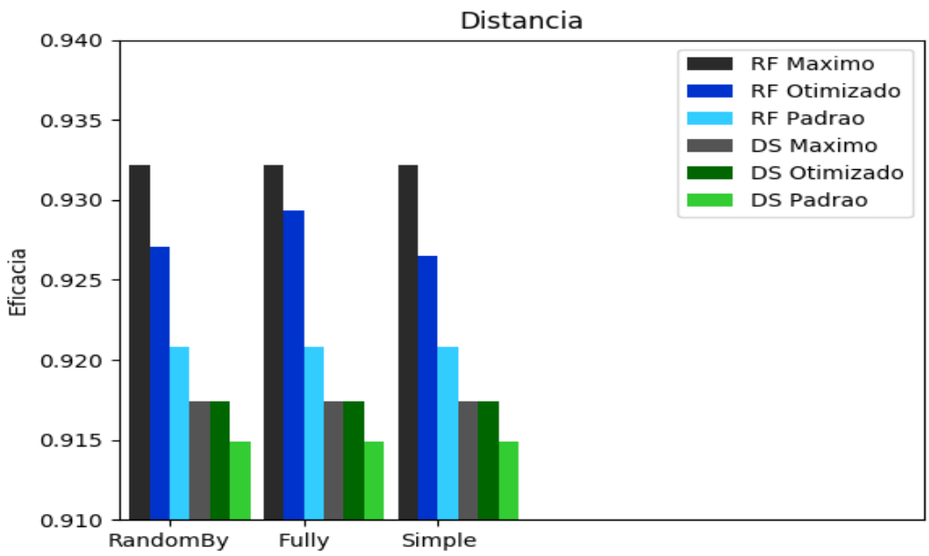
DS padrão: Representa a Medida-F do classificador *Decision Stump* antes de ser otimizado, sem ajustes de parâmetros.

Figura 2 - Eficácia na base de ensino Presencial



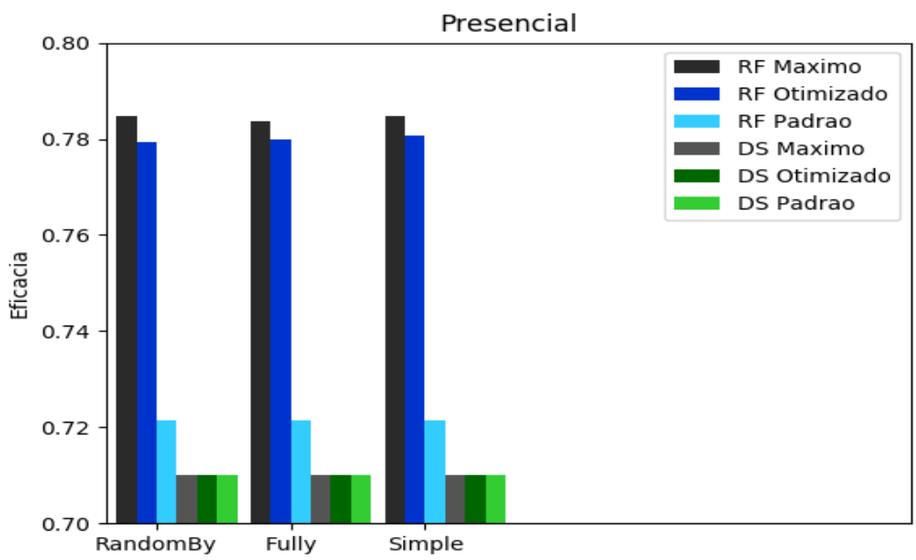
Neste cenário de ensino Presencial, utilizando a base completa (com notas dos alunos), os dois algoritmos obtiveram ganho após otimização. Em todas as execuções do PSO, o método de atualização de partículas que obteve maior ganho com o *Random Forest* foi o *Fully Particle Update* aumentando a Medida-F de 1,73 a 2,50 pontos percentuais. Para o *Decision Stump*, tanto o método *Random By Particle Update* quanto o *Simple Particle Update* obtiveram ganho idêntico de Medida-F: 0.76 pontos percentuais. Utilizando o método *Fully Particle Update* não houve aumento da Medida-F com o *Decision Stump*.

Figura 3 - Eficácia na base de ensino à Distância



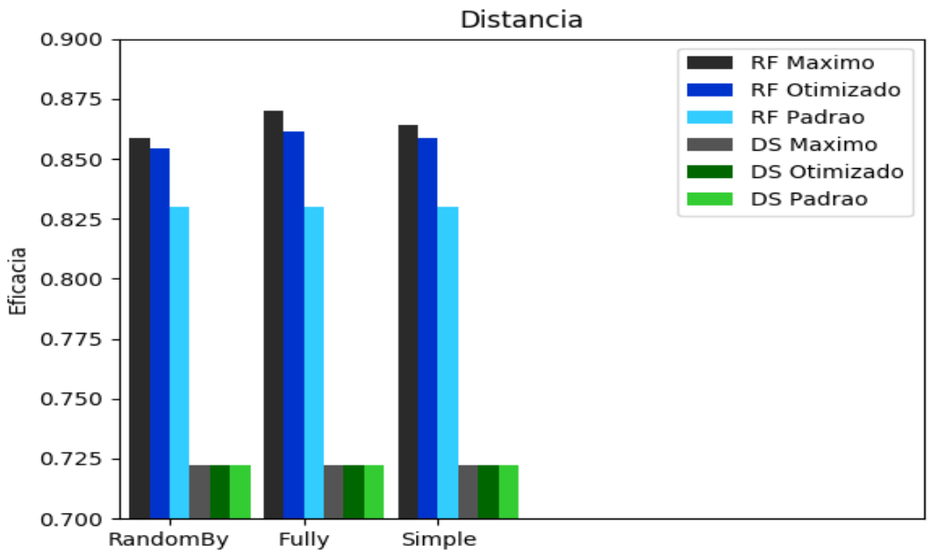
No cenário de ensino à distância, utilizando a mesma base completa (com notas dos alunos), também houve ganho nos dois algoritmos após otimização. O método de atualização de partículas com maior ganho no algoritmo *Random Forest* novamente foi o *Fully Particle Update* o qual aumentou a Medida-F de 0,85 a 1,13 pontos percentuais. Já o *Decision Stump*, o aumento foi de 0,24 pontos percentuais em todos os métodos de atualização de partículas.

Figura 4 - Eficácia na base de ensino Presencial sem Notas



No cenário de ensino presencial sem notas é bastante curioso, pois mesmo sem os atributos de notas dos alunos, o método *Simple Particle Update* obteve um ganho de 5,93 a 6,32 pontos percentuais na Medida-F do algoritmo *Random Forest. A Medida-F no Decision Stump* permaneceu em 71,01% antes e depois da otimização, independentemente do método de atualização de partículas do PSO.

Figura 5 - Eficácia na base de ensino à Distância sem Notas



Neste último cenário, modalidade ensino à distância sem as notas dos alunos, a Medida-F cresceu 3,13 pontos percentuais em média com o método *Fully Partic*le *Update* utilizando o algoritmo *Random Forest.* Já no *Decision Stump*, a Medida-F se manteve em 72,23%.

Em todos os casos testados observa-se que o algoritmo *Random Forest* atinge uma maior eficácia após otimização em relação aos valores padrão, ainda que o valor máximo alcançado em algumas situações, tenha sido superior à média de todas as execuções. Isso acontece, pois, a inicialização das partículas do algoritmo PSO é feita de forma aleatória. Para garantir uma maior confiabilidade dos resultados, foram feitas dez execuções e calculada a média da Medida-F de todas as execuções, que é exatamente o valor médio otimizado. O ganho do *Decision Stump* é menor, pois o mesmo só teve dois parâmetros de entrada otimizados, enquanto no *Random Forest* foram oito. Além disso, quando criadas as novas bases de dados, mesmo sem os atributos de notas dos alunos, também houve um aumento na eficácia do classificador *Random Forest*.

O *Decision Stump* ao remover as notas dos alunos, manteve o valor da Medida-F, mostrando a relevância dos atributos de notas dos alunos para a classificação.

Podemos citar diversas formas que esses algoritmos otimizados podem auxiliar os gestores acadêmicos, com alta taxa de precisão:

- Informar antecipadamente alunos com baixo rendimento acadêmico
- Diminuir a possibilidade de reprovação e evasão escolar, coletando os resultados dos experimentos e executando ações pedagógicas nos alunos candidatos à reprovação

- Mostrar estatísticas e formar ranking de alunos
- Identificar assuntos e disciplinas em que há maiores índices de reprovação
- Identificar candidatos com maior rendimento, podendo ser utilizado para monitoria, olimpíadas, dentre outros benefícios.

5. CONCLUSÃO

O trabalho desenvolvido foi motivado pela alta taxa de reprovação e evasão no Brasil, que podem gerar vários tipos de prejuízos. Um grande desafio é detectar antecipadamente alunos candidatos ao insucesso para que sejam feitos os acompanhamentos e intervenções pedagógicas necessárias.

Com a grande quantidade de dados gerados a partir das interações dos alunos com o AVA, é importante existir um algoritmo eficaz, capaz de fazer a indicação de alunos candidatos ao insucesso.

A principal proposta deste trabalho foi aumentar a eficácia de algoritmos de aprendizagem de máquina que auxiliam a avaliação formativa, possibilitando docentes e gestores acadêmicos uma antecipação na predição de alunos com baixo desempenho acadêmico e indícios de reprovação ou evasão. Essa melhoria foi possível através da utilização de algoritmos de otimização.

Outro destaque foi a utilização de duas bases de dados adicionais, complementando as outras duas, das modalidades de ensino presencial e à distância, com todas as notas dos alunos removidas.

Este trabalho abre oportunidade para que pesquisadores da área de inteligência artificial possam efetuar diversas melhorias. Dentre essas melhorias, podemos citar:

- Utilização de outras bases de dados com volume de dados maiores.
- Utilização de outros atributos nas bases de dados, como por exemplo, assiduidade e atributos textuais.
- Maior estudo na escolha de outros algoritmos classificadores para terem seus parâmetros otimizados.
- Maior estudo na escolha de outro algoritmo de otimização dos parâmetros, em substituição ao PSO

REFERÊNCIAS

BARANAUSKAS, J. A.; MONARD (2000), “Reviewing Some Machine Learning Concepts and Methods. Relatórios Técnicos do ICMC/USP, v. 102, 2000”

BLOOM, B.S., HASTINGS, J. T., MADAUS, G. F. (1993) “Manual de avaliação formativa e somativa do aprendizado escolar. São Paulo: Pioneira”

DILLENBOURG P. S., SCHNEIDER D. (2002) “Virtual learning environments,” in Proceedings of the 3rd Hellenic Conference on Information & Communication Technologies in Education, pp. 3–18.

ERNANI GOTTARDO (2012) “Estimativa de desempenho acadêmico de estudantes em um AVA utilizando técnicas de mineração de dados.”

GOTTARDO E., KAESTENER C., NORONHA R. V. (2012) “Previsão de desempenho de estudantes em cursos EAD utilizando Mineração de Dados: uma Estratégia Baseada em Séries Temporais. ”

FAYYAD, U.M., PIATESKY-SHAPIRO, G., SMYTH, P. (1996) “The KDD Process for Extracting Useful Knowledge from Volumes of Data”. Comunications of ACM, vol. 39, no. 11, p. 27-34.

FRANK, E., HALL, M., HOLMES, G., KIRKBY, R., PFAHRINGER, B., WITTEN, I. H., E TRIGG, L. (2010) “Weka-a machine learning workbench for data mining. Data Mining and Knowledge Discovery Handbook, pp. 1269–1277”

HALL, M., FRANK, E., HOLMES, G., PFAHRINGER B., REUTEMANN P., E WITTEN I. H. (2009) ”The weka data mining software: an update, ACM SIGKDD explorations newsletter, vol. 11, no. 1, pp. 10–18”

HAMIDAH JANTAN, ABDUL RAZAK HAMDAN, ZULAIHA ALI OTHMAN (2010) “Classification and Prediction of Academic Talent Using Data Mining Techniques”

HAN, J.; KAMBER, M.; PEI, J. (2011) “Data Mining: Concepts and Techniques. 3rd Ed.”

HO, TIN KAM (1995) “Random Decision Forests”

JAIN, POOJA (2015) “Virtual learning environment." International Journal in IT & Engineering 3.5 (2015): 75-84.”

KENNEDY, J.; EBERHART, R. (1995). “Particle Swarm Optimization. Proceedings of IEEE International Conference on Neural Networks. IV. Pp. 1942-1948”

KOTSIANTIS, S. B. “Use of machine learning techniques for educational proposes: a decision support system for forecasting students’ grades. Artificial Intelligence Review, v. 37, n. 4, p. 331-344, 2012”

MIRANDA, P. B. C. (2013) “Arquitetura híbrida para otimização multi-objetivo de SVMs”

MIRANDA, P. B. C. (2016) “Uma hiper-heurística híbrida para a otimização de algoritmos”

MARI, M.M; OPRIME, P.C; MARI, C.M.M; COSTA, M.A.B. (2011) “Análise da evasão e reprovação de alunos em cursos à distância: Um estudo empírico”

M.C. MONARD, J.A. BARANAUSKAS (2003) “Sistemas Inteligentes-Fundamentos e Aplicações”

MÜLBERT, A.L; GIRONDI, A; PEREIRA, A.T.C; NAKAYAMA, M.K, (2011) “A interação em ambientes virtuais de aprendizagem: motivações e interesses dos alunos”

OSHIRO, THAIS MAYUMI (2013) “Uma abordagem para a construção de uma única árvore a partir de uma Random Forest para classificação de bases de expressão gênica

P.B.C. DE MIRANDA (2013) “Arquitetura híbrida para otimização multi-objetivo de SVMs”

QUINLAN, J. R. (1993) “C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers”

RICARTE, I. L. M.; FALCI JUNIOR (2011) “A Methodology for Mining Data from Computer-Supported Learning Environments.”

ROLIM, V.B (2016) “Método Supervisionado para identificação de dúvidas em postagens de fóruns educacionais”

ROMERO, CRISTOBAL; VENTURA, SEBASTIAN (2017) “Educational data science in massive open online courses. WILEY INTERDISCIPLINARY REVIEWS-DATA MINING AND KNOWLEDGE DISCOVERY, v. 7, n. 1, 2017”

SANTANA, MARCELO A (2015) “Um estudo comparativo das técnicas de predição na identificação de insucesso acadêmico dos estudantes durante cursos de programação introdutória”

STEINER, M.T.A.; SOMA, N.Y.; SHIMIZU, T.; NIEVOLA, J.C.; LOPES, F.; SMIDERLE, A. (2004) “Data Mining como Suporte à Tomada de Decisões - uma Aplicação no Diagnóstico Médico”

WEISS, S. M. e KULIKOWSKI, C. A. (1991) “Computer Systems that Learn: Classification and Prediction Methods from Statistics, Neural Nets, Machine Learning, and Expert Systems. Morgan Kaufmann Publishers Inc., São Francisco, CA, USA, 1991.”

ZAIDAH IBRAHIM e DALIELA RUSLI (2007) “Predicting Students Academic Performance: Comparing Artificial Neural Network, Decision Tree and Linear Regression”