

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/316648085>

Uma abordagem para a previsão de desempenho de alunos de Computação em disciplinas de programação

Conference Paper · December 2015

CITATION

1

READS

94

3 authors, including:



Túlio Pascoal

University of Luxembourg

6 PUBLICATIONS 9 CITATIONS

SEE PROFILE

Uma abordagem para a previsão de desempenho de alunos de Computação em disciplinas de programação

Túlio Albuquerque Pascoal
Centro de Informática
Universidade Federal da Paraíba
(UFPB) – João Pessoa – PB, Brasil
tulioapascoal@gmail.com

Daniel Miranda de Brito
Programa de Pós-Graduação em
Informática
Universidade Federal da Paraíba
(UFPB) – João Pessoa – PB, Brasil
britmb@gmail.com

Thaís Gaudencio do Rêgo
Centro de Informática
Universidade Federal da Paraíba
(UFPB) – João Pessoa – PB, Brasil
gaudenciothais@gmail.com

ABSTRACT

Programming courses of Computer Science degrees are essential for the professional formation of area's students. Nevertheless, the low performance of students in these courses has concerned universities' coordinators and educators, given that this fact may be related to the rise of droppings in the area's courses. In this paper, we propose a method to identify students more prone to fail in the first programming courses taken by the students in a public Brazilian university's Informatics Center. Results show that is possible to identify these students by gathering their results on the admission exams and on prerequisite courses, obtaining success rates above 80%.

RESUMO

As disciplinas de programação de cursos de Computação são essenciais para a formação profissional dos estudantes da área. No entanto, o baixo rendimento de estudantes nestas disciplinas tem preocupado educadores e gestores das universidades, já que este fato pode ainda estar relacionado ao aumento da evasão nos cursos da área. Neste trabalho, propomos um método de identificação de estudantes propensos ao fracasso nas disciplinas iniciais de programação de cursos do Centro de Informática de uma universidade pública brasileira. Resultados mostram que é possível identificar estes estudantes usando informações de desempenho na prova de ingresso e nas disciplinas pré-requisito com taxas de acerto superiores a 80%.

Categories and Subject Descriptors

H.2.8 [Database Applications]: Data Mining; I.2.1 [Artificial Intelligence]: Applications and Expert Systems; K.3.2 [Computers and Education]: Computer science education.

General Terms

Design, Experimentation, Human Factors.

Keywords

Educational Data Mining, Machine Learning, Prediction, Students' performance, Academic failure.

1. INTRODUÇÃO

Nas universidades brasileiras é comum que estudantes de cursos de Computação apresentem dificuldades no aprendizado de disciplinas de programação, acarretando aumento das taxas de reprovação e evasão das disciplinas da área [14]. Esse fato se agrava quando consideramos que essas disciplinas são essenciais para uma boa formação profissional do aluno da área [6], logo é interessante prover meios de identificar os fatores responsáveis pelas dificuldades dos alunos, a fim desenvolver estratégias para superá-las. Vários trabalhos na literatura dedicam-se neste estudo, tais como: [6], [8], [11], [12], [18] e [20].

Fatores que contribuem para um baixo rendimento e dificuldade em assimilar conteúdos das disciplinas de programação foram investigados por diversos autores, dos quais podemos destacar: Domínio de interpretação e compreensão de texto [7], habilidade de resolução de problemas [7,18], embasamento matemático [9,18,20] e experiência prévia em programação e computadores e softwares (Games, Internet, etc) [20]. Esses, entre outros fatores, estão diretamente relacionados com os índices de evasão e reprovação de disciplinas nos cursos na área de Computação [14], que ainda apresentam taxas de evasão acima da média, se comparados com cursos de outras áreas [19].

Diante desta problemática, técnicas de Mineração de Dados Educacionais (Educational Data Mining, EDM) podem ser aplicadas com intuito de aprimorar o processo de ensino-aprendizagem nas disciplinas de programação. EDM consiste na aplicação de técnicas de Mineração de Dados (Data Mining) no campo educacional, que por sua vez, segundo [2] é composta por um conjunto de metodologias com o objetivo de extrair conhecimentos não explícitos e não conhecidos previamente de um conjunto de dados. A descoberta de conhecimento possui grande valia em diversas situações, por exemplo, ajudando em tomadas de decisões de universidades [21], empresas e governos, baseando-se no novo conhecimento adquirido a partir da aplicação de técnicas de Data Mining nos dados. EDM vem ganhando força e popularidade entre instituições de ensino, que buscam melhorar a qualidade do seu ensino e desempenho de seus alunos. Aplicando técnicas de Data Mining, podemos analisar dados educacionais e buscar por padrões e relações desses dados com o objetivo de detectar características particulares do mesmo, gerando conhecimento, e a partir daí realizar previsões de desempenho de um estudante [1].

Motivados pela importância das disciplinas de programação na formação profissional de estudantes de cursos de Computação e da popularização da área de EDM, propomos neste artigo um método para previsão de desempenho dos estudantes nas principais disciplinas de programação, utilizando, como base, os seus resultados nas provas de ingresso na instituição de ensino e,

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

TISE'15, December 1–3, 2015, Santiago, Chile.

Copyright 2015 ACM 1-58113-000-0/00/0010 ...\$15.00.

para alguns casos, o desempenho dos mesmos em disciplinas pré-requisitos para as disciplinas avaliadas. A viabilidade do método é estudada a partir de testes feitos com um conjunto de registros de estudantes dos cursos de Ciência da Computação, Engenharia da Computação e Matemática Computacional, do Centro de Informática de uma universidade brasileira.

O restante do artigo está organizado da seguinte forma: Na seção 2, apresentam-se os trabalhos relacionados; na seção 3 descreve-se a metodologia empregada no desenvolvimento do trabalho; os resultados são apresentados e discutidos na seção 4; na seção 5 mostram-se as conclusões e trabalhos futuros e, por fim, as referências utilizadas no trabalho são exibidas.

2. TRABALHOS RELACIONADOS

A Mineração de Dados Educacionais é um campo de pesquisa bastante promissor e tem apresentado um grande crescimento no Brasil e no Mundo nos últimos anos [1]. [17] divide EDM em oito categorias principais, sendo uma delas, a predição de desempenho de alunos, que é uma das mais antigas e populares aplicações de Mineração de Dados Educacionais. Predição de desempenho de alunos pode ser aplicada por uma boa variedade de técnicas de Data Mining, como Redes Neurais, Redes Bayesianas e análises de Regressão e Classificação. Além de possuir uma vasta aplicabilidade, desde a predição de desempenho acadêmico de estudantes universitários até a previsão da quantidade de erros um estudante irá cometer num teste [17].

Autores tentam entender os fatores responsáveis pelas dificuldades enfrentadas pelos alunos nas disciplinas de programação de um curso de Computação, [15] e [16] analisam as dificuldades enfrentadas por alunos durante a aprendizagem destas disciplinas, relacionando-as com diferentes metodologias de ensino, de modo a desenvolver um melhor ambiente de ensino e aprendizagem entre professores e alunos. [9] criou um projeto chamado “Ensino de Lógica de Programação e Estrutura de Dados para Alunos do Ensino Médio”, para recrutar alunos do Ensino Médio de escolas brasileiras, interessados em programação, com o objetivo de iniciar, o mais cedo possível, o contato de alunos com os conceitos de raciocínio lógico por meio de resolução de problemas utilizando algoritmos e estrutura de dados, buscando capacitar o aluno com as habilidades necessárias na programação e motivá-lo a continuar na área, consequentemente, diminuindo a quantidade de evasão e reprovação em disciplinas de Programação no futuro.

Usando técnicas de Data Mining, como Árvore de Classificação e Regressão, [13] analisa dados sociais, demográficos e do ambiente de estudo de estudantes da disciplina de Sistemas de Informações da Open Polytechnic of New Zealand entre os anos de 2006 e 2009, a fim de detectar quais fatores são os mais importantes, entre os demais, para caracterizar um aluno de sucesso (que será aprovado na disciplina) ou de insucesso (que será reprovado na disciplina). Ele conclui que a etnia, o curso (Administração ou Ciências Aplicadas, cursos avaliados em seu estudo) e o semestre que o curso foi oferecido, são os principais fatores para a previsão de alunos que obterão sucesso na disciplina. Alguns trabalhos de EDM vêm sendo aplicados em cursos de Computação, como forma de previsão de desempenho de alunos. [4] analisam a possibilidade de predição de desempenho de alunos do primeiro período nas disciplinas de cálculo, com base no seu desempenho no vestibular. Os autores argumentam que essas informações possuem grande valor

acadêmico, já que o fracasso nas disciplinas iniciais do curso está muitas vezes relacionado com o aumento nas taxas de evasão nos cursos. [3] também propõem um classificador para analisar as notas de alunos nos cursos de Computação, com base em dados sociais, econômicos e de desempenho no ensino médio, usando o classificador *Naive Bayes*.

3. METODOLOGIA

3.1 Obtenção dos dados e pré-processamento

Os dados utilizados neste trabalho foram obtidos através da Superintendência de Tecnologia da Informação (STI) da Universidade Federal da Paraíba (UFPB) e contém registros anônimos de alunos dos três cursos de graduação (Ciência da Computação, Engenharia da Computação e Matemática Computacional) do Centro de Informática, da mesma instituição, do ano de 1995 até 2014. Os registros são compostos de informações de desempenho dos alunos nas matérias do vestibular e as suas notas em todas as disciplinas cursadas (Neste trabalho só são analisadas as disciplinas de programação). Como as informações foram obtidas em tabelas separadas, fez-se necessária uma etapa de pré-processamento, na qual se utilizaram as ferramentas Pentaho e PostgreSQL para combinar as informações a fim de utilizá-las na ferramenta Weka, introduzida na Seção 3.3.

A Tabela 2 contém informações de todos os alunos que ingressaram na universidade no período descrito e a Tabela 3 contém informações das disciplinas cursadas pelos alunos, bem como a sua situação final na disciplina (Reprovado, Aprovado, Interrompido ou Reprovação por Falta). Neste trabalho não foram consideradas instâncias de alunos que interromperam a disciplina ou reprovaram por falta. O cruzamento das informações das tabelas foi realizado de acordo com a matrícula do aluno, com propósito de agrupar em um só conjunto de dados, todas as informações necessárias para o processamento pela ferramenta de Data Mining. Ao término desta etapa, realizou-se outra etapa com intuito de identificar a presença de ruídos e ausência de informações importantes. Instâncias que apresentaram algumas destas características foram eliminadas. O número total de instâncias resultantes para cada teste é apresentado na Tabela 1.

Tabela 1. Resultados obtidos na previsão de desempenho na disciplina “Introdução a Programação”

| Disciplina | Instâncias da classe “Insucesso” | Instâncias da classe “Sucesso” |
|-----------------------------|----------------------------------|--------------------------------|
| Introdução a Programação | 212 | 846 |
| Linguagem de Programação I | 143 | 739 |
| Linguagem de Programação II | 94 | 514 |
| Estruturas de Dados | 51 | 167 |

3.2 Modelo proposto

Neste trabalho, propomos um modelo para identificar, precocemente, alunos propensos ao insucesso nas disciplinas de programação em cursos de Computação. Utiliza-se como base para construção do método, o fluxograma dos cursos de Computação avaliados, mas espera-se que o mesmo possa ser adaptado a cursos de outras instituições de ensino com alguns ajustes. Realizamos a previsão de desempenho dos alunos em quatro disciplinas obrigatórias dos três primeiros períodos do curso, são elas: “Introdução da Programação”, “Linguagem de Programação I”, “Linguagem de Programação II” e “Estruturas de Dados”. Acreditamos que a identificação precoce dos estudantes propensos a insucesso nestas disciplinas auxilie professores e coordenadores de curso no desenvolvimento de ações que visem reduzir a quantidade de reprovações e, possivelmente, evasão por fracasso nas disciplinas iniciais, ajudando a aprimorar os sistemas de ensino das universidades.

A situação do aluno nas disciplinas (classe) é definida a partir da sua média final. Alunos que obtiveram nota menor que 5,0 (nota mínima para aprovação em uma disciplina na instituição) são considerados como sendo da classe “Insucesso” e os alunos que alcançaram pelo menos 5,0 são considerados da classe “Sucesso”. As informações utilizadas como atributos de entrada para o problema de classificação diferem-se de acordo com a disciplina analisada, pois elas são ministradas em períodos diferentes do curso. A Tabela 2 mostra os atributos de entrada para as classificações em cada uma das disciplinas.

Tabela 2. Atributos de entrada para as tarefas de classificação

| Disciplina | Atributos de Entrada |
|-----------------------------|---|
| Introdução a Programação | Desempenho nas disciplinas (Matemática, Física, Química, Biologia, Geografia, História, Comunicação e Expressão e Língua Estrangeira) da prova de ingresso na instituição (decimal) |
| Linguagem de Programação I | Desempenho nas disciplinas da prova de ingresso na instituição (decimal), reprovou algum pré-requisito (booleano) e nota pré-requisito (decimal) |
| Linguagem de Programação II | Desempenho nas disciplinas da prova de ingresso na instituição (decimal), reprovou algum pré-requisito (booleano), nota pré-requisito I (decimal), nota pré-requisito II (decimal) |
| Estruturas de Dados | Desempenho nas disciplinas da prova de ingresso na instituição (decimal), reprovou algum pré-requisito (booleano), nota pré-requisito I (decimal), nota pré-requisito II (decimal) |

3.3 Avaliação do modelo

Para estudar a viabilidade do método proposto na previsão de desempenho dos alunos dos cursos de Computação foram realizados diversos testes na ferramenta Weka, que possui uma coleção de algoritmos de aprendizagem de máquina implementados e permite que os resultados obtidos com os vários métodos sejam facilmente explorados e comparados,

possibilitando que os algoritmos mais adequados sejam definidos para o problema [10].

Selecionaram-se quatro algoritmos de aprendizagem de máquina distintos: IBk, RandomForest (RF), BayesNET (BNet) e MultilayerPerceptron (MLP) e, para cada um deles, buscou-se, por tentativa e teste, encontrar o conjunto de parâmetros que produzisse os melhores resultados. Em ambos os casos a divisão entre conjunto de treinamento e testes foi feita a partir de Validação Cruzada (do Inglês *Cross Validation*) com 10 *folds*. Nesta abordagem o conjunto de dados original é dividido em *k* subgrupos disjuntos (folds), onde para *k-1* subgrupos é feito o treinamento do modelo, que é testado com o subconjunto restante. O processo é repetido usando-se subconjuntos diferentes para treinamento e teste, até completar *k* repetições. Ao término dos testes, calcula-se a média das taxas de acerto e erro de todos os testes. Com o intuito de se obter resultados mais consistentes nos testes, realizou-se uma etapa de pré-processamento no conjunto de dados a fim de balanceá-lo, dado que a quantidade de instâncias da classe “Sucesso” era bem maior que as da classe “Insucesso” (Ver Tabela 1) Aplicou-se o filtro SMOTE [5] para realizar o balanceamento.

Como forma de avaliação dos métodos utilizou-se a acurácia (número de instâncias classificadas corretamente) e as taxas de verdadeiros positivos (VP) e verdadeiros negativos (VN), já que a primeira métrica, quando usada isoladamente, pode não nos dar informações consistentes sobre o desempenho dos classificadores. Tanto os VP quanto os VN são classificações corretas e indicam as porcentagens de classificações corretas de estudantes que reprovaram nas disciplinas e a classificações corretas de estudantes que foram aprovados nas disciplinas, respectivamente. As taxas de falsos negativos (FN) e falsos positivos (FP) são complementares as medidas de VP e FP, respectivamente, e serão omitidas dos resultados por questão de simplicidade.

No intervalo de tempo dos dados obtidos para testes, a UFPB utilizou dois sistemas de ingresso diferentes (PSS e ENEM), sendo o segundo deles mais recente (datando do ano de 2011), assim tem-se um número reduzido de registros. Dessa maneira, os testes detalhados foram realizados com os registros do sistema de ingresso PSS e os registros da forma de ingresso ENEM foram utilizados para corroborar os resultados obtidos nos testes.

4. RESULTADOS E DISCUSSÃO

Nesta seção apresentamos os resultados dos testes de classificação dos estudantes na ferramenta Weka, para as disciplinas avaliadas. O primeiro teste de predição de sucesso foi realizado para a disciplina Introdução a Programação, ministrada no primeiro semestre dos cursos, utilizando os atributos de entrada descritos na Tabela 2, cujos resultados podem ser vistos na Tabela 3.

Tabela 3. Resultados obtidos na previsão de desempenho na disciplina “Introdução a Programação”

| Métrica/Algoritmo | IBk | RF | BNet | MLP |
|----------------------------|-------|-------|-------|-------|
| Acurácia | 81,9% | 82,9% | 74,2% | 73,1% |
| Verdadeiros Positivos (VP) | 93,6% | 81,8% | 70,1% | 72% |
| Verdadeiros Negativos (VN) | 73,2% | 83,8% | 77,2% | 74% |

A partir dos resultados mostrados na Tabela 3, vemos que os classificadores produziram bons resultados, sempre com acurácias e taxas de VP e VN superiores a 70%. Dentre os algoritmos testados, o RandomForest obteve acurácia próxima a 83%, com taxas de VP e VN a 81%, porém o algoritmo que obteve melhor taxa de classificação de VP foi o IBk, com resultados superiores a 93%. A classificação de VP positivos é importante, pois representa os alunos que provavelmente terão insucesso na disciplina, então a sua identificação é primordial para que ações sejam tomadas de forma a minimizar os índices de reprovação nas disciplinas.

O segundo experimento realizado consistiu na predição de desempenho de alunos na disciplina Linguagem de Programação I, que é ministrada no segundo semestre dos cursos. Os resultados dos testes realizados são mostrados na Tabela 4.

Tabela 4. Resultados obtidos na previsão de desempenho na disciplina “Linguagem de Programação I”

| Métrica/ Algoritmo | IBk | RF | BNet | MLP |
|----------------------------|-------|--------|-------|-------|
| Acurácia | 81,8% | 83,05% | 72,8% | 74,2% |
| Verdadeiros Positivos (VP) | 89% | 76% | 67,4% | 66,2% |
| Verdadeiros Negativos (VN) | 77,7% | 87,1% | 76% | 78,9% |

Verificamos na Tabela 4 que os resultados produzidos pelas classificações também foram bons, com acurácia de até 83% no melhor algoritmo testado. Nesta classificação, além das notas da prova de ingresso, consideram-se informações do desempenho do aluno na disciplina pré-requisito. As taxas de VP alcançam 89% quando utilizado o algoritmo IBk. No terceiro experimento, a predição de desempenho foi realizada para o conjunto de alunos na disciplina de Linguagem de Programação II, ministrada no terceiro período do curso. Os resultados desse experimento encontram-se na Tabela 5.

Tabela 5. Resultados obtidos na previsão de desempenho na disciplina “Linguagem de Programação II”

| Métrica/ Algoritmo | IBk | RF | BNet | MLP |
|----------------------------|-------|-------|-------|-------|
| Acurácia | 77,6% | 84,7% | 72% | 75,9% |
| Verdadeiros Positivos (VP) | 87,6% | 75,9% | 66% | 67,7% |
| Verdadeiros Negativos (VN) | 72,2% | 89,5% | 75,3% | 80,4% |

Pelos resultados da Tabela 5, podemos ver que os algoritmos também obtiveram bons resultados, com acurácia sempre acima de 71%, e taxas de VP variando entre 66% à 87,6% e VN variando entre 72,2% e 89,5%. Dos quatro algoritmos, podemos destacar, novamente, o IBk, que obteve 87,6% de taxa de VP, e como já explicado anteriormente, o conhecimento gerado pelos resultados do VP nos traz um maior conhecimento a fim de minimizar índices de reprovação e evasão. Outro ponto positivo

do algoritmo IBk, é que o mesmo conseguiu obter um bom resultado, mesmo com outros dois algoritmos (BayesNET e Multilayer Perceptron) obtendo resultados não tão bons, na casa dos 66% a 68%, mostrando que o IBk foi muito superior aos outros, para essa situação. Destaca-se também o algoritmo RandomForest que alcançou a maior acurácia (84,7%), apesar de menor taxa de VP.

O último experimento realizado levou em consideração o desempenho dos alunos na disciplina “Estrutura de Dados”. Na Tabela 6, verifica-se o desempenho dos algoritmos testados, que produziram acurácias sempre superiores a 73%, sendo o RandomForest o que obteve a melhor acurácia (81,25%) e o IBk aquele que obteve melhor taxa na classificação de VP.

Tabela 6. Resultados obtidos na previsão de desempenho na disciplina “Estruturas de Dados”

| Métrica/ Algoritmo | IBk | RF | BNet | MLP |
|----------------------------|-------|-------|-------|-------|
| Acurácia | 79,6% | 81,2% | 75% | 73,7% |
| Verdadeiros Positivos (VP) | 89,5% | 84,3% | 77% | 78,4% |
| Verdadeiros Negativos (VN) | 69,5% | 78,4% | 72,5% | 69,5% |

Através dos resultados apresentados, podemos concluir que o algoritmo IBk foi o que produziu os melhores resultados, destacando-se entre os demais. Possuindo resultados sempre superiores a 87,6% nas taxas de VP, entre os quatro testes realizados. Assim, apontando uma boa qualidade dos testes realizados e do modelo utilizado quanto a capacidade de prever o desempenho dos estudantes em disciplinas de programação.

Como contribuição, temos que, em posse do modelo de previsão apresentado, as tomadas de decisões tanto de professores como de coordenadores serão facilitadas, pois, com a análise realizada, os docentes terão uma previsão dos alunos propensos a falhar numa disciplina, de modo que possam tomar ações educativas prévias a fim de reverter o possível insucesso do mesmo na disciplina e, consequentemente, ajudar a diminuir os índices de reprovação e evasão em disciplinas de programação.

Como forma de validação dos resultados obtidos com o conjunto de dados de alunos provenientes da primeira forma de ingresso (PSS), realizaram-se experimentos similares com o conjunto de alunos provenientes do ENEM e os resultados obtidos ficaram bem próximos dos do primeiro teste. Isso sugere que o nosso método pode ser utilizado para várias formas de ingresso, não só as utilizadas neste artigo, bem como para diferentes tipos de cursos de graduação, desde que o mesmo apresente as disciplinas do exame de admissão referenciando disciplinas básicas do ensino médio.

5. CONCLUSÃO

Apresentamos neste trabalho um método para identificação de estudantes propensos a reprovação nas disciplinas de programação dos cursos de Computação. Utilizaram-se como forma de validação do método, registros de estudantes dos três cursos de graduação do Centro de Informática, da UFPB. Os resultados

obtidos com classificadores populares na ferramenta Weka atestaram a viabilidade das predições realizadas. Esperamos que os resultados obtidos neste trabalho possam ser usados no planejamento de estratégias para reduzir o problema da reprovação nas disciplinas de programação e, consequentemente, reduzir os índices de evasão na área que são influenciados pelas reprovações. Como trabalho futuro, pretendemos utilizar dados socioeconômicos dos alunos juntamente com as informações de desempenho do vestibular e disciplinas, na tentativa de criar um modelo mais robusto de classificação de estudantes.

6. REFERÊNCIAS

- [1] Baker, R. S. J., Isotani, S., & de Carvalho, A. M. J. B. (2011). Mineração de dados educacionais: Oportunidades para o Brasil. *Revista Brasileira de Informática na Educação*, 19(2).
- [2] Baradwaj, B. K. & Pal, S. (2012). Mining educational data to analyze students' performance.
- [3] Bhardwaj, B. K., & Pal, S. (2012). Data Mining: A prediction for performance improvement using classification.
- [4] Brito, D. M., de Almeida Júnior, I. A., Queiroga, E. V. & do Rêgo, T. G. (2014). Predição de desempenho de alunos do primeiro período baseado nas notas de ingresso utilizando métodos de aprendizagem de máquina. In *Anais do Simpósio Brasileiro de Informática na Educação* (Vol. 25, No. 1, pp. 882-890).
- [5] Chawla, N. V., Bowyer, K. W., Hall, L. O. & Kegelmeyer, W. P. (2002). SMOTE: synthetic minority over-sampling technique. *Journal of artificial intelligence research*, 321-357.
- [6] Costa, T. H. & Bublit, F. M. (2013). Análise dos Principais Problemas que Afetam Alunos de Programação: uma investigação empírica no Estado da Paraíba. Em *CSBC 2013 XXXIII Congresso da Sociedade Brasileira de Computação*.
- [7] Delgado, C., Xexeo, J. A., Souza, I. F., Rapkiewicz, C. E. & Pereira Jr, J. (2005). Identificando competências associadas ao aprendizado de leitura e construção de algoritmos. In *XIII Workshop sobre Educação em Computação*.
- [8] Fincher, S., Robins, A., Baker, B., Box, I., Cutts, Q., Raadt, M., Haden, P., Hamer, J., Hamilton, M., Lister, R., Petre, M., Sutton, K., Tolhurst, D. & Tutty, J. (2006). Predictors of success in a first programming course. In *Proceedings of the 8th Australasian Conference on Computing Education-Volume 52* (pp. 189-196). Australian Computer Society, Inc.
- [9] Garcia, R. E., Correia, R. C. M. & Shimabukuro, M. H. (2008). Ensino de Lógica de Programação e Estruturas de Dados para Alunos do Ensino Médio. In *XVII WEI-Workshop sobre o Ensino de Computação. Belém do Pará-PA*(pp. 246-249).
- [10] Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P. & Witten, I. H. (2009). The WEKA data mining software: an update. *ACM SIGKDD explorations newsletter*, 11(1), 10-18. REFERENCIA DO WEKA
- [11] Jenkins, T. (2002). On the difficulty of learning to program. In *Proceedings of the 3rd Annual Conference of the LTSN Centre for Information and Computer Sciences* (Vol. 4, pp. 53-58).
- [12] Jesus, A. & Brito, G. S. (2009). Concepção de ensino-aprendizagem de algoritmos e programação de computadores: A prática docente. *VARIA SCIENTIA*, 9(16).
- [13] Kovacic, Z. (2010). Early prediction of student success: Mining students' enrolment data.
- [14] Prietch, S. S. & Pazeto, T. A. (2010). Estudo sobre a Evasão em um Curso de Licenciatura em Informática e Considerações para Melhorias. *WEIBASE, Maceió/AL*.
- [15] Ramos, J. L. C., Rodrigues, R. L., Silva, J. C. S. & Gomes, A. S. (2014). Analisando Fatores que Afetam o Desempenho de Estudantes Iniciantes em um Curso a Distância. In *Anais do Simpósio Brasileiro de Informática na Educação* (Vol. 25, No. 1, pp. 99-108).
- [16] Rodrigues, R. L., de Medeiros, F. P. & Gomes, A. S. (2013). Modelo de Regressão Linear aplicado à previsão de desempenho de estudantes em ambiente de aprendizagem. In *Anais do Simpósio Brasileiro de Informática na Educação* (Vol. 24, No. 1).
- [17] Romero, C., & Ventura, S. (2010). Educational data mining: a review of the state of the art. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 40(6), 601-618.
- [18] Santos, R. P. & Costa, H. A. X. (2006). Análise de Metodologias e Ambientes de Ensino para Algoritmos, Estruturas de Dados e Programação aos iniciantes em Computação e Informática. *INFOCOMP Journal of Computer Science*, 5(1), 41-50.
- [19] Silva Filho, R. L. L., Motejunas, P. R., Hipólito, O. & Lobo, M. B. C. M. (2007). A evasão no ensino superior brasileiro. *Cadernos de Pesquisa*, 37(132), 641-659.
- [20] Wilson, B. C. & Shrock, S. (2001). Contributing to success in an introductory computer science course: a study of twelve factors. In *ACM SIGCSE Bulletin* (Vol. 33, No. 1, pp. 184-188). ACM.
- [21] Yadav, S. K., Bharadwaj, B. & Pal, S. (2012). Data mining applications: A comparative study for predicting student's performance.