Computers in Human Behavior 73 (2017) 247–256

Contents lists available at ScienceDirect

Computers in Human Behavior

journal homepage: www.elsevier.com/locate/comphumbeh

Full length article

Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses

Evandro B. Costa [a, *], Baldoino Fonseca [a], Marcelo Almeida Santana [a], Fabrísia Ferreira de Araújo [b, c], Joilson Rego [d]

[a] Federal University of Alagoas (UFAL), Brazil
[b] Federal Institute of Alagoas (IFAL), Brazil
[c] Federal University of Campina Grande, Brazil
[d] Federal University of Rio Grande do Norte (UFRN), Brazil

ARTICLE INFO

Article history:
Received 13 January 2016
Received in revised form
16 January 2017
Accepted 26 January 2017
Available online 4 February 2017

Keywords:
Artificial intelligence in education
Automatic instructional planner
Automatic prediction
Educational data mining
Interactive learning environment
Learner modeling

ABSTRACT

The data about high students' failure rates in introductory programming courses have been alarming many educators, raising a number of important questions regarding prediction aspects. In this paper, we present a comparative study on the effectiveness of educational data mining techniques to early predict students likely to fail in introductory programming courses. Although several works have analyzed these techniques to identify students' academic failures, our study differs from existing ones as follows: (i) we investigate the effectiveness of such techniques to identify students likely to fail at early enough stage for action to be taken to reduce the failure rate; (ii) we analyse the impact of data preprocessing and algorithms fine-tuning tasks, on the effectiveness of the mentioned techniques. In our study we evaluated the effectiveness of four prediction techniques on two different and independent data sources on introductory programming courses available from a Brazilian Public University: one comes from distance education and the other from on-campus. The results showed that the techniques analyzed in our study are able to early identify students likely to fail, the effectiveness of some of these techniques is improved after applying the data preprocessing and/or algorithms fine-tuning, and the support vector machine technique outperforms the other ones in a statistically significant way.

1. Introduction

The alarming indexes of students' academic failures, along the years, in universities' introductory programming courses (Bennedsen & Caspersen, 2007; Watson & Li, 2014) have been concerning educators. Studies (Hanks et al., 2004; Iepsen et al., 2013; Tan, Ting, & Ling, 2009) show that students face many difficulties during their programming activities in such a way that many of them end up failing or quitting the course at some initial stage.

In the above context, one relevant problem is on the ability to accurately predict the students likely to fail in introductory programming courses at early enough stage for possibiliting pedagogical interventions to be taken to avoid students' failures. In order to deal with this problem, some works (Arora, Singhal, & Bansal, 2014; Bayer et al., 2012; Manhães et al., 2014; Marquez-Vera, Morales, & Soto, 2013; Martinho, Nunes, & Minussi, 2013; Sim et al., 2006; Watson, Li, & Godwin, 2013) have proposed and analyzed the use of Educational Data Mining (EDM) techniques to predict students' academic failures. However, in general, these works are not concerned with two important questions: (i) how effective are the EDM techniques to early identify students likely to fail?; and (ii) Do the data preprocessing (Hu, 2003; Crone et al., 2006; Zaki & Jr.W. M., 2014) and algorithms fine-tuning (Gunawan et al., 2011; Hutter, Hoos, Leyton-Brown, & Stützle, 2009) impact the effectiveness of EDM techniques?

In order to answer the above mentioned questions, we present a comparative study on the effectiveness of EDM techniques to early predict students likely to fail in introductory programming courses. Given the amount of existing EDM techniques available (Caruana &

* Corresponding author.
E-mail addresses: evandro@ic.ufal.br (E.B. Costa), baldoino@ic.ufal.br (B. Fonseca), marceloalmeidasantana@gmail.com (M.A. Santana), fabrisia.araujo@gmail.com (F.F. de Araújo), jotarego@gmail.com (J. Rego).

http://dx.doi.org/10.1016/j.chb.2017.01.047
0747-5632/

Niculescu-Mizil, 2006), we used the following classifiers: Neural Networks (Nürnberger et al., 2002; Rumelhart, Hinton, & Williams, 1988), Decision Tree (Breiman et al., 1984; Salzberg, 1994), Support Vector Machine (SVM) (Cortes & Vapnik, 1995; Vapnik, 1995) and Naive Bayes (Domingos & Pazzani, 1997). These techniques have been widely investigated by existing EDM works (Wu et al., 2008) and they have presented interesting results.

In our study we used the f-measure (Han et al., 2011) to evaluate the effectiveness of the selected techniques on two different and independent data sources concerning two introductory programming courses available from a Brazilian Public University: one comes from distance education and the other from on-campus. The experiment was performed by considering the preprocessing of these data sources and the fine-tuning of the analyzed techniques.

The results showed that the techniques analyzed in our study are able to early identify students likely to fail, and demonstrated that the data preprocessing and algorithms fine-tuning tasks influence the effectiveness of these techniques. The SVM technique outperformed the other ones by predicting with 92% and 83% of effectiveness the failures of students that have performed at least 50% of the courses by distance education or on-campus, respectively.

This paper is organized as follows. Section 2 we present the method applied in our experiment. In Section 3 we present the results and discussions of the experiment. In Section 4 we discuss some similar work. Conclusions and future work are presented in Section 5.

2. Method

The general goal of this study is to compare the effectiveness of existing EDM techniques for early identification of students likely to fail with high precision. This section is organized as follows. Section 2.1 poses four research questions that drive our assessment. Section 2.2 presents the data sources and the EDM techniques we have analyzed in our experiment. Sections 2.3 and 2.4 indicate the tools and metrics, respectively, we have used when conducting the experiment, and, finally, Section 2.5 presents some details about the steps and configurations used to perform the experiment.

2.1. Planning

Our comparative study is guided by the following research questions:

Question 1. How effective are the EDM techniques to early identify students likely to fail?

Our aim with the Question 1 is to evaluate the effectiveness of the EDM techniques that have been used by existing approaches to early identify students likely to fail. To answer Question 1, we performed these techniques on two different data sources and then we used the F-measure to evaluate the effectiveness of such techniques.

Question 2. Is the data preprocessing able to increase the effectiveness of the EDM techniques?

The **Question 2** aims to analyse if the effectiveness of EDM techniques increases after performing the data preprocessing. In order to answer **Question 2**, we performed a preprocessing of the two data sources used in this experiment, then we applied the EDM techniques on these data sources. Subsequently, we evaluated the effectiveness of these techniques and we compared such results with the effectiveness obtained by performing the same techniques on the data without the preprocessing.

Question 3. Is the fine-tuning of algorithms able to further increase the effectiveness of the EDM techniques?

The **Question 3** aims to analyse if the effectiveness of EDM techniques further increases after performing the fine-tuning of their parameters. In order to answer **Question 3**, we performed a fine-tuning of the EDM techniques, then we performed the fine-tuned techniques on the preprocessed data source, as well as we evaluated the effectiveness of techniques and we compared their effective-ness with the results obtained by performing the EDM techniques without the fine-tuning.

Question 4. After performing the data preprocessing and fine-tuning of algorithms, which of the EDM techniques are more effective for early identification of students likely to fail?

The **Question 4** aims to find the most effective techniques for early identification of students likely to fail. In order to answer **Question 4**, we analyzed the effectiveness of the EDM techniques after performing the fine-tuning of their parameters and the preprocessing of the data sources.

2.2. Data sources and EDM techniques selection

In this experiment we have analyzed two data sources extracted from introductory programming courses performed either on-campus or distance education. In what follows a brief description of these two data sources:

(Distance Education) The first data source contains information about 262 undergraduate students that took the introductory programming course performed in a distance education modality in our university in 2013 during 10 weeks. In this course the students were weekly evaluated according to their activities plus two exams that were applied in the fifth and last week of the course. These activities and exams were applied through an online system used in our university.

This data source contains the following information about the students: age, gender, civil status, city, income, student registration, period, class, semester, campus, access frequency of the students in the system, participation in the discussions forum, amount of received and viewed files, use of the educational tools provided by the system as blog, glossary, quiz, wiki, message, year of enrolling in the course, status on discipline, and performance of the students in the weekly activities and exams.

(On-campus) The second data source contains information about161students that took the introductory programming course performed on-campus in our university in 2014, during 16 weeks. In this course the students were weekly evaluated according to their activities plus four exams that were applied in the fourth, eighth, twelfth and sixteenth week of the course.

The data source contains the following students information: age, gender, civil status, city, income, student registration, period, class, semester, campus, year of enrolling in the course, status on discipline, amount of exercise performed by the student, number of correct exercises, and performance of the students in the weekly activities and exams.

Our main goal is to evaluate the effectiveness of the EDM techniques to predict students likely to fail at early enough stage for supporting future pedagogical interventions to be taken to avoid

students' failures. Thus, in this experiment we used the students information only until the application of the first exams to analyse the effectiveness of four EDM techniques: Naive Bayes classifier, which is based on Bayes' theorem (Domingos & Pazzani, 1997), decision tree (Breiman et al., 1984; Salzberg, 1994) (in this case, we used the J48 algorithm (Witten et al., 2011) to implement the Decision Tree technique), multilayer neural network (Nürnberger et al., 2002; Rumelhart et al., 1988) and support vector machine (Cortes & Vapnik, 1995; Vapnik, 1995).

These techniques were selected since they have presented good effectiveness in different domains (Caruana & Niculescu-Mizil, 2006). In particular, they have been used in existing approaches (Wu et al., 2008) for the identification of students likely to fail.

2.3. Instrumentation

We used the Pentaho Data Integration tool (Pentaho, 2015) to perform all the preprocessing of the data sources. Pentaho is an open-source software, developed in Java, which is able to: (i) extract information from data sources; (ii) select attributes; (iii) discretize data and (iv) generate files compatible with the format used by data mining tools. In addition, we used the WEKA tool (Weka, 2015) to apply the EDM techniques analyzed in this experiment.

2.4. Effectiveness metrics

To characterize the effectiveness of the EDM techniques analyzed in this experiment, we decided to adopt the F-Measure (Han et al., 2011), which is widely used in domains such as information retrieval, machine learning and other domains that involve binary classification (Olson & Delen, 2008). In short, F-Measure (Eq. (1)) is the harmonic mean between Precision (Eq. (2)) and Recall (Eq. (3)), as described below:

$$Fmeasure = 2x\frac{Precision\ x\ Recall}{Precision + Recall} \quad (1)$$

$$Precision = \frac{TP}{FP + TP} \quad (2)$$

$$Recall = \frac{TP}{FN + TP} \quad (3)$$

where:

(True Positives - TP) TP is the number of positive instances correctly classified as positive.
(False Positives - FP) FP is the number of negative instances incorrectly classified as positive.
(False Negatives - FN) FN is the number of positive instances incorrectly classified as negative.

2.5. Operation

In this Section we describe in more details the preprocessing performed on the two data sources and the fine-tuning of the EDM techniques used in our experiment. Section 2.5.1 describes the preprocessing and Section 2.5.2 describes the fine-tuning of algorithms.

2.5.1. Data preprocessing

Before applying the EDM techniques, we performed the preprocessing of each data source, separately, in order to deal with two important problems that may exist frequently in educational data (Marquez-Vera et al., 2013): (i) high dimensionality, that is, a large number of attributes. A large number of attributes may hinder prediction algorithms to reach interesting results in a short time; and (ii) unbalanced data. When the number of instances from one class is much larger than the number of instances from other classes (Gu et al., 2008), prediction algorithms tend to focus on learning from classes with larger number of instances.

The two data sources analyzed in this experiment suffer with the high dimensionality problem since they contain many attributes to be handled, as described in Section 2.2. In order to deal with the high dimensionality problem, we evaluated experimentally the attributes selection algorithms provided by WEKA on each data source analyzed in this experiment, and then we selected the

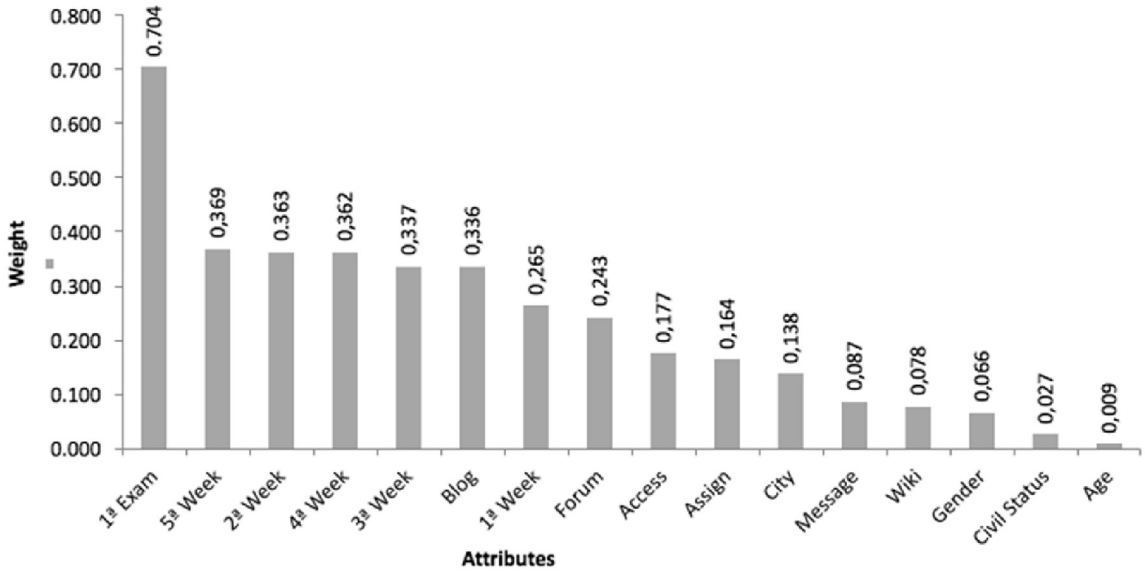

Fig. 1. Attributes weights of the distance education data source.

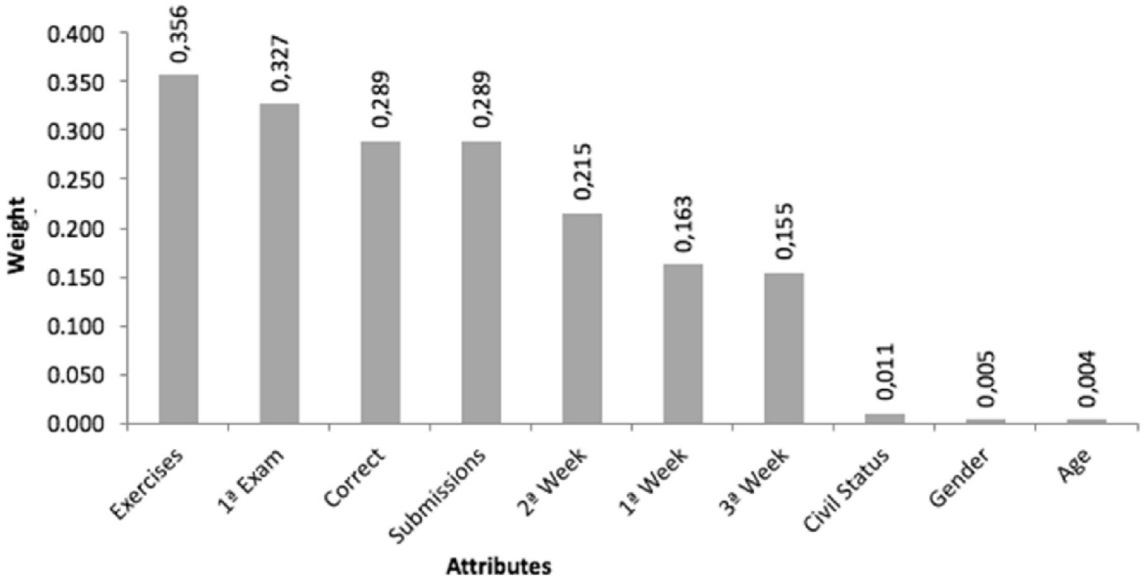

Fig. 2. Attributes weights of the on-campus data source.

Information Gain algorithm (InfoGainAttributeEval) (Quinlan, 1986) since it presented the best results in both data sources. Figs. 1 and 2 show the attributes' weights defined by the InfoGainAttributeEval algorithm after applying it on the distance education and on-campus data sources.

In addition, according to the students' information stored in the data sources, the number of students that have failed (or succeeded) are not balanced. In order to deal with such problem, we evaluated experimentally the balancing algorithms provided by WEKA and then we selected the SMOTE algorithm (Synthetic Minority Oversampling Technique) (Chawla et al., 2002) to be applied on each data source.

2.5.2. Fine-tuning of the EDM techniques

After preprocessing the data sources, we performed the fine-tuning of the four EDM techniques as follows:

(Support vector Machine) Studies (Viana et al., 2007) show that the SVM algorithm is very sensitive to fine-tuning, mainly in real word problems. As the manual fine-tuning is undesirable because it is imprecise and it does not guarantee the quality of the results (Imbault & Lebart, 2004), we used the "Grid-Search" method (Han et al., 2011) to perform the fine-tuning of the SVM.
(Decision Tree via J48) According to (Witten et al., 2011), the effectiveness of the J48 decision tree algorithm can be improved by performing a fine-tuning of two parameters: (i) the amount of leaf nodes and (ii) the decision-tree pruning. Thus, we performed some comparative experiments in order to perform the fine-tuning of these two parameters.
(Neural Network) We performed the fine-tuning of three parameters of the Neural Network algorithm: (i) the learning rate of the weights; (ii) the momentum applied to the weights during their updating; (iii) the number of hidden layers existing in the network. According to (Witten et al., 2011), the fine-tuning of these parameters can improve the effectiveness of neural network algorithm.
(Naive Bayes) The fine-tuning of the Naive Bayes algorithm was performed by following the approach described in (John & Langley, 1995), which uses a method based on the kernel estimation to perform the fine-tuning of parameters of Naive Bayes algorithm.

3. Results and discussions

In this section, we present the main results of the experiment outlined in Section 2. In Section 3.1 we answer the research questions listed in Section 2.1. Section 3.2 discusses some threats to validate our experiment.

3.1. Research questions

Next we answer and discuss the following research questions.

3.1.1. How effective are the prediction algorithms to early identify students likely to fail?

In order to answer this question, we performed the four EDM techniques, analyzed in this experiment, on the data sources: distance education and on-campus. Figs. 3 and 4 present the effectiveness (represented by F-measure) of the EDM techniques to identify students likely to fail. In this case, we considered the students' information by the application of first exams of the distance education and on-campus courses, respectively.

We observe that the techniques present an effectiveness that varies from 0.55 to 0.82 in the distance education course, and from 0.50 to 0.79 in the on-campus course. These results indicate that after the first week of the courses the EDM techniques are able to identify with at least 50% of effectiveness the students likely to fail.

We also note that Decision Tree techniques present the highest effectiveness on both data sources. It reached a F-measure value equal to 0.82 after applying the first exam of the distance education course, and 0.79 in the second week of the on-campus course.

Given that the distance education and on-campus courses have duration of 10 and 16 weeks, respectively, we can state that the Decision Tree technique is able to reach an effectiveness of 82% when the students have performed at least 50% of the distance education course, and an effectiveness of 79% when the students have performed at least 25% of the on-campus course.

The results present evidences that the EDM techniques analyzed in these experiments are effective to early identify students likely

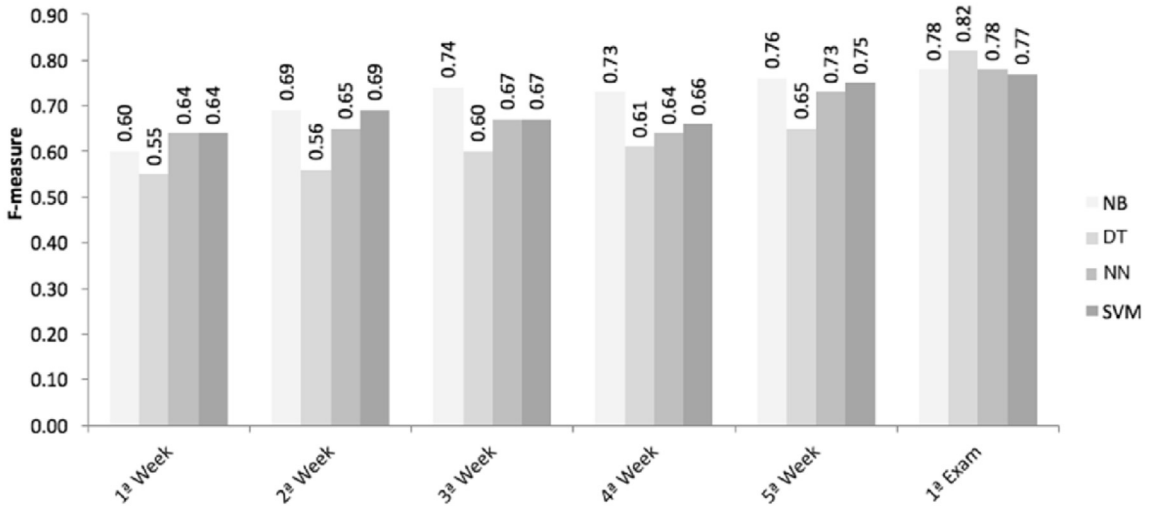

Fig. 3. Effectiveness of the EDM Methods on the Data Source Online (NB - Naive Bayes; DT - Decision Tree; NN - Neural Network; SVM - Support vector Machine).

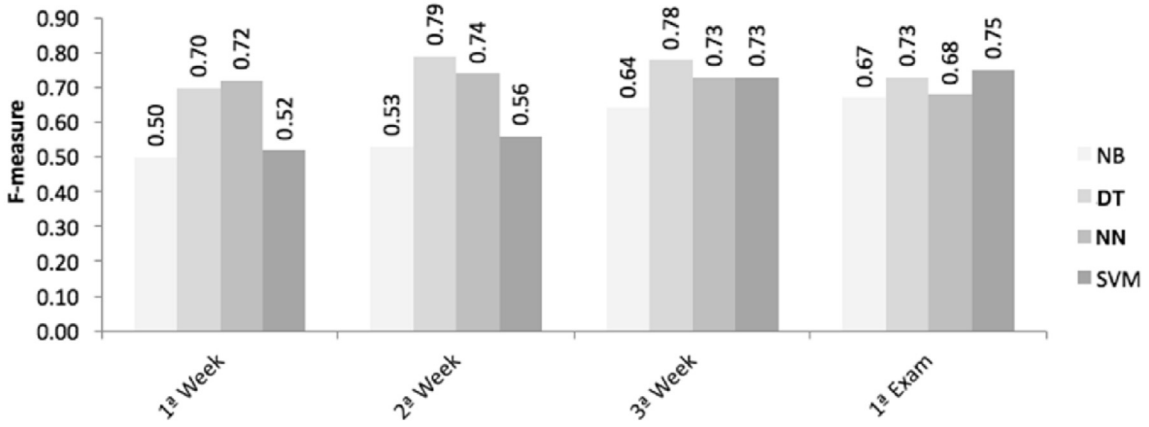

Fig. 4. Effectiveness of the EDM Methods on the Data Source On-campus (NB - Naive Bayes; DT - Decision Tree; NN - Neural Network; SVM - Support vector Machine).

to fail. In the next section, we will show that the effectiveness of these techniques can be improved by performing the two additional steps: data preprocessing and fine-tuning of algorithms.

3.1.2. Is the data preprocessing able to increase the effectiveness of the EDM techniques?

In order to answer this question we performed a preprocessing of the two data sources used in our study, as described in Section 2.5.1. Then, we applied the four EDM techniques on the preprocessed data sources and we evaluated the effectiveness of these techniques on the preprocessed data sources. Finally, we compared such results with the ones obtained by the techniques when we applied them on the data sources without preprocessing.

Fig. 5 presents the comparative results of the effectiveness of the four techniques (Naive Bayes, Decision Tree, Neural Network and Support Vector Machine) when we applied them on the distance education data and then on the preprocessed distance education data. The results shown in Fig. 5 indicate that the effectiveness of all techniques was improved when we applied them on the preprocessed distance education data.

Given the mentioned scenario, it is necessary to use a statistical test to verify whether such improving is statistically significant. By applying the *t*-test (Han et al., 2011) on the results as shown in Fig. 5, we obtained the following p-values: (Naive Bayes) p-value = 0.1158; (Decision Tree) p-value = 0.006349; (Neural Network) p-value = 0.002343; and (Support vector Machine) p-value = 0.0005339. According to (Han et al., 2011), in order to represent a significant difference, normally, the p-value should be lower than 0.05. Thus, we conclude that the Naive Bayes is the only technique that does not present a statistically significant increase when we applied the techniques on the preprocessed distance education data.

Fig. 6 presents the comparative results of the effectiveness of the four EDM techniques when we applied them on the on-campus data (see results shown in Fig. 4) and then on the preprocessed on-campus data. The results shown in Fig. 6 do not make clear whether the effectiveness of the algorithms was improved.

By applying the *t*-test on the results shown in Fig. 6 we obtained the following p-values: (Naive Bayes) p-value = 0.7793; (Decision Tree) p-value = 1; (Neural Network) p-value = 0.468; and (Support vector Machine) p-value = 0.8422. We note that the algorithms do not present a significant improvement when we applied them on the preprocessed on-campus data.

Based on this discussion, we can conclude that the preprocessing on the distance education data was able to increase the effectiveness of most of the techniques, but the preprocessing on the on-campus data did not significantly impact the effectiveness of the techniques.

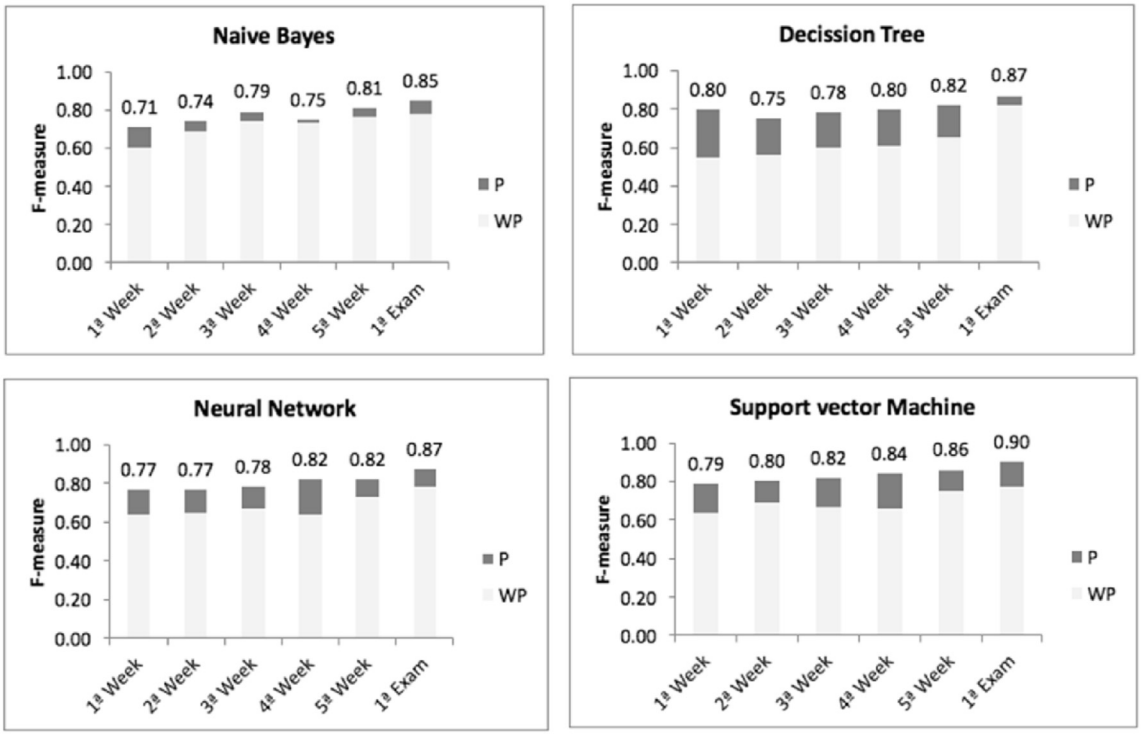

Fig. 5. Comparative Results of the Effectiveness of the EDM Methods on the Data On-line without Preprocessing (WP - Without Preprocessing) and then with Preprocessing (P - Preprocessed).

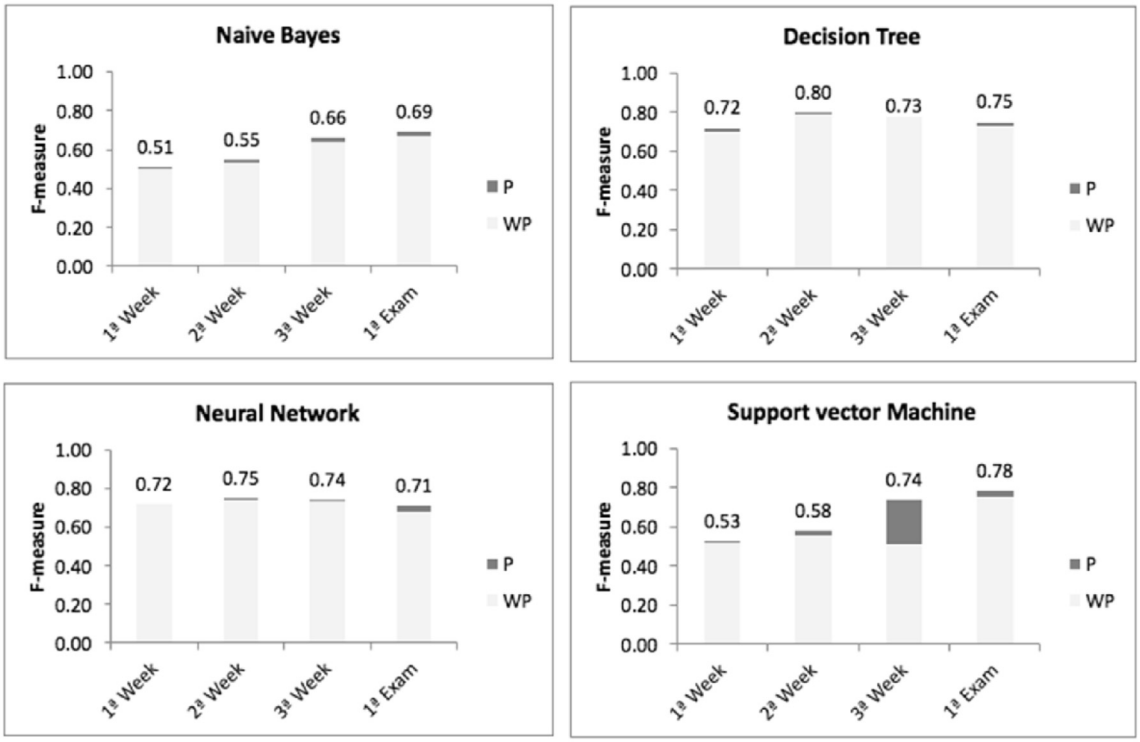

Fig. 6. Comparative Results of the Effectiveness of the EDM Methods on the Data On-campus without Preprocessing (WP - Without Preprocessing) and then with Preprocessing (P - Preprocessed).

3.1.3. Is the fine-tuning of algorithms able to further increase the effectiveness of the EDM techniques?

Studies (Gunawan et al., 2011; Hutter et al., 2009) indicate that the effectiveness of some EDM techniques can be further improved by fine-tuning them, specially in real-world problems. In order to investigate such evidences we performed the fine-tuning of the

EDM techniques, as described in Section 2.5.2, then we used the preprocessed data sources to compare the effectiveness of the techniques without the fine-tuning and then by performing the fine-tuning of them.

Fig. 7 presents the comparative results of the EDM techniques effectiveness when we applied them on the preprocessed distance education data without performing their fine-tuning and then by performing their fine-tuning. The results shown in Fig. 7 indicate that the effectiveness of the techniques was improved after performing the fine-tuning of the techniques.

By applying the *t*-test on the results shown in Fig. 7 we obtained the following p-values: (Naive Bayes) p-value = 0.01182; (Decision Tree) p-value = 0.005095; (Neural Network) p-value = 0.001285; and (Support vector Machine) p-value = 0.0003295. Thus, we conclude that the fine-tuning improved the effectiveness of all EDM techniques.

Fig. 8 presents the comparative results of the effectiveness of the EDM techniques when we applied them on the on-campus data. The results shown in Fig. 8 do not make clear whether the effectiveness of all techniques was improved by fine-tuning them.

By applying the *t*-test on the results shown in Fig. 8 we obtained the following p-values: (Naive Bayes) p-value = 0.02317; (Decision Tree) p-value = 0.2945; (Neural Network) p-value = 0.6532; and (Support Vector Machine) p-value = 0.03911. According to these p-values, only the effectiveness of the Naive Bayes and Support Vector Machines algorithms were improved.

Based on this discussion, we can conclude that the fine-tuning of the EDM techniques was able to increase their effectiveness when we applied them on the preprocessed distance education data, but only two fine-tuned techniques (Naive Bayes and Support Vector Machine) increased the effectiveness when we applied them on the preprocessed on-campus data.

3.1.4. After performing the data preprocessing and algorithms fine-tuning, which of the EDM techniques are more effective for early identification of students likely to fail?

According to the results shown in the previous Section, after preprocessing the data sources and performing the fine-tuning of the techniques, the fine-tuned Support Vector Machine algorithm presented the best effectiveness on both data sources by reaching a F-measure value equal to 0.92 and 0.83 after the first exams application of the distance education and on-campus courses, respectively. In other words, the fine-tuned Support vector Machine techniques is able to identify with at least 92% and 83% of effectiveness the students likely to fail when they have performed at least 50% and 25% of the distance education and on-campus courses, respectively.

3.2. Threats to validity

Although our experiment provided interesting evidences about the effectiveness of existing EDM techniques to early identify students likely to fail, it is important to note some threats.

First, the data sources used in our experiment represent information about students of two courses (distance education and on-campus) from only one university. Therefore, the experiment results are not general.

Second, we adopted the f-measure to characterize the effectiveness of the EDM techniques. Although this measure has been widely used by existing EDM works, other measures, such as, accuracy and Kappa, could be used. Finally, only the fine-tuning of the SVM and Naive Bayes techniques were performed in an automatic way. This is an important threat because the manual fine-tuning of the techniques can impact their effectiveness.

4. Related work

Several studies have been reported in the literature to predict students' academic failures by using EDM techniques. Although such studies have presented promising ways to identify whether a given student will fail in a given course, some of them are somewhat limited in terms of predicting failure accurately and early enough to allow for timely intervention delivery. Moreover, part of

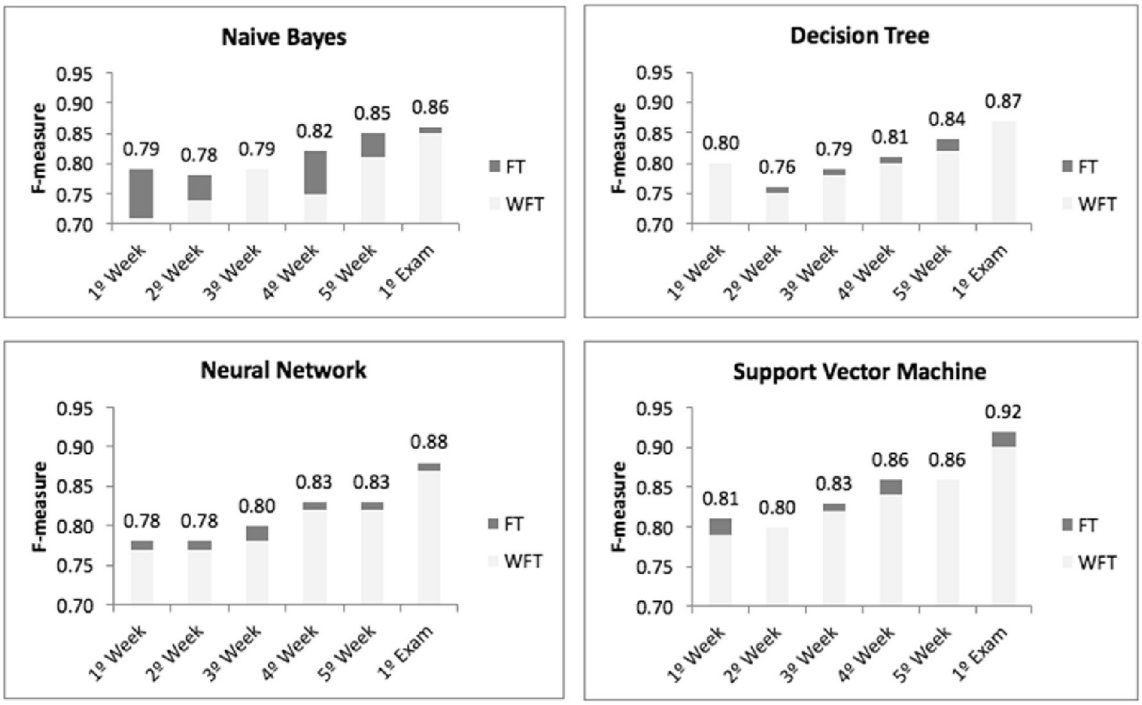

Fig. 7. Comparative Results of the Effectiveness of the EDM Methods without Fine-Tuning (WFT - Without Fine-tuning) and then with Fine-tuning (FT - Fine-tuned).

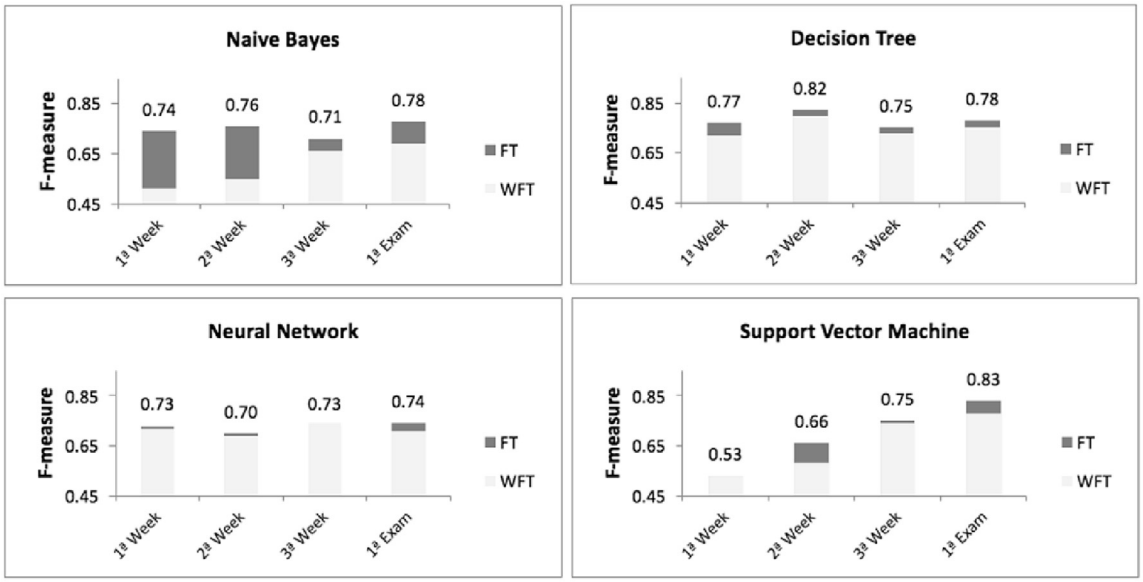

Fig. 8. Comparative Results of the Effectiveness of the EDM Methods without Fine-Tuning (WFT - Without Fine-tuning) and then with Fine-tuning (FT - Fine-tuned).

those works requires many non-academic data to be used by the prediction algorithm, normally, using time-consuming questionnaires. In addition, most of such studies do not properly investigate the influence of data preprocessing and algorithms fine-tuning on the effectiveness of the analyzed EDM techniques.

The approach proposed in (Marquez-Vera et al., 2013), the authors evaluate the use of EDM techniques: induction rules and decision trees, to predict students' academic failures in middle or secondary education. The results show evidences that such techniques are a promising way to perform such kind of predictions with relevant accuracy. However, to achieve those results, this work had to consider many different variables from several data sources, including non-academic data, such as personal and family, from time-consuming survey, as well as students' data from grades obtained in several courses. Moreover, they do not predict whether a student will fail at early enough phase.

The work in (Khobragade & Mahadik, 2015) follows a similar approach as the one previously discussed work, to predict the students' failure, this paper invested in some White-Box classifiers algorithms to induce rules and decision tree, involving the use of two algorithms for rules and two for decision tree. Moreover, it also used Naive Bayes algorithm. It selected 11 best attributes and most of the features selected were based on real-world data of student information (e.g. student marks, family background, social and academic related features) and also their past performance, as the past performance of a student is indicative of his present/future performance. In most of the cases, these data can be collected by using college reports and surveys. Naive Bayes algorithm provided the best accuracy with 87.12. Therefore, this work obtained result for prediction was high, but it does not invested in early prediction and it used many other data source, besides academic data.

The work proposed in (Ahmad, Ismail, & Aziz, 2015), presents an approach for predicting students academic performance of first year bachelor students in Computer Science course. It uses a framework, which supports Decision Tree, Naive Bayes, and Rule Based classification techniques, to be applied to the students' data to produce the best students' academic performance prediction model. The used data were collected during 8 semesters, containing the students' demographics data, previous academic records, and family background information. The results show that the Rule Based classifier is a best model among the other techniques by receiving the highest accuracy value of 71.3.

The work in (Yukselturk et al., 2014), examined, based on the surveyed data, the prediction of students dropouts from a course through the EDM classifiers: Decision tree, Naive Bayes, Neural network, and K-Nearest Neighbor (k-NN). The data was collected through online questionnaires (Demographic Survey, Online Technologies Self-Efficacy Scale, Readiness for Online Learning Questionnaire, Locus of Control Scale, and Prior Knowledge Questionnaire). The collected data included 10 variables, which were gender, age, educational level, previous online experience, occupation, self efficacy, readiness, prior knowledge, locus of control, and the dropout status as the class label (dropout/not). The precisions obtained were k-NN (87%), Decision Tree (79.7%), Naive Bayes (76.8%) and Neural Network (73.9%). As stated, it just used data from surveys. In addition, it does not predict early the student performance.

The work in (Bydzovska, 2016) addressed the problem of predicting final grades of students at the beginning of the semester with the emphasis on identifying unsuccessful students. To do this, it used two different approaches, where the first was based on classification and regression algorithms. This approach was considered interesting when used for the grade prediction of courses with a small number of students. The employed algorithms were: Support Vector Machine, Random Forest, Rule-based classifier, Decision Tree, Part, IB1, and Naive Bayes. In this study, SVM reached the best performance. The results were improved by also using data about social behavior of students in the predictions. The second used approach was in a different line by considering collaborative filtering techniques and predicted grades, based on the similarity of students' achievements. This way is very different of the one discussed in our approach. It included data about social behavior of students, being in this way different from our approach, but the best performance with SVM was similar to the one obtained in our approach.

The work in (Bayer et al., 2012) uses a method to classify students at risk of failures throughout the course. It uses personal data of students enriched with data related to social behaviors. It uses data preprocessing techniques and evaluates the effectiveness of seven EDM algorithms in order find the best one. The results show

that the analyzed algorithms are able to reach effectiveness up to 93.51%. However, this effectiveness is only reached at the end of the course, therefore it does not predict early, making difficult to apply pedagogical interventions in order to avoid students' failures. In addition, it uses non-academic data, such as personal and social behavior, from time-consuming survey.

In (Watson et al., 2013), the authors present an approach, called Watwin, based on a dynamic algorithm designed to predict student performance in a programming course. In a nutshell, the approach works as follows: when a student compiles its program on a university PC, the approach takes a snapshot of the program source code and it collects information about the success or fail of the program, execution time, error messages and code line number. Such information is used by the Watwin approach to predict the failure of students. The results indicate that the approach is able to reach effectiveness up to 75%. This performance is not so effective, but it just used data generated from the system. Moreover, it does not predict early.

In (Er, 2012) is proposed an approach for predicting students' performance based on three EDM techniques: instance-based learning Classifier, Decision Tree and Naive Bayes. The experiment was performed in a distance education course and it was performed in three steps, which correspond to different stages in a semester. At each step, new instances were added to the data sources until the course achieved the final stage. The results suggest that the model was able to reach effectiveness up to 85%. However, it does not explicit how to predict early.

The work in (Manhães et al., 2014) presents an approach that uses EDM techniques to predict students likely to fail in an academic course. To do this, it uses data from three undergraduate engineering courses from a Brazilian public university. According to the experiments, the Naive Bayes technique presented the best effectiveness for all data sources analyzed in the experiment. It is not clear in this paper whether the used approach predict in an early stage.

The study of (Martinho et al., 2013) investigates the use of the Neural Network technique to early identify groups of students likely to fail. The experiment results indicate that the proposed approach is able to reach effectiveness equal to 76%. This approach is similar to ours in the sense that it pursued the aim to early identify groups of students likely to fail. However the obtained performance to do that was not so good.

As discussed before, our approach to determine students that might be at risk of failing, firstly takes into account the need of early prediction because the information result of this could potentially provide early educational intervention for students. Additionally, we pursued the requirements of obtaining results with high precision, as well as to avoid explore non-academic data, just using data from the academic systems. To achieve this aim, we invested in EDM techniques by considering the influence of the data preprocessing and algorithms fine-tuning techniques on the predictions. In this aim, the works (Ahmad et al., 2015; Bayer et al., 2012; Bydzovska, 2016; Khobragade & Mahadik, 2015; Marquez-Vera et al., 2013) are, mainly in some technical data mining way, very similar to our approach, in the sense that they invest in a comparative study by exploring, besides the data mining technique in itself, other involved aspects, such as data preprocessing mechanisms. Moreover, all of them obtained high precision in their results. However, the majority of them do not invest in early prediction, as well as, some them use data from survey or questionnaire.

5. Conclusions and future work

We have presented the results of an investigation on the effectiveness of EDM techniques for early identification of students likely to fail in introductory programming courses.

Our investigation differs from related works and in this reside our contributions, in that: (i) we investigate the effectiveness of EDM techniques to identify students likely to fail at early enough stage for action to be taken to reduce the failure rate; and, (ii) we analyse the impact of data preprocessing and algorithms fine-tuning tasks on the effectiveness of these techniques.

Specifically, the study was conducted by performing a comparative study on the effectiveness of four EDM techniques (Decision Tree, Support Vector Machine, Neural Network and Naive Bayes). These techniques were evaluated on two different and independent data sources on introductory programming courses available from a Brazilian university: one comes from distance education and the other from on-campus. In addition, we performed data preprocessing and fine-tuning tasks during the realization of the experiment.

The study results allow us to draw one important conclusion, indicating that the analyzed EDM techniques are sufficiently effective to early identify students' academic failures, and then they are useful to provide educators or teachers with relevant information to help your decisions.

We also investigated the effectiveness of the EDM techniques in other data sources related to advanced programming courses and the techniques present similar results. As future work, this study can be improved by considering other data sources from different universities as well as the use of other techniques of data preprocessing and algorithms fine-tuning.

References

Ahmad, F., Ismail, N. H., & Aziz, A. A. (2015). The prediction students' academic performance using classification data mining techniques. *Applied Mathematical Sciences, 9*, 6415–6426. http://dx.doi.org/10.12988/ams.2015.53289.

Arora, Y., Singhal, A., & Bansal, A. (2014). A method to improve student's performance. *SIGSOFT Software Engineering Notes, 39*, 1–5. http://dx.doi.org/10.1145/2557833.2557842.

Bayer, J., Bydzovska, H., Geryk, J., Obsivac, T., & Popelinsky, L. (2012). Predicting drop-out from social behaviour of students. In *Proceedings of the 5th international conference on educational data mining - EDM 2012* (pp. 103–109) (Greece).

Bennedsen, J., & Caspersen, M. E. (2007). *Failure rates in introductory pro- gramming, 39* pp. 32–36). SIGCSE Bull. http://dx.doi.org/10.1145/1272848.1272879.

Breiman, L., Friedman, J., Olshen, R., & Stone, C. (1984). *Classification and regression trees*. Monterey, CA: Wadsworth and Brooks.

Bydzovska, H. (2016). *A comparative analysis of techniques for predicting stu- dent performance* (pp. 306–311). International Educational Data Mining Society.

Caruana, R., & Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on machine learning ICML '06* (pp. 161–168). New York, NY, USA: ACM. http://dx.doi.org/10.1145/1143844.1143865.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). Smote: Synthetic minority over-sampling technique. *J. Artif. Int. Res., 16*, 321–357. http://dl.acm.org/citation.cfm?id=1622407.1622416.

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Machine Learning, 20*, 273–297. http://dx.doi.org/10.1023/A:1022627411411.

Crone, S. F., Lessmann, S., & Stahlbock, R. (2006). The impact of preprocessing on data mining: An evaluation of classifier sensitivity in direct marketing. *European Journal of Operational Research, 173*, 781–800. http://dx.doi.org/10.1016/j.ejor.2005.07.023. http://www.sciencedirect.com/science/article/pii/S0377221705006739.

Domingos, P., & Pazzani, M. (1997). On the optimality of the simple bayesian classifier under zero-one loss. *Machine Learning, 29*, 103–130. http://dx.doi.org/10.1023/A:1007413511361.

Er, E. (2012). Identifying at-risk students using machine learning techniques: A case study with is 100. In *International journal of machine learning and computing* (pp. 476–481). Singapore: IACSIT Press. http://dx.doi.org/10.1145/2554850.2555135.

Gu, Q., Cai, Z., Zhu, L., & Huang, B. (2008). Data mining on imbalanced data sets. In *Advanced computer theory and engineering, 2008. ICACTE '08. International conference on* (pp. 1020–1024). http://dx.doi.org/10.1109/ICACTE.2008.26.

Gunawan, A., Lau, H., & Lindawati. (2011). Fine-tuning algorithm parameters using the design of experiments approach. In C. Coello (Ed.), *Learning and intelligent optimization* (pp. 278–292). Springer Berlin Heidelberg volume 6683 of Lecture Notes in Computer Science. http://dx.doi.org/10.1007/978-3-642-25566-3_21.

Han, J., Kamber, M., & Pei, J. (2011). *Data Mining: Concepts and techniques* (3rd ed.).

San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
Hanks, B., McDowell, C., Draper, D., & Krnjajic, M. (2004). *Program quality with pair programming in cs1, 36* pp. 176–180). SIGCSE Bull. http://dx.doi.org/10.1145/1026487.1008043.
Hu, X. (2003). Db-hreduction: A data preprocessing algorithm for data mining applications. *Applied Mathematics Letters, 16*, 889–895. http://dx.doi.org/10.1016/S0893-9659(03)90013-9. http://www.sciencedirect.com/science/article/pii/S0893965903900139.
Hutter, F., Hoos, H. H., Leyton-Brown, K., & Stützle, T. (2009). Paramils: An automatic algorithm configuration framework. *Journal of Artificial Intelligence Research, 36*, 267–306. http://dl.acm.org/citation.cfm?id=1734953.1734959.
Iepsen, E., Bercht, M., & Reategui, E. (2013). Detection and assistance to students who show frustration in learning of algorithms. In *Frontiers in Ed- ucation conference, 2013 IEEE* (pp. 1183–1189). http://dx.doi.org/10.1109/FIE. 2013.6685017.
Imbault, F., & Lebart, K. (2004). A stochastic optimization approach for pa- rameter tuning of support vector machines. In *In pattern recognition, 2004. ICPR 2004. Proceedings of the 17th international conference on* (Vol. 4, pp. 597–600). http://dx.doi.org/10.1109/ICPR.2004.1333843.
John, G. H., & Langley, P. (1995). Estimating continuous distributions in bayesian classifiers. In *Proceedings of the eleventh conference on uncertainty in artificial intelligence UAI'95* (pp. 338–345). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc. http://dl.acm.org/citation.cfm?id=2074158.2074196.
Khobragade, L. P., & Mahadik, P. (2015). Students academic failure prediction using data mining. *International Journal of Advanced Research in Computer and Communication Engineering, 4*.
Manhães, L. M. B., da Cruz, S. M. S., & Zimbrão, G. (2014). Wave: An architecture for predicting dropout in undergraduate courses using edm. In *Proceedings of the 29th annual ACM symposium on applied computing SAC '14* (pp. 243–247). New York, NY, USA: ACM. URL. http://dx.doi.org/10.1145/2554850.2555135.
Marquez-Vera, C., Morales, C., & Soto, S. (2013). Predicting school failure and dropout by using data mining techniques. *Tecnologias del Aprendizaje IEEE Revista Iberoamericana de, 8*, 7–14. http://dx.doi.org/10.1109/RITA.2013.2244695.
Martinho, V., Nunes, C., & Minussi, C. (2013). Prediction of school dropout risk group using neural network. In *Computer science and information systems (FedCSIS), 2013 federated conference on* (pp. 111–114).
Nürnberger, A., Pedrycz, W., & Kruse, R. (2002). In *Handbook of data mining and knowledge discovery. Chapter data mining tasks and Methods: Classi- fication: Neural network approaches* (pp. 304–317). New York, NY, USA: Oxford University Press, Inc. http://dl.acm.org/citation.cfm?id= 778212.778259.
Olson, D. L., & Delen, D. (2008). *Advanced data mining techniques* (1st ed.). Springer Publishing Company, Incorporated.
Pentaho. (2015). *Pentaho - Pentaho data integration*. http://www.pentaho.com/ Accessed January 2015.
Quinlan, J. R. (1986). Induction of decision trees. *Machine Learning, 1*, 81–106. http://dx.doi.org/10.1023/A: 1022643204877.
Rumelhart, D. E., Hinton, G. E., & Williams, R. J. (1988). Neurocomput- ing: Foundations of research. In *Chapter learning representations by back- propagating errors* (pp. 696–699). Cambridge, MA, USA: MIT Press. http://dl.acm.org/citation.cfm?id=65669.104451.
Salzberg, S. (1994). C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993. *Machine Learning, 16*, 235–240. http://dx.doi.org/10.1007/BF00993309.
Simon, Fincher, S., Robins, A., Baker, B., Box, I., Cutts, Q., et al. (2006). Predictors of success in a first programming course. In *Proceedings of the 8th Australasian conference on computing education - volume 52 ACE '06* (pp. 189–196). Darlinghurst, Australia, Australia: Australian Computer Society, Inc.. http://dl.acm.org/citation.cfm?id=1151869.1151894
Tan, P.-H., Ting, C.-Y., & Ling, S.-W. (2009). Learning difficulties in program- ming courses: Undergraduates' perspective and perception. In *Computer technology and development, 2009. ICCTD '09. International conference on* (Vol. 1, pp. 42–46). http://dx.doi.org/10.1109/ICCTD.2009.188.
Vapnik, V. N. (1995). *The nature of statistical learning theory*. New York, NY, USA: Springer-Verlag New York, Inc.
Viana, R., Rodrigues, R., Alvarez, M., & Pistori, H. (2007). Svm with stochas- tic parameter selection for bovine leather defect classification. In D. Mery, & L. Rueda (Eds.), *Advances in image and video technology* (pp. 600–612). Springer Berlin Heidelberg volume 4872 of Lecture Notes in Computer Science.
Watson, C., & Li, F. W. (2014). Failure rates in introductory programming revisited. In *Proceedings of the 2014 conference on innovation & tech- nology in computer science education ITiCSE '14* (pp. 39–44). New York, NY, USA: ACM. URL. http://dx.doi.org/10.1145/2591708.2591749.
Watson, C., Li, F., & Godwin, J. (2013). Predicting performance in an introductory programming course by logging and analyzing student programming behavior. In *Advanced learning Technologies (ICALT), 2013 IEEE 13th international conference on* (pp. 319–323). http://dx.doi.org/10.1109/ICALT.2013.99.
Weka (2015), Weka - the University of Waikato, http://www.cs.waikato.ac.nz/ml/weka/, Accessed January 2015.
Witten, I. H., Frank, E., & Hall, M. A. (2011). *Data Mining: Practical machine learning tools and techniques* (3rd ed.). San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.
Wu, X., Kumar, V., Ross Quinlan, J., Ghosh, J., Yang, Q., Motoda, H., et al. (2008). Top 10 algorithms in data mining. *Knowledge and Information Systems, 14*, 1–37. http://dx.doi.org/10.1007/s10115-007-0114-2.
Yukselturk, E., Ozekes, S., & Turel, Y. K. (2014). Predicting dropout student: An application of data mining methods in an online education program. *Eu- ropean Journal of Open, Distance and eLearning, 17*, 1027–5207.
Zaki, M. J., & W.M., , Jr.. (2014). *Data mining and Analysis: Fundamental concepts and algorithms*. New York, NY, USA: Cambridge University Press.