Framing Early Alert of Struggling Students as an Anomaly Detection Problem: An Exploration

Eitel J. M. Lauría
School of Computer Science & Mathematics, Marist College, Poughkeepsie, NY, U.S.A.

Keywords: Early Detection, At-risk Students, Anomaly Detection, Learning Analytics, Predictive Modeling, Machine Learning.

Abstract: This exploratory study analyses the feasibility of implementing an early-alert system of academically vulnerable students using anomaly detection techniques for cases in which the number of struggling students is small in comparison to the total student population. The paper focuses on a semi-supervised approach to anomaly detection where a first stage made up of an ensemble of unsupervised anomaly detectors contributes features to a second-stage binary classifier. Experiments are carried out using several semesters of college data to compare the predictive performance of this semi-supervised approach relative to stand-alone classification-based methods.

1 INTRODUCTION

In the last fifteen years the domain of academic and learning analytics has flourished, with many initiatives and projects being put in place to analyze and monitor the academic performance of students. Higher education has benefited from these implementations, that help students improve their academic performance and their chances of academic success, as well as aiding academic institutions in reducing student attrition, an issue that has direct impact on both the reputation and bottom line of colleges and universities. Most of these systems use predictive modeling and machine learning techniques to build models that help identify academically vulnerable students using student academic data, student demographic data, and student activity in the course (Arnold & Pistilli, 2012; Benablo et al., 2018; Jayaprakash et al., 2014; Lauría et al., 2016, 2019; Martins et al., 2019; Romero et al., 2013; Sheshadri et al., 2019; Zafra & Ventura, 2012).

The early alert of students at risk of poor performance and academic failure has the virtue of enabling early intervention, and early intervention enhances the chances of student success, as has been repeatedly demonstrated in the literature as well as through our work (Dodge et al., 2015; Harackiewicz & Priniski, 2018; Herodotou et al., 2019; Jayaprakash et al., 2014; Lauría et al., 2013; Lauría & Baron, 2011; Smith et al., 2012; Yao et al., 2019).

Different machine learning algorithms have been used by researchers to help improve the accuracy of their models, ranging from traditional statistical approaches like logistic regression (Campbell, 2007), to decision trees (Guleria et al., 2014), Support Vector Machines (Cardona & Cudney, 2019; Pang et al., 2017), Bayesian methods (Hamedi & Dirin, 2018), neural nets (Calvo-Flores et al., 2006; Okubo et al., 2017), the XGBoost algorithm (Chen & Guestrin, 2016; Hu & Song, 2019) and stack ensembles (Lauría et al., 2018).

All of these approaches have a common theme: implement a supervised learning framework, where models learn from past data and supervised learning is accomplished by labeling the data with the student performance in the form of numeric or letter grades, which can give way to regression or multiclass classification; or more typically through the recoding of the grades by establishing a minimum satisfactory threshold, such that students that perform below that threshold are considered at risk. The task of detecting struggling students can then be framed as a binary classification problem. Historically, this is the approach that has been followed by most institutions implementing early detection systems. The approach is valid, relevant, and relatively easy to implement in those institutions with moderate or large numbers of academically vulnerable students, as in that case the proportions of students in good standing and at academic risk are comparable in size -there is not a

Lauría, E.
Framing Early Alert of Struggling Students as an Anomaly Detection Problem: An Exploration.
DOI: 10.5220/0010471900260035
In *Proceedings of the 13th International Conference on Computer Supported Education (CSEDU 2021) - Volume 1*, pages 26-35
ISBN: 978-989-758-502-9

large difference in proportions- and therefore the success of the implementation is influenced by the quality of the training data, the accuracy of the algorithms used to train the models, and their reliability in terms of the bias-variance trade-off (successful algorithms try to keep low bias, while keeping variance at bay).

But in those institutions, like ours, where the proportion of at-risk students in any given semester is small or very small compared to the students in good standing (our College, for example has historically remained below 7%, with mean values slightly above 5%), the binary classification problem described in the previous paragraphs has the additional wrinkle of having to train models with a very high imbalance between both classes. Searching for potential at-risk students is like finding a needle in a haystack. Classifiers naturally tend to identify patterns in the majority class in detriment of the minority class. The small amount of training data tied to the minority class may limit the ability of the classification algorithms to produce reliable model parameter estimates (He & Garcia, 2009). Different approaches have been considered to address this issue, including the most popular one of balancing the training data through oversampling the minority class, subsampling the majority class, or a combination of both; but the topic of data balancing remains controversial in the machine learning community (Provost, n.d.). Still, results have been surprisingly good considering the difficulty of the problem, but evidently more research is needed.

One possible consideration, and the one discussed in this paper, is to regard the minority class as an anomaly or outlier. Anomaly detection is the task of detecting patterns that deviate atypically from what is expected. A typical characteristic of anomalies or outliers is that they are rare occurrences. Although anomaly detection can be considered as an extreme case of imbalanced classification (Kong et al., 2020), there is not much research that considers the possibility of shifting the classification paradigm in highly imbalanced data settings to one of anomaly detection, or improving it with the aid of anomaly detection outcomes. The latter approach, especially useful in the presence of labeled data, uses the outcomes of unsupervised anomaly detector models - called anomaly scores- as features to be added to the labeled training data in a subsequent binary classification stage, to try to improve the predictive performance of the classifier. This method has a recent implementation in the semi-supervised XGBOD algorithm (Zhao & Hryniewicki, 2018) and is the subject of this paper.

In this paper we therefore investigate the following research questions:

- Is a semi-supervised anomaly detection method a feasible approach for early detection of academically at-risk students?
- How does this approach compare to stand-alone classification methods?

The paper makes two main contributions: 1) it describes a methodology for implementing a two-stage semi-supervised anomaly detection algorithm in for early alert of small populations of struggling students, where an ensemble of unsupervised anomaly detection algorithms feeds a subsequent binary classifier; 2) it empirically compares the predictive performance of this approach with those of well-established and state of the art classification algorithms.

We begin with a brief review of the extant literature on anomaly detection, its applications and algorithms, focusing on unsupervised anomaly detection methods currently in use. Then we explain the semi-supervised methodology we chose to apply anomaly detection in our specific domain. We follow with a description of the experimental setup, including details of the input data, and the algorithms used in each of the different experiments. We present and discuss the experiments' results. Finally, we close the paper with comments on the limitations of the research and our conclusions.

2 ANOMALY DETECTION: ALGORITHMS AND APPLICATIONS

Anomaly detection has proven to be especially useful in a broad range of applications, including fraud detection, intrusion detection, fault detection, and identification of rare diseases in medical data. Outstanding values in credit card transactions may help analysts detect credit card fraud (Sharmila et al., 2019). Anomalous traffic patterns in network data can help identify malicious attacks (García-Teodoro et al., 2009). And the automated analysis of medical images using outlier detection techniques can help detect brain tumors (Wang et al., 2020).

As mentioned before, an anomaly or outlier is an observation that strongly deviates from the normal expectation. This means that an anomaly detection method should identify a decision function that separates normal from anomalous. But the task is challenging for several reasons: (i) the boundary

between normal and abnormal may not be clear cut; (ii) the notion and magnitude of an anomaly may be domain specific; (iii) data may be noisy; (iv) labeled data for training and validation purposes is usually scarce.

Numerous anomaly detection algorithms have been developed, drawn from traditional statistics, machine learning and lately deep learning. Their formulation is determined by the problem domain, the characteristics of the data (structured / unstructured), and availability of labeled data. It is not the purpose if this paper to provide a survey of anomaly detection methods, we refer the audience to (Chandola et al., 2009). In this paper we will focus on a set of unsupervised algorithms (unsupervised anomaly detection is the most common approach as it does not require labeled training data) and XGBOD as a semi-supervised technique. The anomaly detection algorithms considered, given the structured nature and limited dimensionality of the data are the following:

- K-nearest neighbors, (Knorr et al., 2000), referred to as KNN, a distance-based method, and closely related to the classifier of the same name. This family of methods assumes that normal observations occur close to each other whereas anomalies occur far away from their closest neighbors.
- Isolation Forest (Liu et al., 2008) uses recursive partitioning to create a tree structure to isolate anomalous data points -anomalies are easier to isolate and therefore have shorter tree path length. The process is repeated over multiple random trees and an average path length is computed, which is used as the outlier detector decision function.
- Local Outlier Factor, or LOF (Breunig et al., 2000) computes a score (called local outlier factor) that measures the local deviation of the data point with respect to the surrounding neighborhood, and with it its degree of anomaly.
- One-class SVM (Schölkopf et al., 2001) is an unsupervised extension of support vector machines that learns from a dataset containing data of only one class, and with it is able to identify anomalous data (outliers).

Unsupervised anomaly detection algorithms are sensitive to noise. Therefore combining them in an ensemble typically provides more stable results, an approach that follows the well-established ensemble methodology in supervised learning (Lauría et al., 2018; Zimek et al., 2014). The idea of extracting representations from the data through anomaly detection and inserting them as features in a classification setting was introduced by (Micenková et al., 2014) and (Aggarwal & Sathe, 2015). The XGBOD algorithm (Zhao & Hryniewicki, 2018) follows this same approach and derives its name from its use of the state of the art XGBoost classifier (Chen & Guestrin, 2016) in its second stage. In this paper we use XGBoost, but we also implement Random Forests (Breiman, 2001) and logistic regression as alternative second-stage classifiers with the purpose of widening the analysis of the predictive performance of this semi-supervised ensemble technique. We also consider different ways of combining multiple anomaly scores: averaging, maximization, straight forward use of all anomaly scores or feature selection (for details of the algorithm see section 3).

3 SEMI-SUPERVISED METHOD

We implement a two stage semi-supervised approach to train the models, with a first stage made up of an ensemble of anomaly detectors and a second stage given by a binary classifier.

In training mode: (see Figure 1)

- In **Step(i)** k anomaly detection algorithms $\mathrm{A}_1 \ldots \mathrm{A}_k$ are applied on training dataset D_{trn} and learn anomaly scores and decision functions algorithms $sc_1, df_1 \ldots sc_k, df_k$.
- In **Step(ii)** anomaly scores are chosen using some criterion $\varphi(sc_1 \ldots sc_k)$: all scores, the average of the scores, the best anomaly score, or a feature selection of anomaly scores. In the original XGBOD paper, the selection of anomaly scores is made by using a metric that balances the accuracy and diversity of the ensemble of anomaly detectors See (Zhao & Hryniewicki, 2018) for details.
- In **Step(ii)**, data set D_{trn} is supplemented with the selected anomaly scores $\varphi(sc_1 \ldots sc_k)$, resulting in $\mathrm{D}_{trn}^{(aug)}[X; \varphi(sc_1 \ldots sc_k); y]$.
- **Step(iii)** Training Classifier: use data set $\mathrm{D}_{trn}^{(aug)}[X; \varphi(sc_1 \ldots sc_k); y]$ as input data to train model M using classifier C.

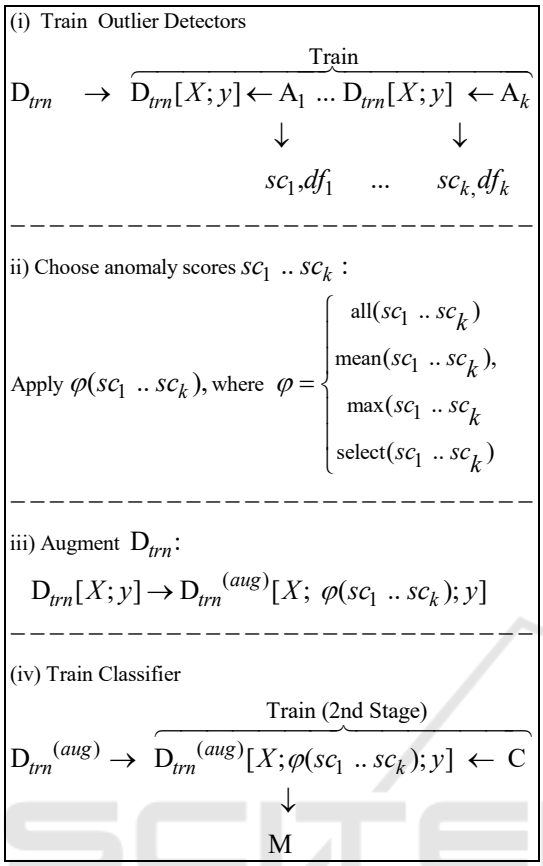

Figure 1: Training mode.

In prediction mode:

- In **Step(i)**, decision functions $df_1 \ldots df_k$ are applied on data set D_{tst} to compute predicted scores $psc_1 \ldots psc_k$.
- In **Step(ii)** predicted anomaly scores are chosen using the same criterion $\varphi(psc_1 \ldots psc_k)$ used during training mode.
- In **Step(iii)**, data set D_{tst} is supplemented with predicted anomaly scores $\varphi(psc_1 \ldots psc_k)$, resulting in $D_{tst}^{(aug)}[X; \varphi(psc_1 \ldots psc_k); y]$.
- **Step(iv)** In it, binary classification model M is applied on data set $D_{tst}^{(aug)}$ to make predictions. The classifier reports predictions and probability of predictions $(\hat{y}_{tst}; prob_{tst})$.

4 EXPERIMENT SETUP

In the experiments we investigated whether the semi-supervised ensemble approach described in section 3

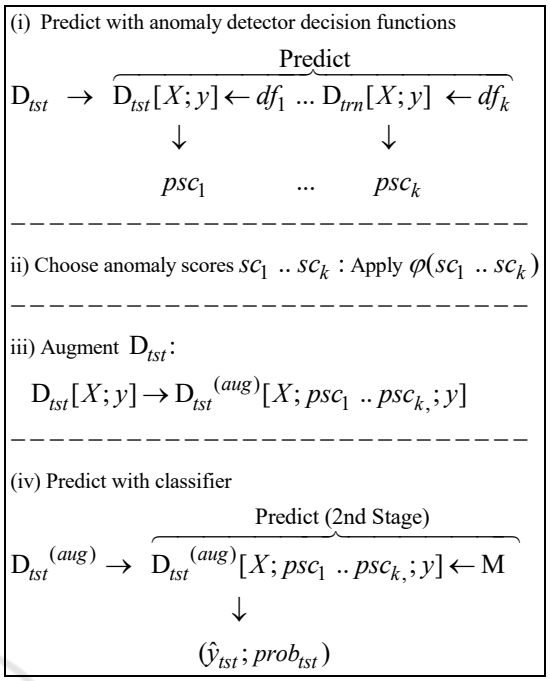

Figure 2: Prediction mode.

when compared to binary classifiers. The goal is to learn from the data in order to detect early on in the semester (6 weeks into a semester of 15 weeks) those students that are struggling in their course.

4.1 Datasets

For the purpose of this study we used four semesters of undergraduate 15-week courses, corresponding to Fall 2018, Spring 2019, Fall 2019 and Fall 2020, enriched with student academic data, some demographics and course activity metrics collected from the LMS logs. Data was extracted from four sources: (i) student demographic and aptitude data; (ii) course related data and students' final grades in those courses; c) student activity data logged by the LMS, corresponding to the first six weeks of each semester; (iv) a composite score aggregating grades on assignments, projects, exams and any other activities contributing to the student's final grade in the first six weeks of the semester, logged by the LMS's gradebook tool. Data was subsequently transformed and cleaned into a complete unit of analysis without missing values.

The LMS activity data (number of LMS logins, number of access to LMS resources, and total activity over all LMS tools) was aggregated weekly into frequency values computed as ratios between the student activity and the average class activity, to account for potential variability among different courses.

Data was pre-processed and aggregated into a unit of analysis -students' data in each course over four semesters. The binary label for each record in the unit of analysis was computed by recoding the student's final grade in the course: those students with a final letter grade C or more were considered in good standing, whereas those students with less than a final letter grade C were considered academically vulnerable (at risk) students. Table 1 depicts the file structure of the unit of analysis.

Table 1: File structure of the unit of analysis.

Predictors	Data type
Enrolment	Numeric
Online	Categorical
Age	Numeric
GPA	Numeric
Aptitude Score (SAT/ACT)	Numeric
Gender	Categorical
Class (Fresh, Soph, Jr, Sr)	Categorical
LMS Total Activity (weeks 1-6)	Numeric x 6
Login (weeks 1-6 + sum)	Numeric x 6
Content Read (weeks 1-6 + sum)	Numeric x 6
Gradebook Composite Score (wks 1-6)	Numeric
Target feature: Academic_Risk (1=at risk; 0=good standing)	

The full unit of analysis contained data of all four semesters (Fall 2018 – Spring 2020) with the following proportions of good-standing and at-risk students (see Table 2):

Table 2: Data per semester.

Semester	Total Count	Percent at risk
Fall 2018	10809	6.6%
Spring 2019	14133	6.8%
Fall 2019	11089	6.1%
Spring 2020	17206	3.0%

4.2 Methods

Each experiment was performed by randomly selecting a semester and subsequently partitioning the semester into training and testing data using an 80/20 ratio. We selected 30 randomly chosen (semester, partition) pairs, creating 30 dataset pairs (training and testing) to perform experiment runs, using 4 anomaly detection methods with varying parameters, 3 selection criteria for anomaly scores (all, average, max) and 3 binary classifiers. We also implemented a feature selection criterion of the anomaly scores prior to the XGBoost classifier as presented in the XGBOD original paper.

Additionally, we trained all three classifiers with balanced data for comparison purposes (see section 4.2.3). This amounted to a total of 16 experiments repeated over 30 runs, for a total of 480 experiments (for details see section 4.3 and Table 3).

4.2.1 Anomaly Detection Algorithms

Four detection algorithms were used in the experiment:

- KNN: K nearest Neighbours with mean, median and largest distance to the kth neighbour and with k=[1,2,3,4,5,6,7,8,9,10,15, ..,100].
- LOF: Local Outlier Factor with k=[1,2,34,5, 6,7,8,9,10,15,..,100].
- IForest: Isolation Forest with number of base estimators = [10, 30, 50, 70, 100, 150, 200, 250].
- OCSVM: One-class Support Vector Machines with radial basis kernel and different upper bound on the fraction of training errors.

A total of 115 anomaly scores were added to each dataset.

4.2.2 Classifiers

We used three classification algorithms as second stage classifiers:

- XGB: The XGBoost algorithm
- RF: Random Forests
- LOG: regularized logistic regression.

The XGBoost classifier (Chen & Guestrin, 2016), a tree-based classification algorithm, is currently one of the most powerful supervised methods and a popular choice in Kaggle competitions.

The Random Forests algorithm (Breiman, 2001) is a well-regarded ensemble learning technique that builds multiple random CART trees and outputs the most frequently predicted class.

Logistic regression is the workhorse of traditional statistics and an effective classifier when dealing with data that holds both numeric and binary data.

For both XGBoost and Random Forests we fixed the number of estimators to 600. We did not perform any further hyperparameter tuning, to reduce the computational cost of the experiment.

4.2.3 Balanced Data

For comparison purposes, the three classifiers were also trained with balanced data, after balancing the proportions of good-standing and at-risk students, for each of the 30 randomly generated datasets, The

SMOTE algorithm (Bowyer et al., 2011) was used for balancing the data. SMOTE (Synthetic Minority Over-sampling Technique) is an oversampling method that augments the minority class by synthesizing new minority samples instead of using a simple duplication of samples.

4.2.4 Computational Details

The models were coded in Python 3.7 using several libraries. All the unsupervised anomaly detection algorithms were implemented using the PyOD library (Zhao et al., 2019). Although PyOD provides a turn-key implementation of XGBOD, we decided to code our own version of the algorithm to have better control and add flexibility to the execution process, including other second-stage classifiers besides XGBoost, and different anomaly scores selection criteria. For the classifiers we used the Scikit-learn library (Pedregosa et al., 2011), including its API for XGBoost. The experiments were run on an Intel Xeon Silver server, 20 cores 3.00GHz, 2 threads per core, 32GB RAM, 1 GPU (Quadro P4000 8GB). Parallel processing features were used when available for both the first stage (anomaly detection) and second stage (binary classification) algorithms to make use of the multi-core platform and with it reduce the execution time of the experiments.

4.2.5 Assessment

At prediction times the following predictive performance metrics were collected:

- ROC_AUC: the AUC (area under the curve) of the receiver operating characteristics (ROC) curve, that plots TP rate vs FP rate, is the most widely used metric to summarize binary classification performance. Although it is still the most popular metric in imbalanced classification it has received criticism in the case of highly imbalanced settings as it can be overly optimistic due to unreliability of the estimates under class rarity (Fernández et al., 2018).
- PR_AUC: The AUC of the precision-recall curve assesses the performance of the classifier on the minority class and can therefore be more informative of the algorithm performance than the ROC AUC metric (Davis & Goadrich, 2006).
- Training and Prediction execution time, measured in seconds: this could be relevant, particularly for prediction, as the large ensemble of anomaly detectors can introduce a considerable overhead.

Table 3: Predictive Performance Results.

	ROC AUC		PR AUC		TRAIN TIME		PREDICT TIME	
	Mean	SE	Mean	SE	Mean	SE	Mean	SE
XGB	0.9511	0.0012	0.5990	0.0219	1.78	0.06	1.35	0.06
XGBOD	0.9502	0.0012	0.5946	0.0229	320.96	21.77	99.35	14.42
ALL_XGB	0.9508	0.0012	0.5959	0.0217	320.31	26.51	99.27	7.78
AVG_XGB	0.9514	0.0012	0.5998	0.0217	322.71	18.00	102.14	12.22
BEST_XGB	0.9510	0.0012	0.5984	0.0219	315.24	21.47	95.19	11.06
SMOTE_XGB	0.9430	0.0014	0.5346	0.0267	2.73	0.12	2.73	0.12
RF	0.9433	0.0017	0.5914	0.0198	3.58	0.02	3.58	0.02
ALL_RF	0.9246	0.0022	0.5523	0.0205	318.25	31.36	105.84	9.44
AVG_RF	0.9435	0.0017	0.5912	0.0202	319.51	29.13	114.39	8.91
BEST_RF	0.9433	0.0017	0.5918	0.0200	329.42	26.86	99.18	11.12
SMOTE_RF	0.9426	0.0014	0.4900	0.0209	4.48	0.07	4.48	0.07
LOG	0.9261	0.0028	0.5178	0.0206	1.12	0.02	1.12	0.02
ALL_LOG	0.8513	0.0075	0.3007	0.0162	315.98	24.03	105.44	11.09
AVG_LOG	0.8861	0.0075	0.3866	0.0213	314.58	26.37	96.24	8.56
BEST_LOG	0.9023	0.0063	0.4501	0.0224	312.50	25.41	103.46	9.34
SMOTE_LOG	0.9343	0.0019	0.5535	0.0211	1.93	0.03	1.18	0.03

4.3 Results and Discussion

Table 3 displays the assessment of mean predictive performance of the different configurations of the semi-supervised approach considered (second stage trained by adding all anomaly scores, by adding the average anomaly score, by adding only the maximum anomaly score, and by using the feature selection criterion on the anomaly scores described in the XGBOD paper), as well as the mean predictive performance of the stand-alone classifiers trained with both the original training data, and after balancing the training data:

- XGB, RF, LOG: no anomaly detection scores used by second stage classifier (stand-alone classifier, original training data)
- ALL_XGB, ALL_RF, ALL_LOG: all anomaly detection scores were used by the second stage classifier.
- AVG_XGB, AVG_RF, AVG_LOG: the average anomaly detection score was used by the second stage classifier.
- BEST_XGB, BEST_RF, BEST_LOG: the maximum anomaly detection score was used by the second stage classifier.
- XGBOD: the XGBOD algorithm with anomaly score feature selection as proposed by (Zhao & Hryniewicki, 2018).
- SMOTE_XGB, SMOTE_RF, SMOTE_LOG: stand-alone classifiers trained after balancing the original training data (as described in section 4.2.3).

For the XGBoost algorithm, adding the average anomaly score slightly improved both ROC_AUC and ROC_PR: ROC_AUC for AVG_XGB was 0.9514 and PR_AUC was 0.5998, compared to ROC_AUC=0.9511 and PR_AUC=0.5990 for the XGBoost stand-alone classifier, trained without anomaly scores. The other approaches (XGBOD, ALL_XGB, BEST_XGB) degraded the performance instead of improving it. All algorithms performed better in both metrics than the balanced alternative (SMOTE_XGB). Random Forests had mixed results: AVG_RF performed slightly better only for ROC_AUC (0.935 vs 0.9433), but the rest of the approaches had either the same or worse performance metrics than RF. All classifiers performed better than the balanced alternative (SMOTE_RF), except ALL_RF for the ROC_AUC metric.

We ran the Wilcoxon signed test for both XGBoost and Random Forests comparing the metrics of the classifier trained without anomaly scores with each of the different methods for choosing anomaly scores, and the tests were not strong enough to identify significant differences ($p>0.4$ in all cases). There were though significant differences in all cases ($p<0.001$) between all unbalanced configurations and the balanced configurations for both XGBoost and Random Forests (SMOTE).

For logistic regression, the addition of anomaly scores to the training data generally degraded the performance metrics. The balanced data approach also outperformed all other approaches, including training the model without anomaly scores: ROC_AUC was 0.9343 for SMOTE_LOG compared to 0.9261 for LOG. And PR_AUC was 0.5535 compared to 0.5178 for LOG.

Execution time exposes the overhead of running 115 unsupervised anomaly detectors on the data before training the classifier or using it for prediction. Table 3 shows that it took about 3 seconds per outlier detector in training mode and a little less than a second per outlier detector in predict mode. Instead, the execution time of the binary classifiers was negligible in comparison. Training time is not an actual issue given that training does not typically happen in real time; but larger execution times in prediction mode could pose a problem. This is of course a relatively minor issue as it is dependent on the hardware platform used to train and implement the prediction models, but is worth mentioning when comparing the anomaly detection approach with binary classification approaches, which do not require the computation of added anomaly scores.

5 CONCLUDING COMMENTS

In summary, the anomaly detection models did not seem to be able to learn representations that efficiently contribute to the second-stage classifiers. It is a reasonable assumption that the effectiveness of the semi-supervised approach could be dependent on the domain in which it is applied. Anomalies are rare events by definition, but a small proportion of at-risk students may not necessarily qualify as a set of anomalies; the difference between good-standing and struggling students may not be big enough and therefore those struggling students may not necessarily qualify as outliers, the patterns of academic struggle being subtler, which in turn would require more fine tuning of the thresholds that determine anomaly scores.

The current research has several limitations. We did not perform any hyperparameter tuning of the classifiers to limit the execution time of the experiments (XGBoost and Random Forest, would

have benefitted from performance tuning). Also, the study imposed four unsupervised outlier detection algorithms. They are among the most relevant algorithms for the type of data considered, and we did vary their parameters to produce multiple anomaly outcomes, but still, the choice was limited. Other algorithms could also be included to increase the variety of the outlier detectors. Subsampling could be applied to the data used to train the anomaly detection algorithms as proposed by (Aggarwal & Sathe, 2015). And anomaly scores resulting from the anomaly detection algorithms could be subjected to dimensionality reduction, consequently inducing features to be used by the second-stage classifier, rather than directly adding the anomaly scores as new features.

The objective of the study is exploratory and has the purpose of exemplifying the approach applied to early detection of small populations of students at risk and providing a proof of concept, as well as empirically testing its performance. The results are non-conclusive as to the benefits of this approach, but nonetheless, the study is an initial application of the use of a semi-supervised framework int this domain that combines anomaly detection and binary classification.

Hopefully, this paper will open a research avenue for other researchers and practitioners towards new methods in early detection of academically at-risk students using machine learning techniques.

ACKNOWLEDGEMENTS

The author would like to thank Ed Presutti and the rest of the Data Science & Analytics team at Marist College for their continuous collaboration. Maria Kapogiannis provided and double checked the datasets used in the experiments.

REFERENCES

Aggarwal, C. C., & Sathe, S. (2015). Theoretical Foundations and Algorithms for Outlier Ensembles. SIGKDD Explor. Newsl., 17(1), 24–47.

Arnold, K. E., & Pistilli, M. D. (2012). Course signals at Purdue: Using learning analytics to increase student success. Proceedings of the 2nd International Conference on Learning Analytics and Knowledge.

Benablo, C. I. P., Sarte, E. T., Dormido, J. M. D., & Palaoag, T. (2018). Higher Education Student's Academic Performance Analysis through Predictive Analytics. Proceedings of the 2018 7th International Conference on Software and Computer Applications, 238–242. https://doi.org/10.1145/3185089.3185102

Bowyer, K. W., Chawla, N. V., Hall, L. O., & Kegelmeyer, W. P. (2011). SMOTE: Synthetic Minority Over-sampling Technique. CoRR, abs/1106.1813.

Breiman, L. (2001). Random Forests. Mach. Learn., 45(1), 5–32.

Breunig, M. M., Kriegel, H.-P., Ng, R. T., & Sander, J. (2000). LOF: Identifying Density-Based Local Outliers. Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, 93–104. https://doi.org/10.1145/342009.335388

Calvo-Flores, M. D., Galindo, E. G., & Jiménez, M. P. (2006). Predicting students' marks from Moodle logs using neural network models.

Campbell, J. P. (2007). Utilizing student data within the course management system to determine undergraduate student academic success: An exploratory study (UMI No. 3287222) [Doctoral Dissertation]. Doctoral Dissertation, Purdue University.

Cardona, T. A., & Cudney, E. a. (2019). Predicting Student Retention Using Support Vector Machines. 25th International Conference on Production Research Manufacturing Innovation: Cyber Physical Manufacturing August 9-14, 2019 | Chicago, Illinois (USA), 39, 1827–1833.

Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly Detection: A Survey. ACM Comput. Surv., 41(3).

Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. CoRR, abs/1603.02754. http://arxiv.org/abs/1603.02754

Davis, J., & Goadrich, M. (2006). The Relationship between Precision-Recall and ROC Curves. Proceedings of the 23rd International Conference on Machine Learning, 233–240.

Dodge, B., Whitmer, J., & Frazee, J. P. (2015). Improving undergraduate student achievement in large blended courses through data-driven interventions. Proceedings of the Fifth International Conference on Learning Analytics And Knowledge.

Fernández, A., García, S., Galar, M., Prati, R. C., Krawczyk, B., & Herrera, F. (2018). Learning from imbalanced data sets. Springer.

García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., & Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, 28(1), 18–28.

Guleria, P., Thakur, N., & Sood, M. (2014). Predicting student performance using decision tree classifiers and information gain. 2014 International Conference on Parallel, Distributed and Grid Computing, 126–129.

Hamedi, A., & Dirin, A. (2018). A Bayesian approach in students' performance analysis.

Harackiewicz, J. M., & Priniski, S. J. (2018). Improving Student Outcomes in Higher Education: The Science of Targeted Intervention. Annual Review of Psychology, 69, 409–435. PubMed.

He, H., & Garcia, E. A. (2009). Learning from Imbalanced Data. IEEE Transactions on Knowledge and Data Engineering, 21(9), 1263–1284.

Herodotou, C., Rienties, B., Boroowa, A., Zdrahal, Z., & Hlosta, M. (2019). A large-scale implementation of predictive learning analytics in higher education: The teachers' role and perspective. Educational Technology Research and Development, 67(5), 1273–1306. https://doi.org/10.1007/s11423-019-09685-0

Hu, T., & Song, T. (2019). Research on XGboost academic forecasting and analysis modelling. Journal of Physics: Conference Series, 1324, 012091.

Jayaprakash, S. M., Moody, E. W., Lauría, E. J. M., Regan, J. R., & Baron, J. D. (2014). Early Alert of Academically At-Risk Students: An Open Source Analytics Initiative. Journal of Learning Analytics, 1(1), 42.

Knorr, E. M., Ng, R. T., & Tucakov, V. (2000). Distance-Based Outliers: Algorithms and Applications.

Kong, J., Kowalczyk, W., Menzel, S., & Bäck, T. (2020). Improving Imbalanced Classification by Anomaly Detection. In T. Bäck, M. Preuss, A. Deutz, H. Wang, C. Doerr, M. Emmerich, & H. Trautmann (Eds.), Parallel Problem Solving from Nature (PPSN XVI) (pp. 512–523). Springer International Publishing.

Lauría, E. J. M., & Baron, J. (2011). Mining Sakai to Measure Student Performance: Opportunities and Challenges in Academic Analytics. ECC 2011.

Lauría, E. J. M., Moody, E. W., Jayaprakash, S. M., Jonnalagadda, N., & Baron, J., D. (2013). Open academic analytics initiative: Initial research findings. Proceedings of the Third International Conference on Learning Analytics and Knowledge.

Lauría, E. J. M., Presutti, E., & Kapogiannis, M. (2019, June). Of Stacks and Muses: Adventures in Learning Analytics at Marist College. LatinX in AI Research at ICML 2019. https://hal.archives-ouvertes.fr/hal-02265832

Lauría, E. J. M., Presutti, E., Kapogiannis, M., & Kamath, A. (2018). Stacking Classifiers for Early Detection of Students at Risk. Proceedings of the 10th International Conference on Computer Supported Education - Volume 2: CSEDU 2018, 390–397.

Lauría, E. J. M., Presutti, E., Sokoloff, M., & Guarino, M. (2016). Crossing the Chasm to Big Data: Early Detection of at-Risk Students in a Cluster Computing Environment. Proceedings of the 7th International Learning Analytics & Knowledge Conference (LAK'17)- Practitioner Track . Vancouver, Canada.

Liu, F. T., Ting, K. M., & Zhou, Z.-H. (2008). Isolation Forest. Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, 413–422.

Martins, M. P. G., Miguéis, V. L., Fonseca, D. S. B., & Alves, A. (2019). A Data Mining Approach for Predicting Academic Success – A Case Study. In Á. Rocha, C. Ferrás, & M. Paredes (Eds.), Information Technology and Systems (pp. 45–56). Springer International Publishing.

Micenková, B., Mcwilliams, B., & Assent, I. (2014). Learning Outlier Ensembles: The Best of Both Worlds – Supervised and Unsupervised. 1–4.

Okubo, F., Yamashita, T., Shimada, A., & Ogata, H. (2017). A neural network approach for students' performance prediction. Proceedings of the Seventh International Learning Analytics & Knowledge Conference.

Pang, Y., Judd, N., O'Brien, J., & Ben-Avie, M. (2017). Predicting students' graduation outcomes through support vector machines. 2017 IEEE Frontiers in Education Conference (FIE), 1–8.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine Learning in Python. Journal of Machine Learning Research, 12, 2825–2830.

Provost, F. (n.d.). Machine Learning from Imbalanced Data Sets 101 (Extended Abstract).

Romero, C., López, M.-I., Luna, J.-M., & Ventura, S. (2013). Predicting students' final performance from participation in on-line discussion forums. Computers & Education, 68, 458–472.

Schölkopf, B., Platt, J. C., Shawe-Taylor, J., Smola, A. J., & Williamson, R. C. (2001). Estimating the support of a high-dimensional distribution. Neural Computation, 13(7), 1443–1471.

Sharmila, V. C., R, K. K., R, S., D, S., & R, H. (2019). Credit Card Fraud Detection Using Anomaly Techniques. 2019 1st International Conference on Innovations in Information and Communication Technology (ICIICT), 1–6.

Sheshadri, A., Gitinabard, N., Lynch, C. F., Barnes, T., & Heckman, S. (2019). Predicting Student Performance Based on Online Study Habits: A Study of Blended Courses. CoRR, abs/1904.07331.

Smith, V. C., Lange, A., & Huston, D. R. (2012). Predictive Modeling to Forecast Student Outcomes and Drive Effective Interventions in Online Community College Courses. Journal of Asynchronous Learning Networks, 16(3), 51–61. eric.

Wang, N., Chen, C., Xie, Y., & Ma, L. (2020). Brain Tumor Anomaly Detection via Latent Regularized Adversarial Network.

Yao, H., Lian, D., Cao, Y., Wu, Y., & Zhou, T. (2019). Predicting Academic Performance for College Students: A Campus Behavior Perspective. ACM Trans. Intell. Syst. Technol., 10(3). https://doi.org/10.1145/3299087

Zafra, A., & Ventura, S. (2012). Multi-instance genetic programming for predicting student performance in web based educational environments. Applied Soft Computing, 12(8), 2693–2706.

Zhao, Y., & Hryniewicki, M. K. (2018). XGBOD: Improving Supervised Outlier Detection with Unsupervised Representation Learning. 2018 International Joint Conference on Neural Networks (IJCNN).

Zhao, Y., Nasrullah, Z., & Li, Z. (2019). PyOD: A Python Toolbox for Scalable Outlier Detection. Journal of Machine Learning Research, 20(96), 1–7.

Zimek, A., Campello, R. J. G. B., & Sander, J. (2014). Ensembles for Unsupervised Outlier Detection:

Challenges and Research Questions a Position Paper.
SIGKDD Explor. Newsl., 15(1), 11–22.