1. Alqasrawi, Y., Azzeh, M., & Elsheikh, Y. (2022). *Locally weighted regression with different kernel smoothers for software effort estimation*. <https://doi.org/10.1016/J.SCICO.2021.102744>.
2. Boehm, B. (2002). *Software engineering economics*.
3. Conte, S. D., Dunsmore, H. E., & Shen, V. Y. (1986). *Software engineering metrics and models*.
4. Corazza, A., Martino, S. D., Ferrucci, F., Gravino, C., & Mendes, E. (2011). Investigating the use of Support Vector Regression for web effort estimation. *Empirical Software Engineering*. <https://doi.org/10.1007/S10664-010-9138-4>.
5. Corazza, A., Martino, S. D., Ferrucci, F., Gravino, C., Sarro, F., & Mendes, E. (2013). Using tabu search to configure support vector regression for effort estimation. *Empirical Software Engineering*. <https://doi.org/10.1007/S10664-011-9187-3>.
6. Corazza, A., Martino, S., Ferrucci, F., Gravino, C., Sarro, F., & Mendes, E. (2010). How effective is Tabu search to configure support vector regression for effort estimation? *PROMISE ’10*. <https://doi.org/10.1145/1868328.1868335>.
7. Huang, L., Kang, J., Wan, M., Fang, L., Zhang, C., & Zeng, Z. (2021). Solar Radiation Prediction Using Different Machine Learning Algorithms and Implications for Extreme Climate Events. *Frontiers in Earth Science*. <https://doi.org/10.3389/FEART.2021.596860>
8. Kaur, R., & Gupta, N. (2022a). *CFS-MHA*. <https://doi.org/10.4018/IJISP.313663>.
9. Kaur, R., & Gupta, N. (2022b). CFS-MHA: A Two-Stage Network Intrusion Detection Framework. *International Journal of Information Security and Privacy*. <https://doi.org/10.4018/IJISP.313663>.
10. Kocaguneli, E., Menzies, T., & Keung, J. (2012). On the Value of Ensemble Effort Estimation. *IEEE Transactions on Software Engineering*. <https://doi.org/10.1109/TSE.2011.111>.
11. Macho, C., Beyer, S., McIntosh, S., & Pinzger, M. (2021). The nature of build changes. *Empirical Software Engineering*. <https://doi.org/10.1007/S10664-020-09926-4>.
12. Maimuna, M., Rahman, N., Ahmed, R., & Arefin, M. S. (2022). *Data Mining for Software Engineering: A Survey*. <https://doi.org/10.1007/978-3-030-93247-3_86>.
13. Mendes, E., & Kitchenham, B. (2004). Further comparison of cross-company and within-company effort estimation models for Web applications. *10th International Symposium on Software Metrics, 2004. Proceedings.* <https://doi.org/10.1109/METRIC.2004.1357920>.
14. Pascarella, L. (2020). *Augmented Fine-Grained Defect Prediction for Code Review*. <https://doi.org/10.4233/UUID:E553E8AE-73BE-4718-AB93-81F466DB7347>.
15. Sarro, F., Petrozziello, A., & Harman, M. (2016). Multi-objective Software Effort Estimation. *2016 IEEE/ACM 38th International Conference on Software Engineering (ICSE)*. <https://doi.org/10.1145/2884781.2884830>.
16. Shepperd, M., & Macdonell, S. (2012). Evaluating prediction systems in software project estimation. *Information & Software Technology*. <https://doi.org/10.1016/J.INFSOF.2011.12.008>.
17. Shepperd, M., & Schofield, C. (1997). Estimating Software Project Effort Using Analogies. *IEEE Trans. Software Eng.* <https://doi.org/10.1109/32.637387>.
18. Tawosi, V., Moussa, R., & Sarro, F. (2022a). Agile Effort Estimation: Have We Solved the Problem Yet? Insights From A Replication Study. *IEEE Transactions on Software Engineering*. <https://doi.org/10.1109/TSE.2022.3228739>.
19. Tawosi, V., Moussa, R., & Sarro, F. (2022b). Deep Learning for Agile Effort Estimation Have We Solved the Problem Yet? *ArXiv*.
20. Villalobos-Arias, L., Quesada-López, C., Guevara-Coto, J., Martínez, A., & Jenkins, M. (2020). Evaluating hyper-parameter tuning using random search in support vector machines for software effort estimation. *PROMISE*. <https://doi.org/10.1145/3416508.3417121>.
21. Wen, J., Li, S., Lin, Z., Hu, Y., & Huang, C. (2012). Systematic literature review of machine learning based software development effort estimation models. *Information & Software Technology*. <https://doi.org/10.1016/J.INFSOF.2011.09.002>.