Title: A Partial Reproduction of MAHAKIL

Original Authors: Kwabena Ebo Bennin, Jacky Keung, Passakorn Phannachitta, Akito Monden, and Solomon Mensah, titled, "Mahakil: Diversity based oversampling approach to alleviate the class imbalance issue in software defect prediction." TSE 2018

Reproduced Paper Authors: Amritanshu Agrawal, Tim Menzies, titled, “Is better data better than better data miners? (on the benefits of tuning SMOTE for defect prediction)”, ICSE 2018

What: Bennin et al. proposed a method Mahakil which tackles the problem of class imbalance for software defect prediction. In our paper, we reproduced their method and used it to further extend the work on class imbalance problem. Note that this works satisfies the ACM definition of “reproduction” since we coded up their algorithms from scratch (then applied to the same data they explored).

Why: Bennin et al. compared their method against many state-of-the-art methods on multiple measures and showed that their method improves on all, making their method as the most recent state-of-the-art. We reproduced their package as it was important in our analysis of SMOTUNED to compare against the recent state-of-the-art method rather than previous ones.

How and Where: The code and results of Bennin et al. work was not open sourced. We reproduced the code from the pseudocode given in the paper. We verified our implementation on their datasets, and achieved close to their values ± 0.1. The difference could be due to different random seed. It may be possible that some of their initial assumptions could not have been reproduced as those details could be missing. Reproduced package of Mahakil is now available at https://github.com/ai-se/MAHAKIL_imbalance. Our SMOTUNED package is already open-sourced and could be found in the ICSE 2018 work or at http://tiny.cc/smotuned.

Discussion: The authors proposed the pseudocode which made it easier for us to take an effort to reproduce the package. In future, we would like to recommend that every research should be open source including the code, and the results. This will reduce time and effort to reproduce the code and verify the implementations. It will also further any research in a faster way resonating with the rapid changes in current technologies. It will also be nice to use a service like Arxiv.org which is a repository of electronic preprints even before it is published in some conference or journal. This will bring

researchers to get up-to-date with the current ongoing research even before it gets published, further advancing the research at a rapid pace.