

**“PENERAPAN *HYPERPARAMETER OPTIMIZATION* PADA  
ALGORITMA *DECISION TREE CART* DAN *RANDOM FOREST* UNTUK  
MENEMUKAN FAKTOR KREDIT MACET”**

**NANDA RAHMA ANANTA**

**Program Studi Informatika, Fakultas Matematika dan Ilmu Pengetahuan  
Alam, Universitas Sebelas Maret**

**ABSTRAK**

Pada kasus program kredit pinjaman di Bank XYZ, masalah yang sering timbul dalam pelaksanaan perjanjian kredit perbankan adalah keadaan dimana debitur lalai untuk melakukan kewajibannya, yang dapat menyebabkan adanya kredit macet. Penerapan metode klasifikasi kredit macet yang terbaik sangat dibutuhkan untuk penanggulangan dan pembatasan kredit macet di Bank XYZ. Metode klasifikasi *decision tree* CART dan *random forest* kemudian diaplikasikan pada penelitian ini. Untuk menemukan model *tree* yang terbaik, *hyperparameter optimization* diterapkan pada *decision tree* CART dan *random forest*. Sebelum melakukan klasifikasi, dilakukan *data preprocessing*. Namun, pada proses *data preprocessing*, ditemukan bahwa terdapat ketidakseimbangan kelas pada data. Untuk mengatasi permasalahan ini, metode *resampling Random Over Sampling* (ROS), *Random Under Sampling* (RUS), dan *Synthetic Minority Over-sampling Technique* (SMOTE) digunakan. Hasil dari penelitian ini menunjukkan bahwa bila dibandingkan dengan metode *resampling* lain, data yang melalui tahap ROS dan SMOTE menghasilkan data yang optimal. Untuk perbandingan *accuracy*, *precision*, *recall*, dan *f1-score*, metode *Random Forest* dengan data ROS menunjukkan hasil yang terbaik dengan  $n\_estimators = 50$  di persentase 100%, 100%, 100% dan 100%. Dan berdasarkan model terbaik yang telah dihasilkan, tiga faktor utama penyebab kredit macet adalah nilai taksasi agunan, jangka waktu, penghasilan; dimana banyaknya nilai dari masing-masing variabel perlu dipertimbangkan kembali untuk meminimalisir terjadinya kredit macet untuk beberapa tahun yang akan datang.

**Keyword:** *Decision Tree CART, Random Forest, Hyperparameter Optimization, Resampling*

**“PENERAPAN *HYPERPARAMETER OPTIMIZATION* PADA  
ALGORITMA *DECISION TREE CART* DAN *RANDOM FOREST* UNTUK  
MENEMUKAN FAKTOR KREDIT MACET”**

**NANDA RAHMA ANANTA**

*Department of Informatic, Mathematic and Science Faculty,  
Universitas Sebelas Maret*

**ABSTRACT**

*In the case of a loan credit program at Bank XYZ, the problem that often arises in the implementation of banking credit agreements is a situation where the debtors are negligent in performing their obligations, which can lead to non-performing loan. The best method of classification for non-performing loan is needed for the prevention and limitation of non-performing loan at Bank XYZ. The decision tree CART and random forest classification methods are then applied in this study. To find the best tree model, hyperparameter optimization is applied to the CART and random forest decision trees. Before conducting the classification, data preprocessing is carried out. However, in the process of data preprocessing, it was found that there was a class imbalance in the data. To overcome this problem, the resampling methods such as Random Over Sampling (ROS), Random Under Sampling (RUS), and Synthetic Minority Over-sampling Technique (SMOTE) are used. The results of this study indicates that when compared with other resampling methods, data that goes through the ROS stage produces optimal data. For the accuracy comparison, the Decision Tree CART method shows the best results; and for precision, recall, and f1-measure comparisons, the Random Forest method, instead, shows the best results. For accuracy, precision, recall, and f1-score comparison using ROS and SMOTE data, Random Forest method with ROS using  $n\_estimators = 5$  shows the best results at 100%, 100%, 100% and 100%. And based on the best model that has been produced, the three main factors causing bad credit are interest rate, tenor, and income; where the value of each variable needs to be reconsidered to minimize the occurrence of bad credit for the next few years.*

**Keyword:** *Decision Tree CART, Random Forest, Hyperparameter Optimization, Resampling*