**Summary Document**

**Paper Title:** Black-Box Testing of Deep Neural Networks through Test Case Diversity

**DOI:** [10.1109/TSE.2023.3243522](https://doi.org/10.1109/TSE.2023.3243522)

The aim of this research is to evaluate the effectiveness of black-box diversity metrics in the testing of Deep Neural Networks (DNNs) by providing a practical mechanism that does not require access to the internal details of the DNN models or their training data. The study focuses on understanding how diversity metrics relate to fault detection capabilities and compares their performance with current white-box methods.

The research involves a comprehensive empirical evaluation based on several research questions with the scope of understanding the relationship between diversity metrics and fault detection in DNNs. The study uses controlled experiments to measure actual data diversity, using three adapted diversity metrics: Geometric Diversity (GD), Normalized Compression Distance (NCD), and Standard Deviation (STD). The evaluation includes the application of these metrics on four datasets and five DNN models.

The results demonstrate that Geometric Diversity (GD) outperforms the other diversity metrics and state-of-the-art coverage criteria in terms of fault-revealing capabilities. There is a positive and statistically significant correlation between GD and faults in DNNs across all configurations. Furthermore, GD is computationally more efficient than the studied coverage metrics, often being three-to-five times faster to compute. The study also found no strong correlation between diversity and coverage metrics, indicating that diverse input sets do not necessarily increase DNN model coverage and vice versa.

The findings suggest that GD can be effectively used as a black-box approach to guide the testing of DNN models, especially in contexts where access to model internals is restricted. For research, this implies a need for further investigation into black-box diversity metrics and their application in various DNN testing scenarios, such as test selection, minimization, and generation. For practice, adopting GD for testing can enhance fault detection capabilities while reducing computational overhead, making it a viable alternative to white-box coverage metrics in practical testing environments