A Statistical Framework for Predictive Model Evaluation in MOOCs

Josh Gardner
School of Information, University of Michigan
Ann Arbor, MI 48109, USA
jpgard@umich.edu

Christopher Brooks
School of Information, University of Michigan
Ann Arbor, MI 48109, USA
brooksch@umich.edu

ABSTRACT

Feature extraction and model selection are two essential processes when building predictive models of student success. In this work we describe and demonstrate a statistical approach to both tasks, comparing five modeling techniques (a lasso penalized logistic regression model, naïve Bayes, random forest, SVM, and classification tree) across three sets of features (week-only, summed, and appended, from [7]). We conduct this comparison on a dataset compiled from 30 total offerings of five different MOOCs run on the Coursera platform. Through the use of the Friedman test with a corresponding post-hoc Nemenyi test, we present comparative performance results for several classifiers across the three different feature extraction methods, demonstrating a rigorous inferential process intended to guide future analyses of student success systems.

Author Keywords

Predictive modeling; machine learning; MOOC; evaluation

INTRODUCTION

Building predictive models of student success has emerged as a core task in the fields of learning analytics and educational data mining. As these fields continue to grow, and as offerings of Massive Open Online Courses (MOOCs) continue to expand, the need for building effective and reliable predictive models of student success grows too. Despite this, a literature survey[1] by the authors in these areas indicates that research on the creation of models of student success often neglects accepted statistical practices for model comparison. In particular, more than half of the predictive research surveyed did not utilize any statistical testing for the evaluation of model comparisons, despite performing these comparisons in situations where such testing is necessary (such as when model performance is estimated directly on the training set through 10-fold cross-validation). In many cases where significance testing was performed, details with respect to the precision or confidence of estimates were not reported. This leaves such research susceptible to potentially spurious results and low replicability due to concerns with multiple comparisons (comparing multiple algorithms often with multiple hyperparameter settings for each), uncorrected biased estimates caused by estimating model performance directly on training data, and the randomization inherent in sampling schemes such as cross-validation or repeated random subsampling that are used to obtain estimates of model performance.

The lack of adoption of statistical techniques for model evaluation in the field of learning analytics is not due to their nonexistence. Statistical testing of model significance is common in fields such as economics and public policy, where F-testing and ANOVA are frequently used to draw inferences about comparisons between regression models. The broader machine learning research community has produced several additional tools well-suited to evaluating more complex predictive models of student success. In particular, [1] catalogues several approaches for statistical inference about model comparison in various contexts, recommending a Friedman test paired with a post-hoc Nemenyi test for inference about comparisons between multiple models across multuple datasets. Also presented in [1] is an information-dense, interpretable visualization for displaying the results of this procedure, the Critical Difference (CD) diagram. This approach has been implemented in several comparative works, such as [5], but to our knowledge has not been applied to predictive models in learning analytics.

Thus this paper makes two contributions to the area of predictive modeling in education. First, we describe methodological deficiencies in the current practice and outline appropriate methods to mediate these issues, basing our discussion in the statistical machine learning literature. Second, we implement a procedure that addresses these issues through a comparison of five different machine learning algorithms applied to five diverse MOOCs. The results of this comparison demonstrate the value of this rigor-and the limits it enforces on overconfident interpretations of comparisons of multiple algorithms. We introduce this methodology here in context in order to better disseminate the approach and increase rigor and reproducibility in the field.

[1] This survey reviewed the 2014-2016 years of the conference proceedings for the annual International Society for Educational Data Mining (EDM) and the International Learning Analytics and Knowledge (LAK) conferences.

L@S 2017, April 20 - 21, 2017, Cambridge, MA, USA

ISBN 978-1-4503-4450-0/17/04...$15.00
DOI: http://dx.doi.org/10.1145/3051457.3054002

Course	# Offerings	# Students
Intro. to Thermodynamics	5	16,511
Instr. Meth. in Health Prof. Edu.	5	6,212
Fantasy and Science Fiction	8	26,580
Intro. to Finance	7	181,797
Inside the Internet	5	28,229

Table 1. Courses used to build feature sets. Data from all available runs of each course were combined into a single dataset to meet the independence assumptions described below.

DATASET

The dataset used in this analysis is extracted from the raw text clickstream files from 30 offerings of five unique courses at the University of Michigan between 2012 and 2016. A summary of this dataset is shown in Table 1.

We used three different feature engineering methods, described in [7], to derive weekly feature sets from the raw clickstream logs. These different feature approaches represent different methods for aggregating temporal data in predictive models of student success through replicable and platform-independent features. The three feature sets are built from the same basic set of feature definitions, shown in Table 2, using different methods of aggregation. The *week-only* feature sets contain only the values of each feature for the week at which the model is trained (and no data about any other week); *summed* feature sets contain the total values of each feature up to and including the training week; *appended* feature sets concatenate a new set of 8 columns for each week up to and including the training week (e.g. distinct columns for week 1 forum views, week 2 forum views, etc.). The training week in this experiment was the third week, with the goal of prediction being whether students would drop out in the following week of the course. These extraction methods each represent a different approach to capturing student behavior over time in the course, and we direct the interested reader to [7] for details.

After extracting features within each run of each course, data for all runs of each course were combined, because we assumed a high degree of dependence within courses, and therefore utilizing multiple runs of an identical course as "different" datasets would violate the assumptions of dataset independence underlying the comparison process outlined below. This yielded a total of 15 datasets (5 courses $*$ 3 feature extraction methods).

METHODOLOGY

Our goal in this project was to demonstrate a statistical approach to the task of selecting the best of $k > 2$ models across $N > 1$ datasets. While a set of accepted statistical practices for comparing $k = 2$ models across a single dataset is also needed, the multiple-model-multiple-dataset case matches the practical situations learning analytics researchers face most often.[2] Using the feature sets applied above, we trained a set of five classifiers commonly used in predictive models of student success (a lasso penalized logistic regression model, naïve bayes, random forest, SVM, and classification tree) and evaluated their accuracy using 10-fold cross-validation across each of the five datasets within each of the three feature extraction methods. No hyperparameter optimization was performed, and default or standard rule-of-thumb values were chosen for necessary hyperparameters to avoid conducting additional comparisons. The *performanceEstimation* package in R was used as the framework for model building and evaluation [6][3].

Feature	Description
Forum Views	Count of pageviews of course forum pages.
Active Days	Count of days learner registered any activity in the course (maximum of 7).
Quiz Attempts	Count of attempted quiz questions.
Quiz Exams	Count of attempted exam questions.
Quiz Human Graded	Count of attempted human-graded quiz questions.
Forum Posts	Count of forum posts.
Direct Nodes	Count of distinct users a given user responded to on the forums (direct-reply).
Thread Nodes	Count of distinct uses a given user posted in the same forum with (thread-reply).

Table 2. Definitions of features used to build week-only, summed, and appended feature sets from raw clickstream logs.

To draw inferences about the respective differences in model performance, we implemented a procedure from [1], applying it separately to each of the three feature sets. The procedure consists of two steps: first, a Friedman test (non-parametric equivalent of the repeated-measures ANOVA) is used to test the null hypothesis that all the algorithms are equivalent. The Friedman test compares the average rankings of the k algorithms across each of the N datasets, calculating a test statistic measuring the probability of the observed rankings under the null hypothesis of all algorithms having equivalent performance (and therefore equal expected average rankings). The observed value of the Friedman statistic

$$\chi_F^2 = \frac{12N}{k(k+1)} \left[\sum_j R_j^2 - \frac{k(k+1)^2}{4} \right] \tag{1}$$

where R_i^j is the rank of the jth of k algorithms on N datasets and the statistic is distributed according to a chi-square distribution with $k-1$ degrees of freedom, is compared to a critical value for the given values of N and k [3]. If the null hypothesis is rejected at the selected significance level ($\alpha = 0.05$ in this experiment), the post-hoc Nemenyi test is used to compare all classifiers to each other. The Nemenyi test is similar to the Tukey test for ANOVA, and uses a critical difference

$$CD = q_\alpha \sqrt{\frac{k(k+1)}{6N}} \tag{2}$$

[2] See [4] for a discussion of the two-model, single-dataset case.

[3] The models utilized within this framework were from several different standard R packages; contact the authors of this paper for details of implementation and specific models used.

as a threshold to determine whether the performance between any two classifiers is significantly different, where the critical value q_α is based on the Studentized range statistic divided by $\sqrt{2}$.

One advantage of this method is that because the Friedman test uses only the *rankings* of the algorithms on each dataset, it does not require estimates of the variance of model performance. Instead, it only requires that the estimates of model performance and the measured rankings they produce are reliable and "...that enough experiments were done on each data set and, preferably, that all the algorithms were evaluated using the same random samples..." [1] and that the datasets, and therefore the rankings of the algorithms across each dataset, are independent. In contrast to many other statistical approaches to comparing model performance, such as ANOVA, the Friedman test makes no further assumptions about the sampling scheme (in contrast, discussion of the two-model-single-dataset case is almost entirely centered on methods for sampling schemes in an effort to estimate the variance of model performance and fit within the assumptions of the testing procedures used, for example, [2]).

RESULTS

The results of our comparisons are presented in the Critical Difference (CD) diagrams proposed in [1] in Figure 1, and in a more detailed tabular format displaying the performance and respective rankings of each algorithm in Table 3. The CD diagrams visualize the results of the post-hoc Nemenyi test in a compact, information-dense format. The colored line for each algorithm shows its average rank in comparison with the other algorithms across all five course datasets. The bold black line shows the critical difference for the comparison (based on the values of N, k, and α). Models separated by a distance of less than the CD are statistically indistinguishable–the data is not sufficient to conclude whether they have the same performance–and are connected by a black line segment. Models separated by a distance greater than the critical distance have a statistically significant difference in performance.

Our results demonstrate that, while there are several apparent differences in the model performance based on a cursory evaluation of the performance matrix in Table 3, the experimental data only allows us to conclude that a small number of these differences in performance are statistically significant, within any given feature set (*week-only*, *summed*, and *appended*). For week-only features (the top CD diagram in Figure 1), the only statistically significant difference in performance is between the naïve Bayes and rpart (classification tree) algorithms, with all others showing differences of average ranks less than or equal to the critical difference. For summed features (the middle CD diagram in Figure 1), we see a statistically significant difference between both the rpart and SVM algorithms and the naïve bayes, with no significant difference detected between the SVM and rpart algorithms. The appended feature set shows a more nuanced set of differences in performance, and detailed interpretation is left to the reader.

Several features of these comparisons are worth noting. First, the critical difference ($CD = 2.7$ in all three comparisons), calculated according to Equation 2, is already quite high. Adding additional comparisons between algorithms, and thus increasing k while holding N constant-as would be the case if multiple variants of each model with different hyperparameters were tested-would quickly inflate the value of CD due to an increased likelihood of spurious differences in performance. This calls into question the current practice of training many algorithms with a breadth of hyperparameter settings, which has become easy due to increased comprehensiveness of machine learning toolkits, and simply selecting the highest-performing. Second, this comparison (and the accompanying matrix of results shown in Table 3) demonstrates the effect of this procedure's analysis of algorithms' relative ranking, but not their absolute performance. While we can see that the lasso penalized logistic regression model (GLM) consistently achieved performance scores much higher than the naïve bayes (NB) across all datasets, the average rankings were still quite close-leading to statistically indistinguishable performance under this testing procedure. By contrast, the classification tree (rpart) algorithm, which only marginally outperformed the other algorithms (by an average of roughly 0.001), consistently achieved higher rankings, leading to statistically significant differences in its performance across many of the feature sets.

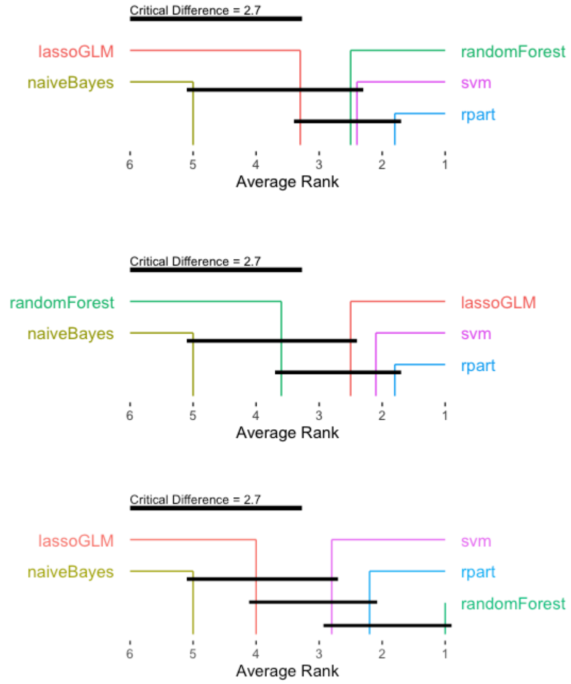

Figure 1. Critical Difference (CD) diagrams demonstrating the results of the Nemenyi post-hoc test for *week-only* features (top), *summed features* (middle), and *appended features* (bottom) with significance level $\alpha = 0.05$. Note that *rpart* refers to the classification tree algorithm. The performance of two classifiers is significantly different if the corresponding average ranks differ by at least the critical difference, calculated according to equation (2) above. Groups of classifiers that are not significantly different are connected by a black CD line. If an algorithm is within one CD of all other algorithms (such as naïve Bayes in the week-only diagram), the correct interpretation is that *the experimental data is not sufficient to reach any conclusion regarding this algorithm*.

Feature	Model	IT	HPE	FSF	IF	ITI	Avg. Rank
Week-only	GLM	0.815 (4)	0.868 (4)	0.844 (4)	0.849 (2)	0.87 (2.5)	**3.3**
	SVM	0.818 (3)	0.871 (1.5)	0.847 (3)	0.849 (2)	0.87 (2.5)	**2.4**
	NB	0.264 (5)	0.285 (5)	0.32 (5)	0.249 (5)	0.226 (5)	**5**
	CART	0.835 (1)	0.871 (1.5)	0.861 (2)	0.849 (2)	0.87 (2.5)	**1.8**
	RF	0.835 (2)	0.871 (3)	0.861 (1)	0.848 (4)	0.87 (2.5)	**2.5**
Summed	GLM	0.818 (2)	0.871 (2)	0.847 (3)	0.849 (1.5)	0.87 (4)	**2.5**
	SVM	0.818 (2)	0.871 (2)	0.847 (1.5)	0.848 (3)	0.87 (2)	**2.1**
	NB	0.247 (5)	0.341 (5)	0.306 (5)	0.255 (5)	0.273 (5)	**5**
	CART	0.818 (2)	0.871 (2)	0.847 (1.5)	0.849 (1.5)	0.87 (2)	**1.8**
	RF	0.818 (4)	0.871 (4)	0.846 (4)	0.848 (4)	0.87 (2)	**3.6**
Appended	GLM	0.81 (4)	0.861 (4)	0.84 (4)	0.844 (4)	0.867 (4)	**4**
	SVM	0.849 (3)	0.893 (2)	0.881 (3)	0.874 (3)	0.891 (3)	**2.8**
	NB	0.252 (5)	0.334 (5)	0.301 (5)	0.256 (5)	0.256 (5)	**5**
	CART	0.855 (2)	0.892 (3)	0.885 (2)	0.874 (2)	0.892 (2)	**2.2**
	RF	0.857 (1)	0.895 (1)	0.886 (1)	0.878 (1)	0.894 (1)	**1**

Table 3. Detailed models results (accuracy; ranks shown in parentheses). Course codes: IT = Introduction to Thermodynamics; HPE = Instructional Methods in Health Practitioners Education; FSF = Fantasy and Science Fiction; IF = Introduction to Finance; ITI = Inside the Internet. Average ranks within a course/feature cell (a single column within a feature, e.g. 2.5 for the ITI course and *week-only* features) are the result of ties.

CONCLUSIONS AND FUTURE RESEARCH

In this work, we contribute to the growing field of predictive models in student success by (1) summarizing the state of the practice when it comes to comparing machine learned models and (2) demonstrating such comparisons using large scaled learning data from a diverse set of MOOCs.

While this project demonstrates a basic implementation of a statistical approach for evaluating predictive model performance in MOOCs, it also reveals the limits of contemporary approaches and the need for research in several directions. Further research is needed on methods for statistical comparison of different feature extraction approaches from raw datasets, a task that is beyond the scope of this study. Testing procedures and inferential methods used for model comparison do not appear to be appropriate for these cases, which violate the dataset independence assumption underlying the Friedman test and many other tools for statistical comparison. Evaluating feature extraction methods from raw data is a key task in learning analytics and educational data mining, which often builds models on data derived from an unstructured source (such as clickstreams or other traces of learner activity), and such research is essential to valid, reproducible research on the application of such methods to MOOC datasets.

Additional research is also needed to extend this approach to the practical case of testing multiple hyperparameter settings of a given model. While we explicitly only used pre-selected hyperparameters for the models used in this analysis to avoid the confounding effects of additional comparisons, most statistical software packages optimize models by testing a grid of hyperparameter settings, which amounts to several additional layers of comparisons, substantially increasing the chance of observing spurious differences in model performance even when no true difference exists. Identifying approaches which can consider even large grids of hyperparameters to match the reality of current available toolkits will be an important next step for research in this area.

While machine learning algorithms are increasingly made available to researchers in open-source projects, tools for conducting statistical testing and evaluation of these models are far less common, and working within existing toolkits presented several challenges to the research team on this project. Future development efforts in this direction, such as providing built-in statistical testing of model comparison using the Friedman/Nemenyi procedure (and others) and visualization of the results in CD diagrams, would significantly advance researchers' ability to implement these practices in their work.

REFERENCES

1. Janez Demšar. 2006. Statistical Comparisons of Classifiers over Multiple Data Sets. *J. Mach. Learn. Res.* 7, Jan (2006), 1–30.
2. T G Dietterich. 1998. Approximate Statistical Tests for Comparing Supervised Classification Learning Algorithms. *Neural Comput.* 10, 7 (15 Sept. 1998), 1895–1923.
3. Milton Friedman. 1940. A comparison of alternative tests of significance for the problem of m rankings. *Ann. Math. Stat.* 11, 1 (1940), 86–92.
4. Josh Gardner and Christopher Brooks. Statistical Approaches to the Model Comparison Task. In *LAK 2017 Workshop on Methodology in Learning Analytics (MLA)* (in submission).
5. Gjorgji Madjarov, Dragi Kocev, Dejan Gjorgjevikj, and Sašo Džeroski. 2012. An extensive experimental comparison of methods for multi-label learning. *Pattern Recognit.* 45, 9 (2012), 3084–3104.
6. Luis Torgo. 2014. An Infra-Structure for Performance Estimation and Experimental Comparison of Predictive Models in R. (1 Dec. 2014).
7. Wanli Xing, Xin Chen, Jared Stein, and Michael Marcinkowski. 2016. Temporal predication of dropouts in MOOCs: Reaching the low hanging fruit through stacking generalization. *Comput. Human Behav.* 58 (2016), 119–129.