Interpretable AI to Understand Early Effective and Ineffective Programming Behaviours from CS1 Learners

Filipe Dwan Pereira
Federal University of Roraima
Boa Vista, Brazil
filipe.dwan@ufrr.br

Elaine H. T. Oliveira
David B. F. Oliveira
Leandro S. G. Carvalho
Federal University of Amazonas
Manaus, Brazil
{elaine,david,galvao}@icomp.ufam.edu.br

Alexandra I. Cristea
Durham University
Durham, UK
alexandra.i.cristea@durham.ac.uk

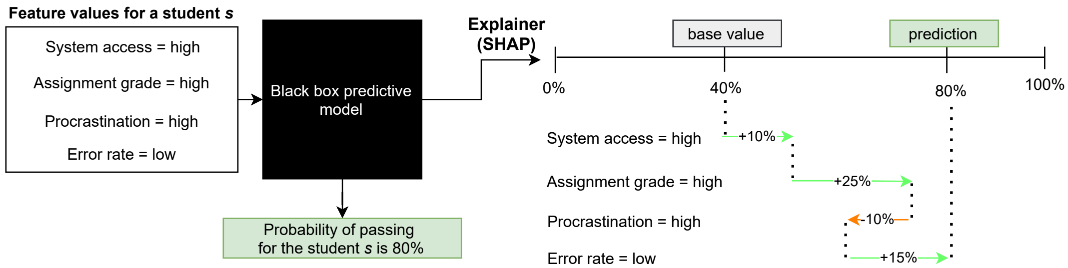

Figure 1: Illustration of how we will explain local predictions from a black box model and features values from the learner.

Many studies [2, 3, 7–10, 14, 18, 21, 25, 29, 31, 32, 35, 36] report high failure rate in CS1 courses. Thus, building predictive models to estimate the learner performance in the beginning of these courses is essential to allow early interventions [1, 4, 7, 11–13, 16, 17, 26, 29, 30, 34]. However, Computing in Education literature [5, 23, 28, 31, 33] notes the lack of studies on early learner behaviours that can be effective or ineffective, that is, programming behaviours that potentially lead to success or failure, respectively. Hence, beyond the prediction, it is crucial to explain what leads the predictive model to make the decisions (e.g., why a given student *s* is classified as 'passed'). Indeed, interpreting why a model makes a certain prediction is essential for stakeholders to verify which student behaviours should be encouraged and which ones need to be improved upon.

Thus, in recent works [22, 24, 27], we extracted useful information from fine-grained log-data collected from CodeBench, an Online Judge (OJ) used at Federal University of Amazonas to support programming classes. Using this data, we then depicted how CS1 students solve the programming problems, how they deal with errors and deadlines, etc. Additionally, we used these programming behaviours (see our previous work [24]) as features in machine learning models to predict the learners performance in the beginning of the course. Still, we compared an optimised Deep Learning (DL) architecture versus a SoA Genetic Algorithm (GA) to find an optimised pipeline to predict the student performance. Moreover, we also fine-tuned a XGBoost model, as this algorithm is recommended to tabular-data [6, 15] as ours. The DL and the XGBoost models achieved cutting-edge results, statistically surpassing the GA approach and SoA related works.

EduComp'21, Abril 27–30, 2021, Jataí, Goiás, Brasil (On-line)

However, the DL and XGBoost models are non-linear techniques that construct black box models. As our goal is to detect and understand early effective and ineffective behaviours, we need to 'open' such a black box to explain the model's decision. To do so, in this work in progress, we use a state-of-the-art unified approach to interpret black-box model predictions [19], which uses SHapley Additive exPlanations (SHAP) method [20]. SHAP method can be used to explain linearly a complex model (e.g. DL or XGboost) in instance level. In our context of CS1 performance prediction, this method gets the predictive model and the features values for a given student as input and the possibility of explanation of which feature values are increasing or decreasing the learner chances of passing as output. That is, using SHAP we can identify early effective and ineffective behaviours in student-level granularity. More than that, using this local explanation as building blocks, we can also extract global data insight and give a summarisation of the model. Thus, our analysis can be useful to support learners (e.g. via formative feedback) and instructors decisions, as a move towards an human/AI OJ system to support CS1 classes.

In Figure 1 we show an example of how we will use the SHAP values to give explanation of an individual prediction. In this hypothetical case, the learner has a high value of *system access*, *grade in the assignment*, and *procrastination*, whilst he/she has a low value of *error rate* when solving the problems from the assignment. Using these values as input, the predictive model estimates that the learner's probability of passing is 80%. The explanation is given by showing the cumulative change in percentage values across features over the base value. The model's base value is the average prediction over the training set, the value that would be predicted if there was no prior information – there are no features for the current instance. Thus, in this hypothetical case, the system access of the student increased 10% her/his chances of passing, her/his assignment grade increased 25%, whereas the procrastination, which is high, decreased her/his chances of passing in 10%. Finally, the error rate value (low) increased her/his chances by 15%.

ACKNOWLEDGMENTS

This research, carried out within the scope of the Samsung-UFAM Project for Education and Research (SUPER), according to Article 48 of Decree nº 6.008/2006 (SUFRAMA), was partially funded by Samsung Electronics of Amazonia Ltda., under the terms of Federal Law nº 8.387/1991, through agreements 001/2020 and 003/2019, signed with Federal University of Amazonas and FAEPI, Brazil. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001 and the Conselho Nacional de Desenvolvimento Científico e Tecnológico - Brasil (CNPq grant 308513/2020-7).

REFERENCES

[1] Gene Charles Lima Aguiar and Filipe Dwan Pereira. 2018. *Uma Abordagem Data-Driven Para Predição Precoce Da Evasão Em Turmas De Programação Que Utilizam Juízes Online.* Graduação em Ciência da Computação, Universidade Federal de Roraima, Boa Vista. Universidade Federal de Roraima, Boa Vista.

[2] Ada Araujo, Daniel Lopes Zordan Filho, Elaine Harada Teixeira de Oliveira, Leandro Silva Galvão de Carvalho, Filipe Dwan Pereira, and David Braga Fernandes de Oliveira. 2021. Mapeamento e análise empírica de misconceptions comuns em avaliações de introdução à programação. In *Anais do Simpósio Brasileiro de Educação em Computação.* SBC, 123–131.

[3] Jens Bennedsen and Michael E Caspersen. 2019. Failure rates in introductory programming: 12 years later. *ACM inroads* 10, 2 (2019), 30–36.

[4] Ryan Bockmon, Stephen Cooper, Jonathan Gratch, Jian Zhang, and Mohsen Dorodchi. 2020. Can Students' Spatial Skills Predict Their Programming Abilities? *(ITiCSE '20).* Association for Computing Machinery, New York, NY, USA, 446–451.

[5] Adam Carter, Christopher Hundhausen, and Daniel Olivares. 2019. Leveraging the IDE for learning analytics. *The Cambridge Handbook of Computing Education Research, Sally A. Fincher and Anthony V. Robins (Eds.). Cambridge University Press, Cambridge* (2019), 679–706.

[6] Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining.* 785–794.

[7] Evandro B Costa, Baldoino Fonseca, Marcelo Almeida Santana, Fabrísia Ferreira de Araújo, and Joilson Rego. 2017. Evaluating the effectiveness of educational data mining techniques for early prediction of students' academic failure in introductory programming courses. *Computers in Human Behavior* 73 (2017), 247–256.

[8] Hermino Barbosa de Freitas Júnior, Filipe Dwan Pereira, Elaine Harada Teixeira de Oliveira, David Braga Fernandes de Oliveira, and Leandro Silva Galvão de Carvalho. 2020. Recomendação Automática de Problemas em Juízes Online Usando Processamento de Linguagem Natural e Análise Dirigida aos Dados. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 1152–1161.

[9] Marcos Avner Pimenta de Lima Lima, Leandro Silva Galvão de Carvalho, Elaine Harada Teixeira de Oliveira, David Braga Fernandes de Oliveira, and Filipe Dwan Pereira. 2021. Uso de atributos de código para classificação da facilidade de questões de codificação. In *Anais do Simpósio Brasileiro de Educação em Computação.* SBC, 113–122.

[10] Joseph de Oliveira, Felipe Salem, Elaine Harada Teixeira de Oliveira, David Braga Fernandes Oliveira, Leandro Silva Galvão de Carvalho, and Filipe Dwan Pereira. 2020. Os estudantes leem as mensagens de feedback estendido exibidas em juízes online?. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 1723–1732.

[11] Ingrid Lima dos Santos, David Braga Fernandes Oliveira, Leandro Silva Galvão de Carvalho, Filipe Dwan Pereira, and Elaine Harada Teixeira de Oliveira. 2020. Tempos de Transição em Estados de Corretude e Erro como Indicadores de Desempenho em Juízes Online. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 1283–1292.

[12] Filipe Dwan, Elaine Oliveira, and David Fernandes. 2017. Predição de zona de aprendizagem de alunos de introdução à programação em ambientes de correção automática de código. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 28. 1507.

[13] Samuel Fonseca, Elaine Oliveira, Filipe Pereira, David Fernandes, and Leandro Silva Galvão de Carvalho. 2019. Adaptação de um método preditivo para inferir o desempenho de alunos de programação. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 30. 1651.

[14] Samuel C Fonseca, Filipe Dwan Pereira, Elaine HT Oliveira, David BF Oliveira, Leandro SG Carvalho, and Alexandra I Cristea. 2020. Automatic Subject-based Contextualisation of Programming Assignment Lists. EDM.

[15] A. Géron. 2019. *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.* O'Reilly Media.

[16] Arto Hellas, Petri Ihantola, Andrew Petersen, Vangel V. Ajanovski, Mirela Gutica, Timo Hynninen, Antti Knutas, Juho Leinonen, Chris Messom, and Soohyun Nam Liao. 2018. Predicting Academic Performance: A Systematic Literature Review. In *Proceedings Companion of the 23rd Annual ACM Conference on Innovation and Technology in Computer Science Education* (Larnaca, Cyprus) *(ITiCSE 2018 Companion).* Association for Computing Machinery, New York, NY, USA, 175–199.

[17] A. Kumar Veerasamy, D. D'Souza, M. V. Apiola, M. J. Laakso, and T. Salakoski. 2020. Using early assessment performance as early warning signs to identify at-risk students in programming courses. In *2020 IEEE Frontiers in Education Conference (FIE).* 1–9. https://doi.org/10.1109/FIE44824.2020.9274277

[18] Marcos Lima, Leandro Silva Galvão de Carvalho, Elaine Harada Teixeira de Oliveira, David Braga Fernandes Oliveira, and Filipe Dwan Pereira. 2020. Classificação de dificuldade de questões de programação com base em métricas de código. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 1323–1332.

[19] Scott M Lundberg, Gabriel Erion, Hugh Chen, Alex DeGrave, Jordan M Prutkin, Bala Nair, Ronit Katz, Jonathan Himmelfarb, Nisha Bansal, and Su-In Lee. 2020. From local explanations to global understanding with explainable AI for trees. *Nature machine intelligence* 2, 1 (2020), 56–67.

[20] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems.* 4765–4774.

[21] Andersen Per-Arne, Christian Kråkevik, M. Goodwin, and A. Yazidi. 2016. Adaptive task assignment in online learning environments. *ACM. Proceedings of the 6th International Conference on Web Intelligence, Mining and Semantics* (2016), 5.

[22] Filipe Pereira, Elaine Oliveira, David Fernandes, Leandro Silva G. Carvalho de Carvalho, and Hermino Junior. 2019. Otimização e automação da predição precoce do desempenho de alunos que utilizam juízes online: uma abordagem com algoritmo genético. In *Brazilian Symposium on Computers in Education (Simpósio Brasileiro de Informática na Educação-SBIE)*, Vol. 30. 1451.

[23] Filipe Dwan Pereira, Linnik Maciel de Souza, Elaine Harada Teixeira de Oliveira, David Braga Fernandes de Oliveira, and Leandro Silva Galvão de Carvalho. 2020. Predição de desempenho em ambientes computacionais para turmas de programação: um Mapeamento Sistemático da Literatura. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 1673–1682.

[24] Filipe Dwan Pereira, Samuel C Fonseca, Elaine HT Oliveira, David BF Oliveira, Alexandra I Cristea, and Leandro SG Carvalho. 2020. Deep learning for early performance prediction of introductory programming students: a comparative and explanatory study. *Brazilian journal of computers in education.* 28 (2020), 723–749.

[25] Filipe D Pereira, Hermino Junior, Luiz Rodriguez, Armando Toda, Elaine HT Oliveira, Alexandra I Cristea, David Oliveira, Leandro Carvalho, Samuel Fonseca, Ahmed Alamri, and Seiji Isotani. 2021. A recommender system based on effort: towards minimising negative affects and maximising achievement in CS1 learning. In *International Conference on Intelligent Tutoring Systems*. Springer.

[26] Filipe D Pereira, Elaine Oliveira, Alexandra Cristea, David Fernandes, Luciano Silva, Gene Aguiar, Ahmed Alamri, and Mohammad Alshehri. 2019. Early dropout prediction for programming courses supported by online judges. In *International Conference on Artificial Intelligence in Education*. Springer, 67–72.

[27] F. D. Pereira, E. H. Oliveira, D. Fernandes, and A. Cristea. 2019. Early performance prediction for CS1 course students using a combination of machine learning and an evolutionary algorithm. In *2019 IEEE 19th International Conference on Advanced Learning Technologies (ICALT)*, Vol. 2161. IEEE, 183–184.

[28] Filipe D Pereira, Elaine HT Oliveira, David BF Oliveira, Alexandra I Cristea, Leandro SG Carvalho, Samuel C Fonseca, Armando Toda, and Seiji Isotani. 2020. Using learning analytics in the Amazonas: understanding students' behaviour in introductory programming. *British Journal of Educational Technology* (2020).

[29] Filipe Dwan Pereira, Elaine H T Oliveira, and David F B Oliveira. 2018. *Uso de um método preditivo para inferir a zona de aprendizagem de alunos de programação em um ambiente de correção automática de código.* Mestrado em Informática. Universidade Federal do Amazonas, Manaus.

[30] Filipe D Pereira, Armando Toda, Elaine HT Oliveira, Alexandra I Cristea, Seiji Isotani, Dion Laranjeira, Adriano Almeida, and Jonas Mendonça. 2020. Can We Use Gamification to Predict Students' Performance? A Case Study Supported by an Online Judge. In *International Conference on Intelligent Tutoring Systems.* Springer, 259–269.

[31] Keith Quille and Susan Bergin. 2019. CS1: how will they do? How can we help? A decade of research and practice. *Computer Science Education* 29, 2-3 (2019), 254–282.

[32] Anthony Robins. 2010. Learning edge momentum: A new account of outcomes in CS1. *Computer Science Education* 20, 1 (2010), 37–71.

[33] A. V. Robins. 2019. Novice programmers and introductory programming. In *The Cambridge Handbook of Computing Education Research.* Cambridge University Press, Cambridge, Chapter 12, 327–376.

[34] Cristóbal Romero and Sebastian Ventura. 2019. Guest editorial: special issue on early prediction and supporting of learning performance. *IEEE Transactions on Learning Technologies* 12, 2 (2019), 145–147.

[35] Christopher Watson and Frederick WB Li. 2014. Failure rates in introductory programming revisited. In *Proceedings of the 2014 conference on Innovation & technology in computer science education.* 39–44.

[36] Daniel Lopes Zordan Filho, Elaine Harada Teixeira de Oliveira, Leandro Silva Galvão de Carvalho, Marcela Pessoa, Filipe Dwan Pereira, and David Braga Fernandes de Oliveira. 2020. Uma análise orientada a dados para avaliar o impacto da gamificação de um juiz on-line no desempenho de estudantes. In *Anais do XXXI Simpósio Brasileiro de Informática na Educação.* SBC, 491–500.