

Bibliography

- [1] Abbeel, P., Ng, A. Y., & Computer Science Department, Stanford University, Stanford, CA 94305, USA. (2004). Apprenticeship Learning via Inverse Reinforcement Learning. In *Proceedings of the 21st International Conference on Machine Learning*. <https://ai.stanford.edu/~ang/papers/icml04-apprentice.pdf>
- [2] Cedazo, R., Cena, C. E. G., & Al-Hadithi, B. M. (2015). A friendly online C compiler to improve programming skills based on student self-assessment. *Computer Applications in Engineering Education*, 23(6), 887–896. <https://doi.org/10.1002/cae.21660>
- [3] Das, R., Ahmed, U. Z., Karkare, A., & Gulwani, S. (2016). Prutor: A System for Tutoring CS1 and Collecting Student Programs for Analysis. *arXiv (Cornell University)*. <https://doi.org/10.48550/arxiv.1608.03828>
- [4] Gupta, R., Pal, S., Kanade, A., & Shevade, S. (2017). DeepFix: Fixing Common C Language Errors by Deep Learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31(1). <https://doi.org/10.1609/aaai.v31i1.10742>
- [5] Rivers, K., & Koedinger, K. R. (2015). Data-Driven Hint Generation in Vast Solution Spaces: a Self-Improving Python Programming Tutor. *International Journal of Artificial Intelligence in Education*, 27(1), 37–64. <https://doi.org/10.1007/s40593-015-0070-z>
- [6] Nguyen, A., Piech, C., Huang, J., & Guibas, L. (2014). Codewebs: scalable homework search for massive open online programming courses. In *WWW '14: Proceedings of the 23rd international conference on World wide web*. ACM Digital Library. <https://doi.org/10.1145/2566486.2568023>
- [7] Liao, S. N., Zingaro, D., Laurenzano, M. A., Griswold, W. G., & Porter, L. (2016). Lightweight, Early Identification of At-Risk CS1 Students. In *ICER '16: Proceedings of the 2016 ACM Conference on International Computing Education Research* (pp. 123–131). ACM Digital Library. <https://doi.org/10.1145/2960310.2960315>
- [8] Robins, A. V. (2019). Novice Programmers and Introductory Programming. In *Cambridge University Press eBooks* (pp. 327–376). <https://doi.org/10.1017/9781108654555.013>
- [9] Piech, C., Sahami, M., Koller, D., Cooper, S., & Blikstein, P. (2012). Modeling how students learn to program. *Proceedings of the 53rd ACM Technical*

- Symposium on Computer Science Education*, 153–160.
<https://doi.org/10.1145/2157136.2157182>
- [10] Sukanya, N. S., & Albert, N. D. D. W. (2023). A Novel Approach to Predict Students Performance through Machine Learning. *International Journal of Engineering Technology and Management Sciences*, 7(5), 278–283.
<https://doi.org/10.46647/ijetms.2023.v07i05.032>
- [11] Alhothali, A., Albsisi, M., Assalahi, H., & Aldosemani, T. (2022). Predicting Student Outcomes in Online Courses Using Machine Learning Techniques: A Review. *Sustainability*, 14(10), 6199. <https://doi.org/10.3390/su14106199>
- [12] Devi, S., Selvam, K., & Rajagopalan, S. (2011). An abstract to calculate big O factors of time and space complexity of machine code. In *International Conference on Sustainable Energy and Intelligent Systems (SEISCON 2011)* (pp. 844–847).
<https://doi.org/10.1049/cp.2011.0483>
- [13] Liao, S. N., Zingaro, D., Thai, K., Alvarado, C., Griswold, W. G., & Porter, L. (2019). A Robust Machine Learning Technique to Predict Low-performing Students. *ACM Transactions on Computing Education*, 19(3), 1–19.
<https://doi.org/10.1145/3277569>
- [14] Kim, B., Heo, S., Lee, J., Jeong, S., Lee, Y., & Kim, H. (2020). Compiler-Assisted Semantic-Aware Encryption for Efficient and Secure Serverless Computing. *IEEE Internet of Things Journal*, 8(7), 5645–5656.
<https://doi.org/10.1109/jiot.2020.3031550>
- [15] Murphy, C., Kaiser, G., Loveland, K., & Hasan, S. (2009). Retina. *ACM SIGCSE Bulletin*, 41(1), 178–182. <https://doi.org/10.1145/1539024.1508929>
- [16] Li, M., Liu, Y., Liu, X., Sun, Q., You, X., Yang, H., Luan, Z., Gan, L., Yang, G., & Qian, D. (2020). The Deep Learning Compiler: A Comprehensive Survey. *IEEE Transactions on Parallel and Distributed Systems*, 32(3), 708–727.
<https://doi.org/10.1109/tpds.2020.3030548>
- [17] Restrepo-Calle, F., Ramírez-Echeverry, J. J., & Gonzalez, F. A. (2018). UNCODE: Interactive System For Learning And Automatic Evaluation Of Computer Programming Skills. *Edulearn Proceedings*, 1, 6888–6898.
<https://doi.org/10.21125/edulearn.2018.1632>

- [18] Ala-Mutka, K. M. (2005). A Survey of Automated Assessment Approaches for Programming Assignments. *Computer Science Education*, 15(2), 83–102. <https://doi.org/10.1080/08993400500150747>
- [19] Ihantola, P., Vihavainen, A., Ahadi, A., Butler, M., Börstler, J., Edwards, S. H., Isohanni, E., Korhonen, A., Petersen, A., Rivers, K., Rubio, M. Á., Sheard, J., Skupas, B., Spacco, J., Szabo, C., & Toll, D. (2015). Educational Data Mining and Learning Analytics in Programming. In *ITICSE-WGR '15: Proceedings of the 2015 ITiCSE on Working Group Reports*. ACM Digital Library. <https://doi.org/10.1145/2858796.2858798>
- [20] Watson, C., Li, F. W., & Godwin, J. L. (2013). Predicting Performance in an Introductory Programming Course by Logging and Analyzing Student Programming Behavior. In *2013 IEEE 13th International Conference on Advanced Learning Technologies* (pp. 319–323). IEEE Xplore. <https://doi.org/10.1109/icalt.2013.99>
- [21] Denny, P., Luxton-Reilly, A., & Tempero, E. (2012). All syntax errors are not equal. In *ITiCSE '12: Proceedings of the 17th ACM annual conference on Innovation and technology in computer science education*. ACM Digital Library. <https://doi.org/10.1145/2325296.2325318>
- [22] Bilegjargal, D., & Hsueh, N. (2021). Understanding Students' Acceptance of Online Judge System in Programming Courses: A Structural Equation Modeling Approach. *IEEE Access*, 9, 152606–152615. <https://doi.org/10.1109/access.2021.3126896>
- [23] Ghosh, A., Malva, L., & Singla, A. (2024). Analyzing-Evaluating-Creating: Assessing Computational Thinking and Problem Solving in Visual Programming Domains. In *SIGCSE 2024: Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 1* (pp. 387–393). <https://doi.org/10.1145/3626252.3630778>
- [24] Lin, P., Cheng, P., & Chen, L. (2019). Using Selective Syntactic Online Compiler to Promote Programming Learning. In *2019 IEEE International Conference on Engineering, Technology and Education (TALE)* (pp. 1–5). IEEE Xplore. <https://doi.org/10.1109/tale48000.2019.9226021>

- [25] Yadav, S. K., & Pal, S. (2012, March 17). *Data Mining: A Prediction for Performance Improvement of Engineering Students using Classification*. arXiv.org. <http://arxiv.org/abs/1203.3832>
- [26] Shen, G., Yang, S., Huang, Z., Yu, Y., & Li, X. (2022). The prediction of programming performance using student profiles. *Education and Information Technologies*, 28(1), 725–740. <https://doi.org/10.1007/s10639-022-11146-w>
- [27] Marjan, M. A., Uddin, M. P., & Afjal, M. I. (2021). An Educational Data Mining System For Predicting And Enhancing Tertiary Students' Programming Skill. *The Computer Journal*, 66(5), 1083–1101. <https://doi.org/10.1093/comjnl/bxab214>
- [28] Sharma, P., & Harkishan, M. (2022). Designing an intelligent tutoring system for computer programming in the Pacific. *Education and Information Technologies*, 27(5), 6197–6209. <https://doi.org/10.1007/s10639-021-10882-9>
- [29] Modi, K., Singh, I., & Kumar, Y. (2024). *Predicting asthma control test score using machine learning regression models*. In *CRC Press eBooks* (pp. 190–197). <https://doi.org/10.1201/9781003466383-29>
- [30] Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/a:1010933404324>
- [31] Buenaño-Fernández, D., Gil, D., & Luján-Mora, S. (2019). Application of Machine Learning in Predicting Performance for Computer Engineering Students: A Case Study. *Sustainability*, 11(10), 2833. <https://doi.org/10.3390/su11102833>
- [32] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. In *KDD '16: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 785–794). ACM Digital Library. <https://doi.org/10.1145/2939672.2939785>
- [33] Altabrawee, H., Ali, O. a. J., & Ajmi, S. Q. (2019). Predicting Students' Performance Using Machine Learning Techniques. *Journal of University of Babylon for Pure and Applied Sciences*, 27(1), 194–205. <https://doi.org/10.29196/jubpas.v27i1.2108>
- [34] Wang, L., Sy, A., Liu, L., & Piech, C. (2017). Deep Knowledge Tracing On Programming Exercises. In *L@S '17: Proceedings of the Fourth (2017) ACM Conference on Learning @ Scale* (pp. 201–204). ACM Digital Library. <https://doi.org/10.1145/3051457.3053985>

- [35] Cabo, C. (2021). Use of Machine Learning to Identify Predictors of Student Performance in Writing Viable Computer Programs with Repetition Loops and Methods. *2021 IEEE Frontiers in Education Conference (FIE)*, 1–9. <https://doi.org/10.1109/fie49875.2021.9637302>
- [36] Alsariera, Y. A., Baashar, Y., Alkaws, G., Mustafa, A., Alkahtani, A. A., & Ali, N. (2022). Assessment and Evaluation of Different Machine Learning Algorithms for Predicting Student Performance. *Computational Intelligence and Neuroscience*, 2022, 1–11. <https://doi.org/10.1155/2022/4151487>
- [37] Rokach, L., & Maimon, O. (2006). Decision Trees. In *Springer eBooks* (pp. 165–192). https://doi.org/10.1007/0-387-25465-x_9
- [38] Gupta, R., Kanade, A., & Shevade, S. (2019). Deep Reinforcement Learning for Syntactic Error Repair in Student Programs. *Proceedings of the AAAI Conference on Artificial Intelligence*, 33(01), 930–937. <https://doi.org/10.1609/aaai.v33i01.3301930>
- [39] Pires, J. P. J., Correia, F. B., Gomes, A., Borges, A. R., & Bernardino, J. (2024). Predicting Student Performance in Introductory Programming Courses. *Computers*, 13(9), 219. <https://doi.org/10.3390/computers13090219>
- [40] Durak, A., & Bulut, V. (2023). Classification and prediction-based machine learning algorithms to predict students' low and high programming performance. *Computer Applications in Engineering Education*, 32(1). <https://doi.org/10.1002/cae.22679>
- [41] Jokhan, A., Chand, A. A., Singh, V., & Mamun, K. A. (2022). Increased Digital Resource Consumption in Higher Educational Institutions and the Artificial Intelligence Role in Informing Decisions Related to Student Performance. *Sustainability*, 14(4), 2377. <https://doi.org/10.3390/su14042377>
- [42] Parihar, S., Dadachanji, Z., Singh, P. K., Das, R., Karkare, A., & Bhattacharya, A. (2017). Automatic Grading and Feedback using Program Repair for Introductory Programming Courses. In *ITiCSE '17: Proceedings of the 2017 ACM Conference on Innovation and Technology in Computer Science Education*. ACM Digital Library. <https://doi.org/10.1145/3059009.3059026>
- [43] Jayasree, S. J. S., & Asim, M. a. M. (2021). A Novel Approach to Predict Students Performance in Online Courses through Machine Learning. *International Journal*

- of Engineering Technology and Management Sciences*, 19–22.
<https://doi.org/10.46647/ijetms.2021.v05i06.004>
- [44] Kotsiantis, S., Pierrakeas, C., & Pintelas, P. (2004). PREDICTING STUDENTS' PERFORMANCE IN DISTANCE LEARNING USING MACHINE LEARNING TECHNIQUES. *Applied Artificial Intelligence*, 18(5), 411–426.
<https://doi.org/10.1080/08839510490442058>
- [45] Rubio, M. A. (2020). Automated Prediction of Novice Programmer Performance Using Programming Trajectories. In *Lecture notes in computer science* (pp. 268–272). https://doi.org/10.1007/978-3-030-52240-7_49
- [46] Moonsamy, D., Naicker, N., T, T., & E, R. (2021). A Meta-analysis of Educational Data Mining for Predicting Students Performance in Programming. *International Journal of Advanced Computer Science and Applications*, 12(2).
<https://doi.org/10.14569/ijacsa.2021.0120213>
- [47] Sehaba, K. (2020). Learner Performance Prediction Indicators based on Machine Learning. In *In Proceedings of the 12th International Conference on Computer Supported Education (CSEDU 2020)* (pp. 47–57). Science and Technology Publications. <https://doi.org/10.5220/0009396100470057>
- [48] Mustapha, S. M. F. D. S. (2023). Predictive Analysis of Students' Learning Performance Using Data Mining Techniques: A Comparative Study of Feature Selection Methods. *Applied System Innovation*, 6(5), 86.
<https://doi.org/10.3390/asi6050086>
- [49] Brooks, C., Kovanović, V., & Nguyen, Q. (2023). Predictive modeling of student success. In *Edward Elgar Publishing eBooks* (pp. 350–369).
<https://doi.org/10.4337/9781800375413.00027>
- [50] Ouahi, M., Khouilji, S., & Kerkeb, M. L. (2024). Advancing Sustainable Learning Environments: A Literature Review on Data Encoding Techniques for Student Performance Prediction using Deep Learning Models in Education. *E3S Web of Conferences*, 477, 00074. <https://doi.org/10.1051/e3sconf/202447700074>
- [51] Ahadi, A. (2016). Early Identification of Novice Programmers' Challenges in Coding Using Machine Learning Techniques. In *Proceedings of the 2016 ACM Conference on International Computing Education Research (ICER '16)*.
<https://doi.org/10.1145/2960310.2960339>

- [52] Ahmed, U. Z., Sindhgatta, R., Srivastava, N., & Karkare, A. (2019). Targeted Example Generation for Compilation Errors. *2021 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 327–338. <https://doi.org/10.1109/ase.2019.00039>
- [53] Alalawi, K., Athauda, R., & Chiong, R. (2023). Contextualizing the current state of research on the use of machine learning for student performance prediction: A systematic literature review. *Engineering Reports*, 5(12). <https://doi.org/10.1002/eng2.12699>
- [54] Anand, V. K., Rahiman, S. K. A., George, E. B., & Huda, A. S. (2018). Recursive clustering technique for students' performance evaluation in programming courses. In *Majan International Conference (MIC)* (pp. 1–5). <https://doi.org/10.1109/mintc.2018.8363153>
- [55] Baker, R. S., & Yacef, K. (2009). The State of Educational Data Mining in 2009: A Review and Future Visions. *Educational Data Mining*, 1(1), 3–17. <https://doi.org/10.5281/zenodo.3554658>
- [56] Bergin, S., & Reilly, R. (2006, December 1). *Predicting Introductory Programming Performance: A Multi-Institutional Multivariate Study*. Learning & Technology Library (LearnTechLib). <https://www.learntechlib.org/p/104475/>
- [57] Chen, S., & Ding, Y. (2023). A Machine Learning Approach to Predicting Academic Performance in Pennsylvania's Schools. *Social Sciences*, 12(3), 118. <https://doi.org/10.3390/socsci12030118>
- [58] Guo, G., Wang, H., Bell, D., Bi, Y., & Greer, K. (2003). KNN Model-Based Approach in Classification. In *Lecture notes in computer science* (pp. 986–996). https://doi.org/10.1007/978-3-540-39964-3_62
- [59] Hosseini, R., Brusilovsky, P., Yudelso, M., & Hellas, A. (2017). Stereotype Modeling for Problem-Solving Performance Predictions in MOOCs and Traditional Courses. In *UMAP '17: Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization* (pp. 76–84). ACM Digital Library. <https://doi.org/10.1145/3079628.3079672>
- [60] Llanos, J., Bucheli, V. A., & Restrepo-Calle, F. (2023). Early prediction of student performance in CS1 programming courses. *PeerJ Computer Science*, 9, e1655. <https://doi.org/10.7717/peerj-cs.1655>

- [61] Samonte, M. J. C., Evangelista, R. C. B., & Marcelo, D. N. B. (2024). Adapting DAS3H Model for a Personalized Distributed Practice Schedule to Improve Long-Term Memorization in Designing an Intelligent Programming Language Tutor. In *Proceedings of the 2024 9th International Conference on Distance Education and Learning* (Vol. 43, pp. 75–82). <https://doi.org/10.1145/3675812.3675854>
- [62] Effenberger, T., & Pelánek, R. (2019). Measuring Students' Performance on Programming Tasks. In *L@S '19: Proceedings of the Sixth (2019) ACM Conference on Learning @ Scale*. ACM Digital Library. <https://doi.org/10.1145/3330430.3333639>
- [63] *Google Colab*. (n.d.). <https://colab.research.google.com/>
- [64] *Wampserver - Fichiers et Addons*. (n.d.). <https://wampserver.aviatechno.net/>