Warming up the Cold Start: Adaptive Step Size Method for the Urnings Algorithm

Gergely, B.; van der Maas, H.L.J.; Maris, G.K.J.; Bolsinova, M.

DOI
10.1007/978-3-031-36336-8_64

Publication date
2023

Document Version
Final published version

Published in
Artificial Intelligence in Education

Warming up the Cold Start: Adaptive Step Size Method for the Urnings Algorithm

Bence Gergely[1,2,3(✉)], Han L.J. van der Maas[4], Gunter K. J. Maris[5], and Maria Bolsinova[6]

[1] Eötvös Lóránt University, Doctorate School of Psychology, Budapest, Hungary
[2] Eötvös Lóránt University, Institute of Psychology, Budapest, Hungary
[3] Károli University of the Reformed Church, Budapest, Hungary
gergely.bence@kre.hu
[4] University of Amsterdam, Amsterdam, The Netherlands
[5] TATA Consultancy Services, Zaventem, Belgium
[6] University of Tilburg, Tilburg, The Netherlands

Abstract. Adaptive learning systems (ALS) tailor educational material to the level of the users. In ALS ability should be continuously estimated based on the users' responses to adaptively selected practice items. However, the large-scale, adaptive, and dynamic nature of ALS poses challenges for traditional estimation methods. The Urnings algorithm [1] has been recently proposed to address these challenges. However, the original algorithm does not address the cold-start problem which ALS suffer from: Initially, it is difficult to adapt item selection to the users' abilities based on limited available information. We develop a modification of the Urnings algorithm aiming to alleviate the cold-start problem by increasing the step size of the algorithm when a systematic change in the ratings is detected, and decreasing it when the ratings are relatively stable. The results of our simulation studies showed that the modified algorithm moves away from the initial values faster, responds to sudden changes in ability better, and results in overall higher accuracy than the original algorithm.

Keywords: adaptive learning systems · trackers · rating systems · Urnings algorithm · cold-start problem

1 Introduction

The goal of an adaptive learning system (ALS) is to tailor the learning material to the behaviour and needs of its users [7,12]. To achieve this, the performance of the users needs to be monitored to continuously estimate the users' ability and items should be selected with difficulty matching the current ability [3, ?]

One line of methods used in ALS are connected to the Elo rating system (ERS), which despite its practicality, lacks desirible statistical properties [2,4].

N. Wang et al. (Eds.): AIED 2023, CCIS 1831, pp. 409–414, 2023.
https://doi.org/10.1007/978-3-031-36336-8_64

Recently a new rating system called "Urnings" has been introduced to address the limitations of the ERS [1]. Urnings estimates the ability and difficulty (under the Rasch model [11]) in an online fashion by tracking these parameters on the probability scale using ratings defined on a grid of discrete proportions. After every response the ratings are updated based on the difference of the observed and simulated outcome (using the current ratings) of the user-item pair. The ratings can adapt to the changes in the true ability, have a tractable limiting distribution and provide unbiased estimates with known standard errors when there is no change in true values [1].

Initially, there is very little or no information about the users' ability in a rating system, thus it takes a long time to achieve an accurate estimate. In this phase, the predictions and the item recommendations can be sub-optimal, failing to adapt to the user's needs. This issue is referred to as the cold-start problem [9,10,14]. It affects new users and makes them more likely to abandon the system due to the inappropriately selected items, which might be experienced as frustrating, demotivating or tedious [7,8,14].Only a few studies provided solutions for the cold-start problem in ALS applications, all of them focusing on the ERS, and its alternatives like Glicko system [5]. These solutions either tried to decrease the prior uncertainty by predicting better starting values based on the background characteristics of the users [10], or implemented continuous control methods to change the size of the updates [5,7,15].

The Urnings algorithm currently does not address the cold-start problem. Therefore, in this paper, we present a modification of the Urnings capable of changing the size of the updates adaptively by analysing the direction and rate of change of the ratings in order to reach the true ability level faster but maintain low standard errors when the estimate is close to the true value.

2 Adaptive Step Size

Opposing to the ERS which tracks the parameters on the logit scale and uses continuous ratings, the ratings in Urnings algorithm are defined in a discrete grid of proportions $\{\frac{0}{n}; \frac{1}{n}; \frac{2}{n}; ...; \frac{n}{n}\}$, where n is the granularity of the grid and in the original algorithm it is kept constant. After each response, ratings can either change by $\frac{1}{n}$ or they constant. The updates are symmetric, meaning that if the student rating increases, the item rating necessarily decreases. The choice of n is very important: A smaller n allows for moving quickly through the parameter space and is better suited for following sudden changes in ability, whereas a large n allows for more precise measurement when the ability is relatively stable.

We build the modified algorithm on the idea that if the ratings are systematically changing in one direction they are likely far from the true ability, whereas if they fluctuate around a constant value (i.e. the chain of rating is stationary), they are likely close to the true ability. Consequently, the objective of an adaptive step size algorithm is to monitor systematic changes in the ratings, and when it occurs increase the step size.

To monitor the rate of change in the ratings, we will consider the chain of differences between the consecutive ratings, which we refer to as the "differential

process."[1] To analyse the change we define a moving window of length $l \in \mathbb{N}$. If the chain of user's ratings is a weak-sense stationary Markov chain (i.e., if the chain has reached its limiting distribution), then the expectation of the differential process in the given time window is 0. If the mean of the differential process is indeed 0, then the step size can decrease to decrease the standard error. If it is not 0, then the chain is not yet in the neighbourhood of the true value and, so the step size should be increased.

One possible method to test whether the expected value of the differential process is different from 0 is using a one-sample permutation test [13]. The permutation test can provide an exact p-value for small window sizes and has no distributional assumptions. If the permutation test is significant, there is evidence that the ratings are systematically changing in one direction.

The permutation test can be applied as follows: First, for small window size (i.e. 10) we create all possible combinations of $\{-1, 1\}$ of length l, for larger l we create these arrays by sampling from this set with replacement. Second, we multiply each of these arrays with the values of the differential process within the given window (i.e., random signs are assigned to these values), and calculate the mean in each permutation. This forms act the null distribution whereas the mean of the differential process is the observed statistic. The p-value is computed by computing the proportions of permutations in which the mean is at least as extreme in the absolute value[2] as the observed mean.

We define a minimum and maximum granularity n_{min} and n_{max}. The algorithm starts from n_{min} (i.e., the largest step size). Then, the step size is modified based on the result of the permutation test. If the permutation test is not significant, n is increased (doubled) after each time window until it reaches n_{max}.[3] Since we change n (the denominator of the rating), we are also doubling the numerator to keep the rating constant. If the permutation test is significant, i.e., when systematic change is detected, we set n back to n_{min}, while the rating's numerator is adjusted accordingly.

3 Methods

Using simulated data we illustrate the cold-start problem in the case of the original Urnings algorithm and show how its modification offers a solution for this problem. Two simulation studies were conducted: 1) with constant user ability; 2) with abilities of all users making a sudden change in the middle of the simulation.

[1] Note, that in this paper we focus only on adapting step size for the users, as there are typically much more responses available per item and item difficulty is less prone to sudden changes which makes the cold-start problem and the problem of following change less important on the item side.

[2] i.e., the two-sided p-value is computed

[3] Note that n_{min} and n_{max} should be chosen in such a way that the $nmin$ can be reached by dividing the n_{max} by a power of 2

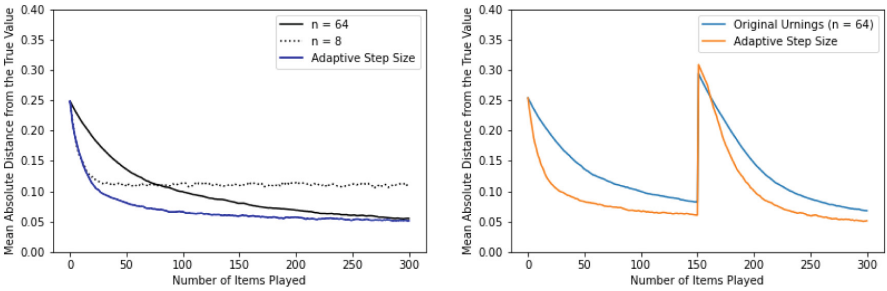
In the first scenario, we simulated an ALS with 300 items and 1500 users, with the true ability and difficulty parameters drawn from $\mathcal{U}(0,1)$. Every user responded to 300 adaptively selected items with the same adaptive mechanism as in [6]. For the original Urnings, we considered $n = 8, 64$ to demonstrate that the bias-variance trade-off of the ratings depends on the $n =$ and how the cold-start problem is affected by it. We expect that while $n = 8$ allows us to move through the parameter space faster (i.e., bias is reduced in the beginning), the total error in the estimation of ability would be higher in the long run because the variance of the ratings is larger than for $n = 64$. The adaptive step size algorithm aims at reducing the bias in the beginning by having a large step size but reducing the variance when the ratings are stabilised and the step size is small. For the modified algorithm we used $n_{min} = 8$ and $n_{max} = 64$ to match the range of the non-adaptive conditions. The step size for the items was equal to $\frac{1}{64}$ in all cases. All user ratings were initialised at 0.5. Item starting values (the numerators of the ratings) were drawn from their limiting distribution [1] which is a binomial distribution with the probability parameter equal to the true value. This mimics a calibrated ALS, as it removes all additional uncertainty of the users' ability coming from the parallel calibration of the items. The window size l was set to 10. The α-level for the permutation test was set to 0.1.

In the second scenario, we examine what happens when true abilities change. The setup of the simulation was the same as before, but after each user responded to 150 items we changed their true value by again drawing from $\mathcal{U}(0,1)$. Here we considered only $n = 64$ for the original algorithm.

We assess the performance of the algorithms by calculating the absolute distance between the estimate and the true value at the given time point for each user and then calculating the mean. This measure (mean absolute difference, MAD) captures the total error in the system. We also look at how fast the MAD reaches the reference value of 0.1, which we consider small enough for the rating to be a good measure of ability (hitting time, HT). We compute the hitting time, defined as the first iteration at which the MAD is at or below 0.1.

4 Results

First, we compare the results of the two cases with fixed n in Fig. 1a. As expected, $n = 8$ allowed for a faster decrease in error, but the MAD stabilises around a higher value based on the last 150 iterations (MAD = 0.11) than the $n = 64$ case (HT = 98, MAD = 0.07). The algorithm with adaptive step size allows for a rapid decrease in the beginning (HT = 27) and at the same time, it was more accurate in the second half of the analysis than the original algorithm (MAD = 0.06). The overall MAD was 0.10, 0.11, and 0.07 for $n = 64$, 8 and the modified algorithm respectively. The modified Urnings algorithm reached the reference level of 0.1 71 iterations faster than the original Urnings. Both the overall error level and the error level of the two halves of the iterations were lower for the modified algorithm than the fixed step size cases.



(a) Simulation with constant true ability (b) Simulation with changing true abilities.

Fig. 1. Mean absolute difference between user ratings and true values over 300 items.

In the second simulation, we presented what happens when the true ability of the students changes with a discrete jump. The modified algorithm (MAD = 0.09, $\text{MAD}_{firsthalf} = 0.10$, $\text{MAD}_{secondhalf} = 0.10$) outperformed the original Urnings (MAD = 0.13, $\text{MAD}_{firsthalf} = 0.13$, $\text{MAD}_{secondhalf} = 0.13$) in terms of total error, both before and after the true values change. Similarly to the first simulation, the modified algorithm showed a steeper decrease in error, than the original Urnings algorithm. The steep decrease in error in the modified case in the first half is due to the large initial step size, the modified algorithm requires some time to detect the change, after which it reduces error compared to the original algorithm.

5 Discussion

In this study, we presented a possible way to implement adaptive step size into the Urnings algorithm, by changing the granularity of the discrete grid of the ratings based on the detection of their systematic change. Our simulation demonstrated that this modified Urnings algorithm can reduce the length of the cold-start problem, meaning that users need to solve fewer items to get close to their true ability. Based on the simulation study the length of the cold-start period with Urnings is about one and a half times longer than with the modified algorithm. The average of the mean absolute difference was approximately one and a half times larger in the original Urnings version.

Further study of the proposed method is needed to investigate how they perform under a wide range of conditions. For example, when different trajectories of ability change are present in the system. Furthermore, the performance of the algorithm in real data needs to be evaluated. However, analysing learning data from a system where item selection was based on ability estimates obtained through a different algorithm than the one that is used can produce biased results since the way items are administered (i.e., which data would be observed) is dependent on the ability estimates (i.e., on the tracking algorithm).

Fine-tuning the parameters of the developed algorithms like the size of the window, the α-level for the permutation test, and minimum and maximum step

size is crucial for successful applications. A way to empirically decide which parameter is suitable for the given problem is yet to be developed [4].

References

1. Bolsinova, M., Maris, G., Hofman, A.D., van der Maas, H.L., Brinkhuis, M.J.: Urnings: a new method for tracking dynamically changing parameters in paired comparison systems. J. Roy. Stat. Soc. Ser. C (Appl. Stat.) (2022)
2. Brinkhuis, M.J., Maris, G.: Dynamic parameter estimation in student monitoring systems. Measurement and Research Department Reports (Rep. No. 2009–1). Arnhem: Cito 146 (2009)
3. Brinkhuis, M.J., Savi, A.O., Hofman, A.D., Coomans, F., van Der Maas, H.L., Maris, G.: Learning as it happens: a decade of analyzing and shaping a large-scale online learning system. J. Learn. Anal. **5**(2), 29–46 (2018)
4. Elo, A.E.: The Rating of Chessplayers, Past and Present. Arco Publications (1978)
5. Glickman, M.E.: Dynamic paired comparison models with stochastic variances. J. Appl. Stat. **28**(6), 673–689 (2001)
6. Hofman, A.D., Brinkhuis, M.J., Bolsinova, M., Klaiber, J., Maris, G., van der Maas, H.L.: Tracking with (un) certainty. J. Intell. **8**(1), 10 (2020)
7. Klinkenberg, S., Straatemeier, M., van der Maas, H.L.: Computer adaptive practice of maths ability using a new item response model for on the fly ability and difficulty estimation. Comput. Educ. **57**(2), 1813–1824 (2011)
8. Ostrow, K.: Motivating learning in the age of the adaptive tutor. In: Conati, C., Heffernan, N., Mitrovic, A., Verdejo, M.F. (eds.) AIED 2015. LNCS (LNAI), vol. 9112, pp. 852–855. Springer, Cham (2015). https://doi.org/10.1007/978-3-319-19773-9_131
9. Pankiewicz, M.: Assessing the cold start problem in adaptive systems. In: Proceedings of the 26th ACM Conference on Innovation and Technology in Computer Science Education, vol. 2, pp. 650–650 (2021)
10. Pliakos, K., Joo, S.H., Park, J.Y., Cornillie, F., Vens, C., Van den Noortgate, W.: Integrating machine learning into item response theory for addressing the cold start problem in adaptive learning systems. Comput. Educ. **137**, 91–103 (2019)
11. Rasch, G.: Studies in mathematical psychology: I. Probabilistic models for some intelligence and attainment tests (1960)
12. Shemshack, A., Spector, J.M.: A systematic literature review of personalized learning terms. Smart Learn. Environ. **7**(1), 1–20 (2020). https://doi.org/10.1186/s40561-020-00140-9
13. Tritchler, D.: On Inverting Permutation Tests. J. Am. Stat. Assoc. **79**(385), 200–207 (1984)
14. Wauters, K., Desmet, P., Van Den Noortgate, W.: Adaptive item-based learning environments based on the item response theory: possibilities and challenges. J. Comput. Assist. Learn. **26**(6), 549–562 (2010)
15. Wauters, K., Desmet, P., Van Noortgate, W.: Monitoring learners' proficiency: weight adaptation in the elo rating system. In: Educational Data Mining 2011 (2010)

[4] The analysis script is hosted at https://github.com/mrpogge/Urnings_AIED.git