Early Detection of Wheel-Spinning in ASSISTments

Yeyu Wang[1][0000−0003−1978−5453], Shimin Kai[2], and Ryan Shaun Baker[3][0000−0002−3051−3232]

[1] University of Wisconsin–Madison, Madison, WI 53706, USA
ywang2466@wisc.edu
[2] Columbia University, New York, NY 10027, USA
[3] University of Pennsylvania, Philadelphia, PA 19104, USA

Abstract. Persistence is a crucial trait for learners. However, a common issue in mastery learning is that persistence is not always productive, a construct termed wheel-spinning. In this paper, we extend on prior work to develop wheel-spinning detectors in the ASSISTments learning system that distinguish between non-persistence, productive persistence and wheel-spinning. To understand how quickly we can detect each state, we use data from different numbers of practice opportunities and compare model performance across student-problem set pairs. We identify that a model constructed using data from the first nine practice opportunities outperforms models using less practice data. However, it is possible to differentiate students who will eventually wheel-spin from learners who will persist productively using data from only the first three opportunities. Wheel-spinning can be differentiated from non-persistence from the first five opportunities, and non-persistence can be differentiated from productive persistence from the first seven opportunities. These results show that early differentiation between wheel-spinning and productive persistence is feasible. These detectors relied upon hint requests, the correctness of prior opportunities, and the amount of practice and time on the skill. Identifying predictive features offer insights into the impact of in-system behaviors on wheel-spinning and guide the system design.

Keywords: Wheel-Spinning · Persistence · Decision Tree · Early Detection · Intelligent Tutoring System.

1 Introduction

1.1 Persistence and Non-Persistence in Learning

Research in recent years has focused on the development of non-cognitive skills to improve student learning, such as resilience and persistence during learning. Persistence is defined as the ability to maintain an action or complete a task regardless of the person's inclination towards the task [5,7]. Recent studies have shown that persistence in educational settings is associated with academic achievement [3,19], creativity [20] and long-term academic outcomes such as later schooling and future earnings [6,19]. However, not all persistence is positive. [2] have argued that some persistence may be unproductive, or wheel-spinning, defined as spending too much time struggling without achieving mastery. The definition of wheel-spinning has varied across different studies and different learning contexts. [2] defined wheel-spinning as not achieving mastery even after attempting 10 or more problems within a problem set; [14] involved two human raters to code wheel-spinning behaviors qualitatively based on a coding manual, with a Cohen's Kappa of 0.9. On the other hand, [10] defined wheel-spinning as attempting more than 10 problems but failing to achieve three consecutive correct responses in a row or demonstrate later retention of the skill.

Non-persistence, or quitting the current learning task without mastering the requisite knowledge, has also been documented in several computer-supported learning environments. For example, in the educational game Physics Playground, non-persistence was defined as quitting the level without successfully solving the problem using the physics knowledge [11,12]. In the learning system ASSISTments, [4] looked at non-persistent behaviors in which students quit the problem set without reaching mastery of a skill, differentiating between quitting immediately and quitting after attempting a few problems. Within the same learning system, [10] defined non-persistence as attempting fewer than ten problems for a skill, but did not consider non-persistence detection in their work.

1.2 Detection of Persistence in Learning

Detection of wheel-spinning behaviors is important in identifying students who may need additional support during a learning task. Because persistence is generally defined by the number of practice opportunities a student has on a learning task, some approaches to modeling or detecting wheel-spinning have been designed to run only after the system has collected student data for a sufficiently large number of practice opportunities. For example, [2], as the first study of wheel-spinning, states that wheel-spinning could be detected as early as the eighth practice opportunity in the ASSISTments system by a logistic regression model. A follow-up study further refined this model and was able to detect wheel-spinning on the seventh practice opportunities [8]. Other machine learning methods such as neural networks [14], gradient boosting [17] and random forest [22] have also been used to enable wheel-spinning detection at earlier stages in practice. Most notably, [4] was able to identify wheel-spinning students at their third opportunity, applying Long Short Term Memory Recurrent Neural Networks. While these studies all take place in an ITS environment, there has also been work on wheel-spinning detection in educational games. [16] constructed a model to detect wheel-spinning based on the features engineered within the first 5 minutes, first 10 minutes, and first 15 minutes of game playing, and [15] constructed a model to differentiate wheel-spinning from productive persistence in a sequence of mathematics games.

In reviewing these prior works, we note that for a wheel-spinning detector to be practical for real-time usage, there are two important criteria to consider. First, a detector should be able to differentiate wheel-spinning from both non-persistence (either successful or non-successful non-persistence) as well as from productive persistence, and should be able to do this at the earliest possible point. With early detection of these states, teachers and system designers may have more opportunities to create interventions to improve the learning experience for students who are at risk of unproductively persisting, or quitting early without completing a learning task. Secondly, predictions based on interpretable models, like decision trees, will offer instructors and system designers more useful insights into the factors influencing persistence and wheel-spinning. Prior work has not yet fully met both of these criteria. Currently, most detectors only account for binary prediction, by either eliminating the non-persistence cases from consideration [17] or treating all cases that are not wheel-spinning as being acceptable [8]. At the same time, recent efforts to improve prediction of wheel-spinning using gradient boosting or neural networks have improved speed and quality of prediction at the cost of interpretability, posing a challenge for educational researchers to uncover and understand the impact of learning behaviors on wheel-spinning.

In this paper, we attempt to address each of these limitations. We 1) construct multi-class detectors distinguishing the three categories discussed above states—non-persistence, productive persistence, and unproductive persistence (wheel-spinning)—so as to capture and compare specific behaviors that differentiate both between persistent vs. non-persistent students, and productively persistent vs. wheel-spinning students; 2) explore the minimum number of practice opportunities that could be used with reasonable accuracy to detect the various persistence states under these conditions, and derive specific features that may be translated into practical interventions. In doing so, in order to compare our results with the previous works on binary wheel-spinning detectors, predicting wheel-spinning vs. non-wheel-spinning [2,4,17], we also build models to make pairwise comparisons for two classes out of the three. In addition, we will summarize the predictive features used across models based on different practice opportunities, to promote better understanding of wheel-spinning. We conclude by discussing the possible impact of the features on persistence and unproductive persistence in learning.

2 Methods

2.1 ASSISTments

ASSISTments is a free online learning platform that provides immediate feedback to students and formative assessment of student performance to teachers [9]. Within the ASSISTments system, Skill Builders are a type of math problem set where students practice randomly generated problems that are based on existing templates and correspond to the same skill [9]. In a Skill Builder, students cannot proceed to the next problem until they submit the correct response. Hints are available to assist them with problem-solving. For each problem, students could make multiple attempts and request multiple hints. In general, there are two to three levels of hint per problem, followed by a bottom-out hint that provides the final answer. Students have to correctly answer three consecutive questions to complete a problem set. They are then given a single-item test after a certain period—usually a week later, though teachers can configure this—with gradually increasing space between reassessments. This test comprises one randomly selected item from a template in the completed problem set, and is delivered through the Automatic Reassessment and Relearning System (ARRS) [21]. The main objective of ARRS is to assess a student's retention of a skill over time. If the student does not answer this item correctly, and therefore fails in skill retention, they will be assigned the corresponding Skill Builder problem set to re-learn the materials.

2.2 Data Collection and Label Generation

Our research dataset is the publicly available ASSISTments Skill Builders data set from the 2014-2015 school year, which consists of 26,522 students who attempted 1,088 Skill Builder problem sets over a year. Each record in the dataset represents a student-problem set pair, which includes the log data when a learner practices a Skill Builder problem set. This data set was chosen due to its use in past research on wheel-spinning and persistence (i.e. [10]). We then constructed eight new datasets: *first-3*, *first-4*, . . . , *first-9*, and *first-10* (*first-1* and *first-2* were not generated, due to not being enough data to infer wheel-spinning in any previous work). Each row in one of these *first-x* datasets shows aggregate data about a student's learning in a certain problem set (i.e., a student-problem set pair), where x is the threshold number of problems over which data is aggregated. For example, *first-3* contains only data about the first 3 problems that the student attempted in each problem set, whereas *first-4*, *first-5* and *first-6* contain data about the first 4, 5 and 6 problems respectively. It should be noted that, given a problem set, a student who attempted only 3 problems would be included in *first-3* but not in *first-4* to *first-10*, while a student who completed 10 problems would be included in every dataset from *first-3* to *first-10*. More generally, the number of student-problem set pairs decreases as x increases, because there are fewer students who attempt more problems.

Table 1. Criteria of non-persistence (NP), productive persistence (PP), and wheel-spinning (WS) in the Skill Builder system.

Definition	Three Correct in a Row (Mastery) on or after the 10th Problem	First ARRS Test	Ten or More Problems
NP	Any	Any	No
PP	Yes	Passed	Yes
WS	No Yes	Any No	Yes

Next, we labeled each row of student-problem set pair as either productive persistence (PP), wheel-spinning (WS) or non-persistence (NP), according to the operational definitions in [10] (Table 1). If a student did fewer than 10 problems in a problem set, the corresponding student-problem set pair is labeled as NP. Otherwise, the pair is labeled as PP if the student reached mastery (i.e., get three correct responses in a row and pass the ARRS test) or WS if she did not.

While our definitions involve the ARRS test, some students were not assigned this test even after getting three correct responses in a row because the teachers turned the ARRS feature off. These instances, which account for 211,612 pairs from the original 287,093 student-problem set pairs, were considered out of scope and removed from further analysis. Of the remaining student-problem set pairs, 6,855 were classified as WS and 2,093 as PP; these pairs are present in every first-x dataset but take on different feature values depending on x. The number of NP pairs in the datasets from *first-3* to *first-10* are 51866, 33197, 26983, 12663, 7833, 4290, 1900 and 0 respectively. As previously noted, there are fewer NP records as x increases; the *first-10* dataset, in particular, has no NP records because students who reached the 10th problem were considered persistent.

2.3 Feature Engineering and Machine Learning

We built upon the feature set developed by [1], which consists of student actions and attributes within the ASSISTments Skill Builder platform that provides information on student persistence and learning. More specifically, we included 25 core features related to student hint usage, number of practice opportunities at a problem set, number of skill opportunities, and time between student actions. As in [10], we calculated the respective sum, minimum, maximum, average and standard deviation values of these core attributes for each student sequence and generated 125 features based on 25 core features. Next, we constructed a set of models to distinguish between NP, PP and WS. Each model is based on one of the *first-x* datasets. This process consists of three main steps:

Data splitting. We performed a student-stratified split of each *first-x* dataset into a train-validate set (90% of students) and a test set (10% of students).

Feature selection. For each value of x, we conducted outer-loop forward feature selection on the train-validate set. This routine starts with an empty feature set and, at each step, selects the feature that would generate the best performance, according to the result of cross-validation. To reduce overfitting, we set the maximum number of features to 20 and imposed an early-stopping condition: if the next candidate feature does not yield a performance improvement of more than 0.001, the routine would stop.

Model evaluation. We built a model based on the features from the previous step, and trained it on the whole train-validate set. Then we evaluated the model on the test set based on macro-average AUC and pairwise AUC between NP-WS, PP-WS, and NP-PP. In this way, we ensured that no data was used for both feature selection and model evaluation, which would bias the results.

In the above steps, our performance metric is 10-fold cross-validated AUC. Due to a class imbalance between WS, NP and PP, we oversampled the training data by randomly adding copies of records from the minority classes. To measure the goodness of the model, we adopted macro-averaging AUC for the multi-class prediction. Finally, to compare to our results with those of [10]'s binary detector that differentiates between wheel-spinning and productive persistent states, we chose the decision tree implementation from [18]. We used entropy as the splitting criterion, set the maximum tree depth as 12 and minimum number of instances per leaf as 2. While these hyperparameters could be individually tuned for each dataset model to potentially yield better performance, our goal is to use the same model construction process across all eight datasets in order to compare their performances as well as the salient features in each, and to avoid the over-fitting associated with hyperparameter tuning.

Among the 125 features, some were computed based on student actions on a certain number of past problems. *Past8BottomOut* and *Past8HelpRequest*, for example, refer to the number of bottom-out hints and help requests made in the past 8 problems. We removed these features from the feature selection process on the datasets where they are not applicable - for this example, the *first-3* to *first-8* datasets, which do not include student-skill data from more than 8 problems.

3 Results

3.1 Feature Selection Results

By applying the forward feature selection algorithm, we identified the feature sets that maximized the model performance for each *first-x* dataset. Among the eight decision tree models, six have root node features which are related to hint usage, such as the mean (*first-3*, *first-7*, *first-8*) and sum (*first-4*, *first-6*) of the total number of hints used, and mean of the bottom-out hint requested in the last eight opportunities (*first-10*). The root nodes of the other two models are time factors, such as sum (*first-5*) and mean (*first-9*) of the duration since the last time the student practiced the skill. While each dataset has its own feature set, we observed that there were features shared across datasets. To better represent this commonality, we summarized all the selected features into seven categories, which include the question type in the problem set, help request behaviors, hint use, scaffolding, opportunity number, amount of practice and time, and the count of failed opportunities. In Table 2, we listed three example feature categories selected with their descriptions [4]. The number list after each feature indicates which *first-x* dataset models it was selected for.

Based on forward selection, all the models from *first-3* to *first-10* include features related to hint requesting behaviors (*HintTotal*). In addition, features related to *HintTotal* are selected for the root node of five models, which indicates that features related to hint requests play a crucial role in predicting WS, NP and PP. Other features, like the number of practice opportunities and amount of time as well as the number of wrong attempts made on the previous problems, are present across different models. We will discuss the implications of these findings in the discussion section.

[4] The full table of features selected for each model can be viewed at https://github.com/yeyuw215/AIED_WS_2020/blob/master/FullTable2.pdf

Table 2. Examples of selected features, categories and descriptions.

Features Categories	Features and Descriptions
Hint	- **HintTotal (3,4,5,6,7,8,9,10)**: The total number of hint requests. - **Past8BottomOut (9,10)**: The number of bottom-out hint requests in the past 8 attempts.
Amount of Practice and Time	- **TimeBetweenProblems (5,7,8,9)**: The duration of time in between problems related to the skill. - **TimeTaken (3,4,5,6)**: The amount of time spent to complete the current problem. - **TotalSkillOpportunities (5,6,7,8,9)**: The total number of problems attempted that are related to the skill in the current problem set. - **TotalTimeOnSkill (3,4)**: The total amount of time spent on the skill in the system.
Wrong Count	- **TotalPastWrongCount (3,4,9)**: The total number of incorrect attempts made on problems within the current problem set. - **TotalPercentPastWrong (4,5)**: The percentage of incorrect attempts made on problems within the current problem set. - **Past5WrongCount (9)**: The number of attempts made that were incorrect in the past 5 attempts.

Table 3. Features selected for each *first-x* dataset model. Root feature denotes the feature at the root node of each decision tree model. *"_m"* indicates the feature is aggregated as mean; *"_s"* indicates the feature is aggregated as sum.

	first-3	**first-4**	**first-5**	**first-6**
# of Features	10	12	11	6
Root Feature	m_HintTotal	s_HintTotal	s_TimeBtwProb	s_HintTotal
	first-7	**first-8**	**first-9**	**first-10**
# of Features	11	8	10	5
Root Feature	m_HintTotal	m_HintTotal	m_TimeBtwProb	m_P8BottomOut

3.2 Model Performance for *"first-x"* Datasets

For all *first-x* datasets, we applied the same feature selection and model evaluation procedure. In order to identify how early we can predict wheel-spinning, we calculated the macro-averaging AUC (for the multiple classes of PP, NP and WS) as goodness measurement and compared the improvement from including more problems, or practice opportunities, into consideration. We also calculated the pairwise AUC for WS-PP, NP-PP, and WS-NP predictions, to understand how well the model can differentiate between specific pairs of states.

When including more and more practice opportunities into consideration, the macro-average AUC scores of the multi-classes detector increases gradually (see Fig. 1). The model including the data for the first 9 opportunities has the best performance, with a macro-averaging AUC of 0.62. For contrasting NP-PP and WS-NP, the AUC shows an increase with more practice opportunity data. For the prediction contrasting NP and PP, including data from the first

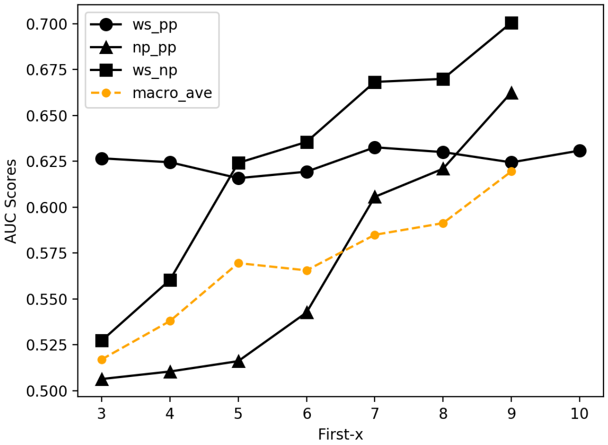

Fig. 1. AUC Scores for different first-x dataset.

7 practice opportunities leads to the largest increase of AUC from 0.54 (*first-6*) to 0.61 (*first-7*). Similarly, the AUC score of contrasting WS and NP increased the most after including data from the first 5 practice opportunities, from 0.56 (*first-4*) to 0.62 (*first-5*). However, the AUC for the WS-PP detector fluctuates around 0.625 and shows no rising trend from first-3 to first-10.

4 Discussion

4.1 Feature Selection Results

We observed that the hint-related features were present in all dataset models as well as at the root node of five models, which indicates these features have the most predictive power. This finding is consistent with previous studies. For instance, [8] identified features involving hints to predict wheel-spinning, such as hint use, count of previous practice opportunities with hint requests and whether students requested at least five hint requests. Another finding in our model is the effect of bottom-out hint requests for predicting WS. The average number of bottom-out hint requests for the past eight practice opportunities is selected as a root node for predicting WS and PP (*first-10*), which indicates that bottom-out hint is a strong predictor for predicting WS against PP. [2] also identified a similar finding: after the 4th practice opportunities, bottom-out hint request is positively associated with wheel-spinning. [10] similarly reported that heavy use of bottom-out hints is associated with wheel-spinning.

Another category of features highly related to wheel-spinning detection is the correctness of previous practice opportunities (*TotalPastWrongCount*, *TotalPercentPastWrong*, and *Past5WrongCount*). This finding is also consistent with previous studies [8,15,22]. In particular, [22] compared wheel-spinning detection across different tutors, algorithms and features. They found that a logistic regression model with only one feature, correct response percentage, achieved less but comparable accuracy with other multi-feature models built using random forest, indicating that correctness is a strong predictor for wheel-spinning prediction. Furthermore, according to [8], the number of previous incorrect responses on the same skill has a positive relationship with wheel-spinning. In a math learning game, [15] also found that prior knowledge measured by missing rate and nonproficiency of skills is highly related to wheel-spinning.

Features related to the amount of practice and time (*TimeTaken*, *TotalTimeOnSkill*, *TotalSkillOpportunities*, *TimeBetweenProblems*) are selected in all the *first-x* models. For models including fewer opportunities to practice the skill (*first-3* to *first-6*), *timetaken* and *totalfrtimeonskill* are predictive of wheel-spinning. However, for the models with more accumulated data (*first-5* to *first-9*), the features switched from time duration (*TotalTimeOnSkill*) to measures of the number of opportunities (*TotalSkillOpportunities*). [2] also found that response time is more predictive on the first several practice opportunities. For the later responses, fast response might indicate either the mastery of skill or gaming the system, which makes the meaning of response time ambiguous.

4.2 Model Performance of Multi-Class and Pairwise Prediction

According to Fig. 1, the performance of the multi-class prediction increases as we include data from more practice opportunities. When including data from the first 9 practice opportunities, the macro-averaging AUC reached 0.62. To our best knowledge, this is the first study exploring the integrated detection of non-persistence, wheel-spinning, and productive persistence together, extending the previous research on WS detector using a decision tree classifier [10]. Therefore, it could be used as a baseline to evaluate model performance in future work.

In differentiating wheel-spinning (WS) from productive persistence (PP), we found that model performance AUC values fluctuate around 0.625 from the first-3 to the first-10 datasets, which implies that our predictive model is stable and able to differentiate between students at-risk of wheel-spinning from students who are productively persistent early on from the third practice opportunity onward. This finding may appear to contradict prior studies that find that models improve with more data [8,14,22]. This difference between studies may be due to the difference in how mastery is defined across the various studies. In prior studies, the criteria of productive mastery is defined based on in-system performance, like three-correct-in-a-row [8]. However, the stricter definition of productive persistence in our study requires students to not only meet the "three-correct-in-a-row" mastery criteria, but also pass the delayed ARRS test to demonstrate learning retention [10]. It is possible that a definition of mastery based on robust learning, a higher bar than simply achieving three correct answers in a row, might be easier to detect early. However, a contrasting finding is obtained by [22], who obtained more accurate prediction and earlier detection when using a more generous criterion of mastery than three-correct-in-a-row.

In our models generated to differentiate between wheel-spinning (WS) and non-persistence (NP), we observed that while model performance increased with the number of practice opportunities, the increase in AUC value is highest between the 4th and 5th practice opportunities. This implies that our detectors may be able to differentiate WS from NP with sufficient accuracy by the 5th practice opportunity. [4] examined the performance of Long-Short Term Memory Networks to predict wheel-spinning and non-persistence on ASSISTments in terms of how many opportunities to practice were provided to the algorithm. They found that the 3rd opportunity might be the earliest timing to predict both WS and NP, an earlier point than seen in our study. Our detectors therefore require more data than [4]. However, we are able to interpret the features in our model based on the decision tree structures to derive more general insights. This tradeoff between model performance and interpretability is also present in other areas of learning analytics such as knowledge component modeling [13].

5 Conclusion

In this study, we explore the potential for early detection of wheel-spinning, productive persistence and non-persistence in ASSISTments. By constructing decision tree models and observing the change of model performance as data about more practice opportunities is aggregated, we found that the model based on nine practice opportunities results in the best performance; the model based on the first three practice opportunities allows early detection of wheel-spinning versus productive persistence, the first five practice opportunities are sufficient for differentiation of wheel-spinning from non-persistence, and the first seven practice opportunities are sufficient for differentiation of productive persistence from non-persistence. Due to the interpretability of decision tree models, we examined the common features across models and the root node features of each. The predictive features, like hint and bottom-out hint usage, correctness and amount of time and opportunities on the previous practice, offer us insights about the factors which might lead to wheel-spinning.

Another potential area for future work, personalized intervention based on which features are predictive could be integrated into the existing learning system to better optimize student learning. Since the features which are predictive of wheel-spinning are at least somewhat consistent across studies and datasets (see discussion above), this may help us to design future intelligent tutoring systems that are more adaptive to the possibility of wheel-spinning in their early stages of learning. Such a system could encourage students to use the bottom-out hints at the first several practice opportunities, if needed; then the system could limit bottom-out hints availability in the later practice opportunities. In this way, the system could leverage what we know about wheel-spinning to help us prevent it.

Acknowledgements. We thank Neil Heffernan and the ASSISTments team, and grant NSF #DRL-1535340.

References

1. Baker, R.S., Goldstein, A.B., Heffernan, N.T.: Detecting learning moment-by-moment. International Journal of Artificial Intelligence in Education **21**(1-2), 5–25 (2011)
2. Beck, J.E., Gong, Y.: Wheel-spinning: Students who fail to master a skill. In: International conference on artificial intelligence in education. pp. 431–440. Springer (2013)
3. Borghans, L., Meijers, H., Ter Weel, B.: The role of noncognitive skills in explaining cognitive test scores. Economic inquiry **46**(1), 2–12 (2008)
4. Botelho, A.F., Varatharaj, A., Patikorn, T., Doherty, D., Adjei, S.A., Beck, J.E.: Developing early detectors of student attrition and wheel spinning using deep learning. IEEE Transactions on Learning Technologies **12**(2), 158–170 (2019)
5. Cloninger, C.R., Svrakic, D.M., Przybeck, T.R.: A psychobiological model of temperament and character. Archives of general psychiatry **50**(12), 975–990 (1993)
6. Deke, J., Haimson, J.: Valuing student competencies: Which ones predict postsecondary educational attainment and earnings, and for whom? final report. Mathematica Policy Research, Inc. (2006)
7. Duckworth, A.L., Peterson, C., Matthews, M.D., Kelly, D.R.: Grit: perseverance and passion for long-term goals. Journal of personality and social psychology **92**(6), 1087 (2007)
8. Gong, Y., Beck, J.E.: Towards detecting wheel-spinning: Future failure in mastery learning. In: Proceedings of the second (2015) ACM conference on learning@ scale. pp. 67–74 (2015)
9. Heffernan, N.T., Heffernan, C.L.: The assistments ecosystem: Building a platform that brings scientists and teachers together for minimally invasive research on human learning and teaching. International Journal of Artificial Intelligence in Education **24**(4), 470–497 (2014)
10. Kai, S., Almeda, M.V., Baker, R.S., Heffernan, C., Heffernan, N.: Decision tree modeling of wheel-spinning and productive persistence in skill builders. JEDM—Journal of Educational Data Mining **10**(1), 36–71 (2018)
11. Karumbaiah, S., Baker, R.S., Barany, A., Shute, V.: Using epistemic networks with automated codes to understand why players quit levels in a learning game. In: International Conference on Quantitative Ethnography. pp. 106–116. Springer (2019)
12. Karumbaiah, S., Baker, R.S., Shute, V.: Predicting quitting in students playing a learning game. International Educational Data Mining Society (2018)
13. Liu, R., McLaughlin, E.A., Koedinger, K.R.: Interpreting model discovery and testing generalization to a new dataset. In: Educational Data Mining 2014. Citeseer (2014)
14. Matsuda, N., Chandrasekaran, S., Stamper, J.C.: How quickly can wheel spinning be detected? In: EDM. pp. 607–608 (2016)
15. Owen, V.E., Roy, M.H., Thai, K., Burnett, V., Jacobs, D., Keylor, E., Baker, R.S.: Detecting wheel-spinning and productive persistence in educational games. International Educational Data Mining Society (2019)
16. Palaoag, T.D., Rodrigo, M.M.T., Andres, J.M.L., Andres, J.M.A.L., Beck, J.E.: Wheel-spinning in a game-based learning environment for physics. In: International Conference on Intelligent Tutoring Systems. pp. 234–239. Springer (2016)
17. Park, S., Matsuda, N.: Predicting students' unproductive failure on intelligent tutors in adaptive online courseware. In: Proceedings of the Sixth Annual GIFT Users Symposium. vol. 6, p. 131. US Army Research Laboratory (2018)

18. 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.: Scikit-learn: Machine learning in Python. Journal of Machine Learning Research **12**, 2825–2830 (2011)
19. Poropat, A.E.: A meta-analysis of the five-factor model of personality and academic performance. Psychological bulletin **135**(2), 322 (2009)
20. Prabhu, V., Sutton, C., Sauser, W.: Creativity and certain personality traits: Understanding the mediating effect of intrinsic motivation. Creativity Research Journal **20**(1), 53–66 (2008)
21. Wang, Y., Heffernan, N.T.: Towards modeling forgetting and relearning in its: Preliminary analysis of arrs data. In: EDM. p. 352 (2011)
22. Zhang, C., Huang, Y., Wang, J., Lu, D., Fang, W., Stamper, J., Fancsali, S., Holstein, K., Aleven, V.: Early detection of wheel spinning: Comparison across tutors, models, features, and operationalizations. Grantee Submission (2019)