



The availability of large amounts of data in Continuous Integration (CI) systems allows companies to utilize machine learning (ML) methods to optimize CI processes. The predictive performance of these methods can be hindered by noise in code change data. Using design science research and controlled experiments, this thesis examines the impact of noise-handling techniques in CI. Two ML-based methods, MeBoTS and HiTTs, are developed for regression testing. A taxonomy and a class noise-handling approach (DB) are created to reduce class noise. Controlled experiments are conducted to examine the effect of class noise-handling on MeBoTS' performance. The results show that handling class noise using DB improves test case selection and code change request predictions. Further, memory management and complexity code changes should be tested with performance-related tests. The "majority filter" algorithm is the most effective in improving the prediction of build outcomes and code change requests.

This thesis highlights the importance of handling class noise in code change data to improve test case selection, build outcomes, and change request predictions. It also shows that using code-to-test dependencies offers an effective way to perform regression testing. Finally, it shows that software engineers do not necessarily need to remove attribute noise to gain improvements in test selection.



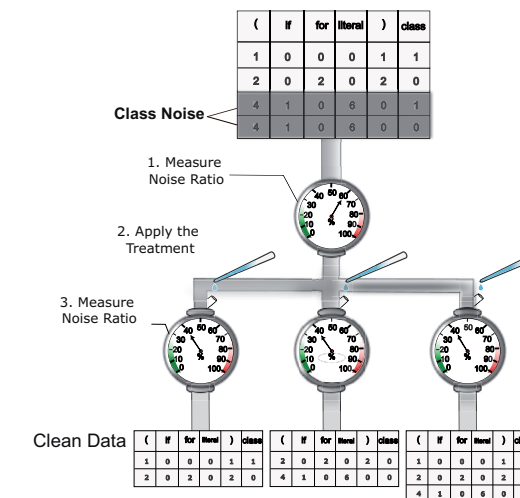
*Khaled Al Sabbagh*  
Department of Computer Science and Engineering  
Division of Interaction Design and Software Engineering

*Khaled Al Sabbagh*

Improving the Performance of Machine Learning-based Methods for Continuous Integration by Handling Noise

# Improving the Performance of Machine Learning-based Methods for Continuous Integration by Handling Noise

*Khaled Al Sabbagh*



DEPARTMENT OF COMPUTER SCIENCE  
AND ENGINEERING

