

# Kombinasi Inisial Filtering Oversampling dengan Metode Ensemble Classifier pada Klasifikasi Data Imbalanced

Gede Angga Pradipta<sup>1</sup>, Putu Desiana Wulaning Ayu<sup>2</sup>

Institut Teknologi dan Bisnis STIKOM Bali

e-mail: <sup>1</sup>angga\_pradipta@stikom-bali.ac.id, <sup>2</sup>wulaning\_ayu@stikom-bali.ac.id

Diajukan: 1 Februari 2023; Direvisi: 4 Maret 2023; Diterima: 7 Maret 2023

## Abstrak

Permasalahan dataset tidak seimbang (*imbalanced dataset*) terjadi ketika distribusi kelas dalam dataset tidak seimbang. Ini bisa menjadi masalah dalam machine learning karena model memiliki kecenderungan untuk memprediksi kelas mayoritas, yang mengakibatkan kinerja bias terhadap kelas mayoritas dan akurasi yang rendah terhadap kelas minoritas. Penelitian ini bertujuan untuk melakukan analisis pengaruh proses oversampling data untuk penanganan klasifikasi data imbalanced. Pada penelitian ini melakukan komparasi pengaruh proses oversampling pada algoritma machine learning. Algoritma machine learning yang digunakan adalah metode ensemble dengan observasi pada dua base classifier yaitu support vector machine (SVM) dan naive bayes (NB). Metode oversampling yang digunakan adalah Synthetic Minority Oversampling Teechnique (SMOTE-IPF) dan SMOTE Borderline. Eksperimen dilakukan terhadap 9 dataset yang bertipe imbalanced. Dari hasil eksperimen dapat diketahui bahwa metode ensemble baik dengan base SVM dan Naive Bayes mengalami peningkatan saat adanya proses oversampling tersebut. Dari 9 dataset yang digunakan pada eksperimen, seluruhnya mengalami peningkatan performa yang diukur menggunakan parameter akurasi, presisi, recall dan F Measure. Pada base classifier SVM peningkatan performa tertinggi pada dataset Abalone 19 dengan penambahan oversampling SMOTE Borderline. Peningkatan akurasi terjadi dari 64.5% menjadi 84.5%. Kemudian pada base classifier Naive Bayes peningkatan peforma teringgi pada dataset Abalone 19 dengan penambahan Borderline SMOTE. Peningkatan akurasi terjadi dari 62.3% menjadi 82.3%.

**Kata kunci:** Imbalanced data, Oversampling, SMOTE-IPF, SMOTE Borderline, Ensemble learning.

## Abstract

The issue of imbalanced datasets arises when the class distribution within the dataset is uneven, meaning that some classes have significantly fewer or more samples than others. This can pose a serious problem in machine learning because models tend to predict the majority class, leading to biased performance towards the majority class and low accuracy towards the minority class. The aim of this study is to analyze the impact of oversampling data processes for handling imbalanced data classification problems. The study compares the effects of oversampling processes on machine learning algorithms. The machine learning algorithms used in this study are ensemble methods with observations on two base classifiers, namely support vector machine and naive Bayes. The oversampling methods used are SMOTE-IPF and SMOTE Borderline. Both methods are based on the concept of filtering samples at the beginning for the process of generating synthetic data. Experiments were conducted on 9 imbalanced datasets. From the experimental results, it can be observed that the ensemble method with both SVM and Naive Bayes base classifiers experiences improvement with the oversampling process. From the 9 datasets used in the experiments, all of them show performance improvement measured using accuracy, precision, recall, and F Measure parameters. In the SVM base classifier, the highest performance improvement is observed in the Abalone 19 dataset with the addition of Borderline SMOTE oversampling. The accuracy improvement increases from 64.5% to 84.5%. Then, in the Naive Bayes base classifier, the highest performance improvement is observed in the Abalone 19 dataset with the addition of Borderline SMOTE. The accuracy improvement increases from 62.3% to 82.3%.

**Keywords:** Imbalanced data, Oversampling, SMOTE-IPF, SMOTE Borderline, Ensemble learning.

## 1. Pendahuluan

Model *ensemble* [1], [2], [3] [4] adalah sebuah pendekatan di dalam *machine learning* yang memperbaiki performa dari single classifier dengan menggunakan beberapa *base classifier* dan mengkombinasikan *output* dari masing-masing *classifier* tersebut untuk menentukan keputusan akhir. Ide utama di balik model *ensemble* adalah membuat beberapa *base classifier* dari dataset asli dan kemudian mengkombinasikan atau menggabungkan prediksi mereka saat diuji data uji baru. Ketika melakukan klasifikasi terhadap data uji baru tersebut, diberikan kepada semua anggota (*classifier*) dalam *ensemble*. Hasil prediksi dari semua *classifier* tersebut kemudian dipertimbangkan dalam fase kombinasi, yang juga dikenal sebagai *classifier fusion* atau *aggregation* [2]. Berdasarkan prinsip ini, cara melakukan kombinasi dari beberapa *classifier* menjadi faktor kunci dalam meningkatkan performa yang dihasilkan.

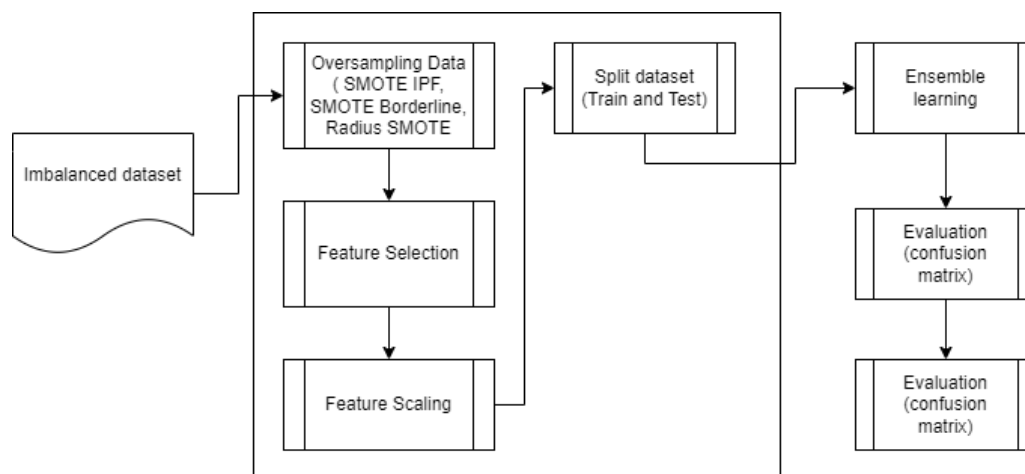
Berdasarkan pada dari struktur kontraksi dari metode *ensemble*, maka metode ini dapat dibedakan menjadi dua kategori yaitu *dependent* dan *independent* [5]. Pada kategori *dependent*, luaran dari metode klasifikasi selanjutnya tergantung dan diukur berdasarkan *output* dari metode klasifikasi sebelumnya. Berbeda dengan kategori *independent*, *output* dari suatu metode klasifikasi yang digunakan didapatkan secara terpisah dan kemudian dikombinasikan untuk mendapatkan *output* yang lebih akurat. Salah satu jenis metode yang termasuk dalam kategori *independent* adalah metode *voting* yang menggabungkan beberapa metode klasifikasi untuk mencapai hasil akurasi prediksi yang lebih baik. Skema *voting* merupakan bentuk dan cara yang paling sederhana untuk diimplementasikan, namun pemilihan kombinasi *rule* yang membentuk sekumpulan metode klasifikasi menjadi faktor utama dalam membangun sebuah model *Esemble Classifier* [6]. Secara garis besar metode *voting* ini dapat dibagi menjadi dua pendekatan yaitu *unweighted* dan *weighted voting*. Metode *unweighted voting* beberapa contoh yang diterapkan adalah *minimum probability*, *maximum probability*, *majority voting*, *product of probability*, dan *average of probabilities*. Sedangkan contoh pada *weighted voting* adalah *simple weighted voting*, *rescaled weighted voting*, *best-worst weighted voting*, dan *quadratic best-worst weighted voting* [7]. *Simple majority voting* merupakan sebuah *decision rule* yang melakukan pemilihan satu dari banyak alternatif *outputs* yang didapatkan dari anggota (*classifiers*) paling banyak divoting. *Majority voting* tidak memerlukan tuning paramaters sehingga hasil dari prediksi akhir merupakan murni dari kelas yang mendapatkan *voting* terbanyak. Berbeda dengan *weighted voting*, bobot *voting* memiliki keberagaman nilai/bobot untuk setiap *classifier* berdasarkan pada performa dari setiap *classifier* tersebut. Semakin baik performa *single classifier* yang ada dalam *members* maka akan mendapatkan bobot lebih tinggi dan begitu pula sebaliknya. Maka dari dari penentuan bobot dari setiap *classifiers* merupakan isu kritis yang dihadapi untuk mampu mendapatkan model *ensemble* yang sesuai. Penelitian yang dilakukan oleh [8] untuk mendeteksi wajah dengan fitur histogram dan menggunakan *ensemble classifier*. Kombinasi metode yang digunakan *Artificial Neural Network* (ANN), *K Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), *Random Forest* (RF), REP tree, *Naive Bayes*, CART. Dataset dibagi menjadi beberapa sub-dataset dengan Metode *bootstrapping* dan masing-masing sub-dataset diberikan *classifier* yang berbeda-beda. Pengujian dilakukan dengan membandingkan beberapa metode perhitungan untuk *voting* menggunakan metode *the product of probabilities*, *majority voting*, *minimum probability*, *maximum probability*, dan *median combination rule*. Petinrin [9] melakukan prediksi *bioactive molecule* menggunakan metode *majority voting*. Penelitian melakukan beberapa eksperimen dengan melibatkan beberapa kombinasi *single classifier* seperti *Support Vector Machine* (SVM), *Naive Bayes* (NB), *Decision Tree* (DT), *k-Nearest Neighbour* (k-NN), and *Random Forest* (RF) untuk mengetahui akurasi yang tertinggi. Metode *voting* yang digunakan adalah *majority voting* dengan bobot yang sama di antara semua *classifier*. Penelitian [10] menggunakan *hard voting* yang di mana Setiap *classifier* mempunyai bobot yang sama dalam perhitungan hasil *majority voting*. Penelitian ini membandingkan hasil antara metode *filtering + Single classifier* dan *majority voting* untuk sebuah dataset dari UCI *machine learning repository* dengan yang memiliki karakteristik *imbalanced dataset*. Pada model *majority voting*, dataset dibagi menjadi beberapa subdata yang lebih kecil dan masing-masing diberikan *classifier* untuk memprediksi *class label* dari sebuah *instance*. Metode klasifikasi yang digunakan adalah *Multilayer Perceptron* dengan *Back Propagation* (MLP), *Decision Tree* (C4.5), *Locally Weighted Learning* (LWL), *5-Nearest Neighbors* (5-NN), *Nearest Neighbor with generalization* (NNge), *Naive Bayes* (NB), *Ripple Down Rule learner* (RIDOR), *Random Forest* (RandForest), RIPPER. Hasil pengujian menunjukkan dengan menggunakan Metode *majority voting* lebih meningkatkan akurasi dibandingkan dengan menggunakan metode *filtering data*. Selain itu metode *majority voting* juga lebih efektif dalam komputasi pada *training data* dibandingkan dengan *filtering data*. Aytung [6] menggunakan *Multiobjective Differential Evolution* untuk melakukan pembobotan pada masing-masing *classifier*. Pengujian dilakukan pada *public data sentiment analysis* dan menunjukkan hasil di mana *MODE-Based Weighted Voting* mampu memperoleh akurasi yang lebih tinggi dari metode *stacking* dan *voting weighted* lainnya. Penggunaan bobot berdasarkan nilai *F Measure* juga dilakukan pada study [11] dengan memanfaatkan

metode *majority voting* yang dipadukan dengan *bagging* (BagMOOV) untuk memprediksi dan analisis penyakit jantung. *Classifier* yang digunakan pada penelitian ini adalah *Naive Bayes* (NB), *linear regression* (LR), *quadratic discriminant analysis* (QDA), *instance based learner* (IBL) dan *support vector machine* (SVM) dan pembobotan dilakukan dengan cara *multi-objective optimization*. Nilai F-measure digunakan sebagai *objective function* untuk mengukur bobot dari setiap *classifier* dalam menghasilkan keputusan akhir dalam memprediksi suatu kelas dari sebuah data. W. Chen [12], mengusulkan metode *adaptive weighting multi-classifier fusion decision* menggunakan *shanon entropy* mengukur performa model klasifikasi.

Entropi Shannon, sebuah konsep dari teori informasi, dihitung dengan menggunakan informasi probabilitas posterior dari metode SVM untuk mendapatkan bobot secara adaptif bagi sampel-sampel yang berbeda yang digunakan dalam pengambilan keputusan bobot. Skema pembobotan adaptif dilakukan pada penelitian [13] untuk klasifikasi hepatocellular carcinoma. Untuk memperoleh bobot dari setiap *classifier* menggunakan parameter akurasi dengan melihat ketetanggaan terdekat. Hasil prediksi dari *single classifier* terhadap *k-nearest neighbors* tersebut kemudian dibandingkan dengan label aktual. Semakin baik akurasi yang diperoleh maka bobot yang diberikan semakin tinggi.

Selain menggunakan pembobotan pada setiap *base classifier* pada metode *ensemble*, penanganan masalah *imbalanced data* juga dapat dilakukan pada level data. Pada penelitian ini melakukan analisis terhadap kombinasi metode *oversampling* dan *ensemble learning* untuk menghadapi kondisi *imbalanced dataset*.

## 2. Metode Penelitian



Gambar 1. Model Klasifikasi pada data *imbalanced*.

Pada penelitian ini, metode oversampling yang digunakan adalah SMOTE IPF dan Borderline SMOTE. Kedua metode ini memiliki pendekatan pembuatan data sintetik berdasarkan pada perhitungan jarak ketetanggaan dan juga lokasi antara data sampel ke data di sekitarnya. Selanjutnya setelah proses pembuatan data sintetik tersebut selesai, maka dilanjutkan dengan tahapan seleksi fitur. Pada proses ini menggunakan metode information gain untuk mendapatkan fitur-fitur yang relevan dan berkontribusi dalam proses *learning* pola data latih. Tahap ketiga adalah proses scaling fitur. Proses ini bertujuan untuk memastikan bahwa semua fitur (atau variabel) memiliki skala yang seragam. Fitur dengan skala yang tidak seimbang dapat menyebabkan masalah saat digunakan dalam beberapa algoritma pembelajaran mesin atau teknik optimasi. Fitur dengan skala yang besar dapat mendominasi fitur dengan skala yang kecil dalam perhitungan, terutama dalam algoritma yang sensitif terhadap perbedaan skala.

Dalam prosesnya ketiga metode oversampling di atas dievaluasi dan dibandingkan hasilnya saat data tersebut dilatih menggunakan metode Ensemble Multiclassifier. Evaluasi performa model dilakukan dengan menggunakan metode confusion matrix dengan menggunakan parameter pengukuran akurasi, presisi, recall, dan F-Measure.

## 2.1 Oversampling Data

### 2.1.1 SMOTE IPF

SMOTE IPF (Iterative Proportional Fitting) adalah modifikasi dari SMOTE yang bertujuan untuk meningkatkan kualitas sintesis yang dihasilkan oleh SMOTE, terutama dalam konteks kelas minoritas yang sangat kecil. SMOTE IPF bekerja dengan cara mengambil sampel dari kelas minoritas dan secara iteratif memperbarui distribusi kelas sintesis yang dihasilkan. Berikut adalah langkah-langkah umum dalam SMOTE IPF:

- a) Menentukan KNN (K-Nearest Neighbors): Seperti halnya SMOTE, SMOTE IPF menggunakan KNN untuk menemukan tetangga terdekat dari setiap contoh dalam kelas minoritas.
- b) Pemilihan Contoh Minoritas: Contoh-contoh minoritas yang akan digunakan untuk menghasilkan sampel sintesis dipilih.
- c) Iterasi: SMOTE IPF melakukan iterasi untuk setiap contoh minoritas yang dipilih. Pada setiap iterasi, contoh minoritas tersebut digunakan untuk membuat sampel sintesis, dan distribusi kelas sintesis diperbarui berdasarkan distribusi aktual dari kelas minoritas dan mayoritas.
- d) Perbarui Distribusi: Distribusi kelas sintesis diperbarui untuk mencerminkan distribusi aktual dari kelas minoritas dan mayoritas. Hal ini bertujuan untuk memastikan bahwa sampel sintesis yang dihasilkan tidak hanya berada di sekitar wilayah yang padat dalam ruang fitur, tetapi juga mencerminkan distribusi yang sebenarnya dari kelas minoritas.

SMOTE IPF bertujuan untuk mengatasi beberapa kelemahan yang terkait dengan SMOTE standar, seperti kecenderungan untuk menghasilkan sampel sintesis yang terletak di wilayah yang padat dari ruang fitur atau untuk menghasilkan sampel sintesis yang terlalu dekat dengan contoh minoritas yang ada. Dengan memperbarui distribusi kelas sintesis secara iteratif, SMOTE IPF dapat menghasilkan sampel sintesis yang lebih berkualitas dan mewakili distribusi kelas yang sebenarnya.

### 2.1.2 Borderline SMOTE

Borderline SMOTE adalah variasi dari algoritma SMOTE (Synthetic Minority Over-sampling Technique) yang dirancang khusus untuk menangani kasus ketidakseimbangan kelas di mana kelas minoritas (kelas yang kurang mewakili) terletak di sekitar batas antara kelas minoritas dan mayoritas. Berikut adalah langkah-langkah umum dari algoritma Borderline SMOTE:

1. Identifikasi Sampel Batas:  
 $\text{BorderlineSamples} = \{x_i \mid x_i \text{ adalah contoh dari kelas minoritas dan mayoritas tetangganya} \geq k\}$   
 $\text{BorderlineSamples} = \{x_i \mid x_i \text{ adalah contoh dari kelas minoritas dan mayoritas tetangganya} \geq k\}$   
 Di mana  $k$  adalah jumlah tetangga dari kelas mayoritas yang dianggap sebagai ambang batas.
2. Seleksi Tetangga:  
 Misalkan  $x_i$  adalah salah satu sampel batas yang telah diidentifikasi, dan  $N_i$  adalah himpunan tetangga dari kelas mayoritas yang terhubung ke  $x_i$ . Proses ini memilih tetangga-tetangga dari  $N_i$  untuk setiap  $x_i$ .
3. Pembuatan Sampel Sintesis:  
 Untuk setiap sampel batas  $x_i$ , Borderline SMOTE menggunakan proses yang mirip dengan SMOTE untuk membuat sampel sintesis. Rumus SMOTE yang umum digunakan adalah sebagai berikut:

$$x_{new} = x_i + \lambda \times (x_j - x_i) \tag{1}$$

Di mana:

- $x_i$  adalah sampel minoritas batas.
  - $x_j$  adalah salah satu tetangga yang dipilih dari himpunan tetangga  $N_i$ .
  - $\lambda$  adalah rasio antara 0 dan 1 yang menentukan seberapa jauh sampel sintesis  $x_{new}$  berada dalam garis antara  $x_i$  dan  $x_j$ .
4. Penambahan Sampel Sintesis:  
 Sampel sintesis  $x_{new}$  yang dihasilkan kemudian ditambahkan ke dataset asli.

**2.1.3 Penskalaan Fitur (Min Max Scaling)**

Metode ini mentransformasikan nilai fitur ke dalam rentang tertentu, seperti [0, 1]. Rumusnya adalah:

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \tag{2}$$

Di mana  $x'$  adalah nilai fitur yang telah dirobustkan,  $x$  adalah nilai asli fitur,  $\max(x)$  adalah median fitur, dan  $\text{IQR}(x)$  adalah rentang antar-kuartil dari fitur.

**2.1.4 Ensemble Multiclasifier**

Ensemble multiclassifier voting adalah teknik di mana beberapa model klasifikasi bekerja sama untuk membuat prediksi akhir dengan cara melakukan pemungutan suara. Setiap model memberikan suara untuk kelas tertentu, dan kelas dengan mayoritas suara dipilih sebagai prediksi akhir. Misalnya, jika terdapat tiga model klasifikasi yang berpartisipasi dalam ensemble, dan prediksi masing-masing model adalah sebagai berikut: Model 1 memprediksi kelas A, Model 2 memprediksi kelas B, dan Model 3 memprediksi kelas A. Dalam kasus ini, kelas A akan dipilih sebagai prediksi akhir karena mendapatkan mayoritas suara dari model. Secara matematis, prediksi akhir  $\hat{y}$  dalam ensemble multiclassifier voting dapat dihitung dengan menggunakan rumus berikut:

$$\hat{y} = \operatorname{argmax}_y \sum_{i=1}^N w_i \cdot 1(\hat{y}_i = y) \tag{3}$$

Di mana:

- $N$  adalah jumlah model dalam ensemble.
- $\hat{y}_i$  adalah prediksi dari model ke  $i$ .
- $w_i$  adalah bobot yang diberikan kepada prediksi model ke  $i$  (biasanya setara untuk setiap model).
- $1(\hat{y}_i=y)$  adalah fungsi indikator yang bernilai 1 jika prediksi model ke  $i$  sama dengan kelas  $y$ , dan 0 jika tidak.

**3. Hasil dan Pembahasan**

**3.1. Deskripsi Dataset Imbalanced**

Pada studi ini menggunakan data dari UCI repository, Keel repository, dan umbilical cord image pada penelitian kami sebelumnya. Beberapa dataset yang digunakan merupakan data yang memiliki karakteristik imbalanced data yang ditandai dengan nilai imbalance ratio yang tinggi. Karakteristik dari dataset yang digunakan ditunjukkan pada tabel 1. Terdapat 9 dataset yang digunakan untuk menguji performa model usulan.

Tabel 1. Karakteristik data imbalanced pada penelitian ini.

Name	Size	Imbalance Ratio	Attributes	Classes	Source
ecoli	336	8.60	7	8	UCI
Abalone19	4175	129.43	7	2	Keel
Glass-6	214	6.37	9	2	Keel
Wine	178	2.01	13	3	Keel
liver disorder	345	0.725	6	2	UCI
Thyroid-2	215	5.14	5	2	Keel
vehicle1	846	2.89	18	2	Keel
Pima-ssl10	691	22.03	8	3	Keel
Page-Block	5471	8.78	10	2	Keel

### 3.2. Eksperimen pada Oversampling dan Ensemble Classifier

Proses eksperimen dan analisis membandingkan antara performa model ensemble saat tidak ada proses oversampling dan ditambahkan metode oversampling yaitu SMOTE-IPF dan Borderline SMOTE. Tabel 2 menunjukkan hasil eksperimen pada Ensemble Classifier tanpa adanya metode oversampling.

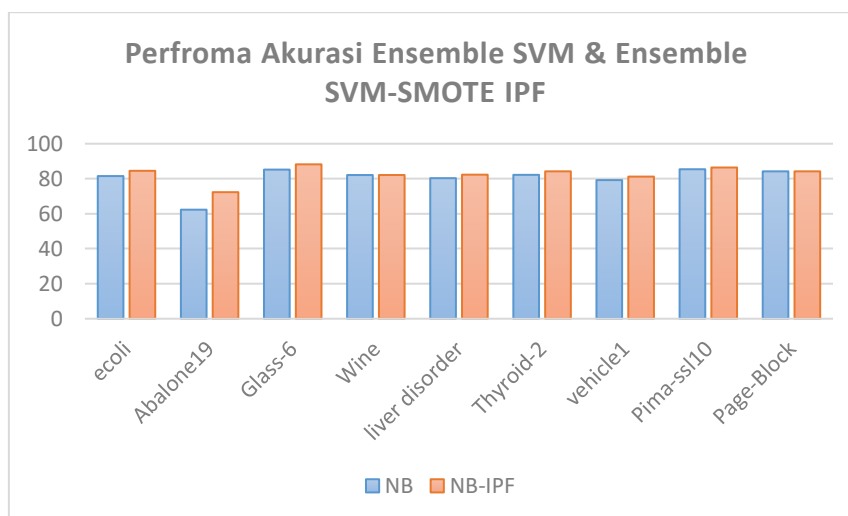
Tabel 2. Hasil Uji Performa Ensemble Classifier tanpa proses oversampling.

Dataset	Akurasi		Presisi		Recall		F Measure	
	SVM	NB	SVM	NB	SVM	NB	SVM	NB
ecoli	82.3	81.5	82.2	82.2	82.1	81.2	82.8	82.2
Abalone19	64.5	62.3	63.2	64.2	63.2	62.2	62.2	63.2
Glass-6	86.2	85.2	87.2	90.2	86.3	91.2	86.7	89.2
Wine	81.2	82.1	89.2	88.4	90.2	86.4	83.2	84.3
liver disorder	82.2	80.3	82.4	89.2	90.2	82.1	92.2	81.4
Thyroid-2	82.2	82.2	83.2	80.1	81.2	83.2	84.2	88.2
vehicle1	81.4	79.2	80.2	79.2	79.3	79.9	80.2	79.3
Pima-ssl10	85.4	85.4	86.2	82.2	86.3	86.2	86.3	86.3
Page-Block	84.3	84.2	84.3	85.2	86.2	86.2	86.3	85.2

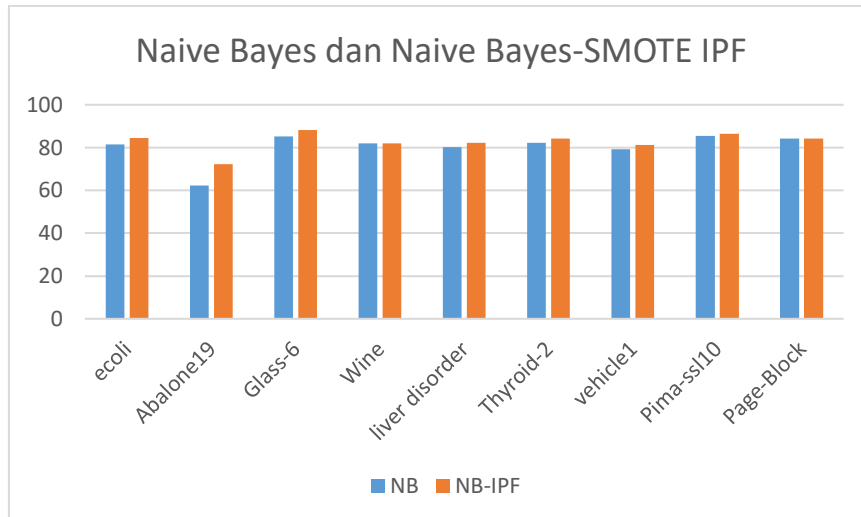
Selanjutnya eksperimen dilanjutkan dengan menambahkan teknik oversampling menggunakan metode SMOTE IPF. Paramater dari SMOTE IPF yang digunakan adalah dengan jumlah  $k = 5$  dan pengukuran *distance matrix* menggunakan formula *euclidean distance*. Metode pengukuran performa menggunakan pendekatan metode confusion matrix dengan paramater pengukuran yaitu akurasi, presisi, recall, dan F-Measure. Pada metode Ensmble melibatkan dua algoritma *machine learning* sebagai *base classifier* yaitu *support vector machine* (SVM) dan *naive bayes* (NB). Berdasarkan dari hasil tersebut dapat diketahui peningkatan performa yang terjadi saat penambahan adanya proses oversampling data menggunakan SMOTE IPF. Dari sembilan dataset yang digunakan, 8 di antara mengalami peningkatan nilai akurasi, presisi, recall, dan F-Measure. Gambar 2 menunjukkan peningkatan peforma akurasi dari algoritma ensemble tanpa dan dengan penambahan proses oversampling.

Tabel 3 Hasil Uji Performa Ensemble Classifier terhadap data oversampling SMOTE-IPF

Dataset	Akurasi		Presisi		Recall		F Measure	
	SVM	NB	SVM	NB	SVM	NB	SVM	NB
ecoli	85.1	84.5	84.2	84.2	85.1	83.2	84.8	83.2
Abalone19	74.5	72.3	73.2	74.2	73.2	72.2	72.2	73.2
Glass-6	89.2	88.2	88.2	92.2	88.3	92.2	88.7	90.2
Wine	83.2	82.1	89.2	92.4	90.2	89.4	89.2	88.3
liver disorder	83.2	82.3	82.4	93.2	93.2	83.1	93.2	83.4
Thyroid-2	84.2	84.2	84.2	82.1	84.2	84.2	83.2	89.2
vehicle1	82.4	81.2	81.2	80.2	80.3	80.9	81.2	80.3
Pima-ssl10	87.4	86.4	87.2	88.2	87.3	87.2	87.3	87.3
Page-Block	84.3	84.2	84.3	85.2	86.2	86.2	86.3	85.2



Gambar 2. Perbandingan akurasi Ensemble SVM dan Ensemble SVM-SMOTE IPF.



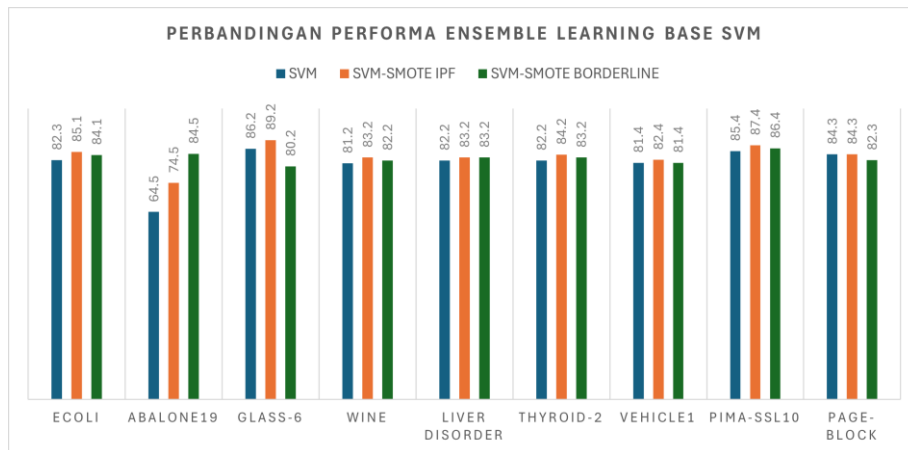
Gambar 3. Perbandingan akurasi Ensemble Naive Bayes dan Ensemble Naive Bayes-SMOTE IPF.

Selanjutnya eksperimen dilanjutkan dengan menambahkan teknik oversampling menggunakan metode SMOTE-Borderline. Parameter dari Borderline-SMOTE yang digunakan adalah yaitu jumlah ketetanggan terdekat adalah 5 dan dengan metode pengukuran jarak *Euclidean*.

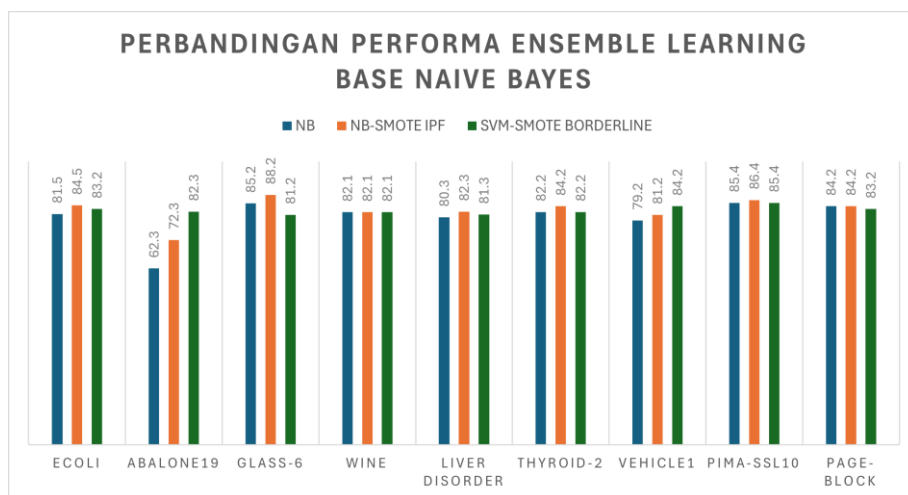
Tabel 4. Hasil Uji Performa *Ensemble Classifier* terhadap data oversampling SMOTE *Borderline*.

Dataset	Akurasi		Presisi		Recall		F Measure	
	SVM	NB	SVM	NB	SVM	NB	SVM	NB
ecoli	84.1	83.2	85.2	83.2	84.1	83.2	84.8	83.2
Abalone19	84.5	82.3	83.2	83.2	83.2	83.2	80.2	80.2
Glass-6	80.2	81.2	81.4	82.2	83.2	82.3	83.2	82.2
Wine	82.2	82.1	89.2	92.4	90.2	89.4	89.2	88.3
liver disorder	83.2	81.3	82.4	91.2	92.2	82.1	91.2	82.4
Thyroid-2	83.2	82.2	82.2	82.1	82.2	82.2	81.2	85.2
vehicle1	81.4	84.2	80.2	81.2	81.3	80.9	80.2	80.3
Pima-ssl10	86.4	85.4	86.2	87.2	85.3	86.2	85.3	84.3
Page-Block	82.3	83.2	83.3	84.2	85.2	85.2	83.3	83.2

Berdasarkan hasil eksperimen ketiga ini, dapat diketahui bahwa pengaruh proses oversampling dengan metode SMOTE Borderline juga mampu memberikan peningkatan yang cukup baik saat dibandingkan dengan tanpa oversampling. Perbandingan antara metode SVM dan Naive Bayes saat menggunakan kombinasi oversampling SMOTE IPF dan Borderline dapat dilihat pada Gambar 4 dan Gambar 5. Saat kedua metode oversampling ini dibandingkan dapat diketahui bahwa peningkatan terjadi hampir pada seluruh dataset. Berdasarkan hasil eksperimen ketiga ini, dapat diketahui bahwa pengaruh proses oversampling dengan metode SMOTE Borderline juga mampu memberikan peningkatan yang cukup baik saat dibandingkan dengan tanpa oversampling. Perbandingan antara metode SVM dan Naive Bayes saat menggunakan kombinasi oversampling SMOTE IPF dan Borderline dapat dilihat pada Gambar 4 dan Gambar 5. Saat kedua metode oversampling ini dibandingkan dapat diketahui bahwa peningkatan performa terjadi hampir pada seluruh dataset. Pada base classifier SVM peningkatan performa tertinggi pada dataset Abalone 19 dengan penambahan oversampling SMOTE Borderline. Peningkatan akurasi terjadi dari 64.5% menjadi 84.5%. Kemudian pada *base classifier* Naive Bayes peningkatan performa tertinggi pada dataset Abalone 19 dengan penambahan Borderline SMOTE. Peningkatan akurasi terjadi dari 62.3% menjadi 82.3%.



Gambar 4. Grafik perbandingan performa akurasi antara SVM, SVM-SMOTE-IPF dan SVM SMOTE Borderline.



Gambar 5. Grafik perbandingan performa akurasi antara Naive Bayes, Naive Bayes-SMOTE-IPF dan Naive Bayes SMOTE Borderline.

#### 4. Kesimpulan

Pada penelitian ini bertujuan untuk melakukan analisis pengaruh proses oversampling data untuk penanganan masalah klasifikasi data imbalanced. Metode *machine learning* saat mempelajari pola dengan data yang kurang akan mengalami kesulitan dalam menemukan decision boundary dari kelas/label tersebut. Pada penelitian ini telah dilakukan komparasi pengaruh proses oversampling pada algoritma *machine learning*. Untuk algoritma *machine learning* yang digunakan pada penelitian ini adalah metode ensemble dengan observasi pada uda base classifier yaitu *support vector machine* dan naive bayes. Metode oversampling yang digunakan adalah SMOTE-IPF dan SMOTE Borderline. Kedua metode ini berlandaskan pada konsep filter sampel di awal untuk proses pembuatan data sintetik. Eksperimen dilakukan terhadap 9 dataset yang bertipe imbalanced. Dari hasil eksperimen dapat diketahui bahwa metode ensemble baik dengan base SVM dan Naive Bayes mengalami peningkatan saat adanya proses oversampling tersebut. Dari 9 dataset yang digunakan pada eksperimen, seluruhnya mengalami peningkatan performa yang diukur menggunakan parameter akurasi, presisi, recall dan F Measure. Pada base classifier SVM peningkatan performa tertinggi pada dataset Abalone 19 dengan penambahan oversampling SMOTE Borderline. Peningkatan akurasi terjadi dari 64.5% menjadi 84.5%. Kemudian pada base classifier Naive Bayes peningkatan performa tertinggi pada dataset Abalone 19 dengan penambahan Borderline SMOTE. Peningkatan akurasi terjadi dari 62.3% menjadi 82.3%.



**Daftar Pustaka**

- [1] L. I. Kuncheva, *Combining pattern classifiers: methods and algorithms*, 2nd ed. Wiley Blackwell, 2014.
- [2] M. Woźniak, M. Graña, and E. Corchado, “A survey of multiple classifier systems as hybrid systems,” *Information Fusion*, vol. 16, no. 1, pp. 3–17, 2014, doi: 10.1016/j.inffus.2013.04.006.
- [3] J. Kittler, I. C. Society, M. Hatef, R. P. W. Duin, and J. Matas, “On Combining Classifiers,” *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE*, vol. 20, no. 3, pp. 226–239, 1998.
- [4] A. Fernández, S. García, M. Galar, and R. C. Prati, “*Learning from Imbalanced Data Sets*,” Springer Nature Switzerland AG 2018, 2018.
- [5] L. Rokach, “Ensemble-based classifiers,” *Artificial Intelligence Review*, no. November 2009, pp. 1–39, 2010, doi: 10.1007/s10462-009-9124-7.
- [6] Aytung Onan, S. Korukoglu, and H. Bulut, “A Multiobjective Weighted Voting Ensemble Classifier Based on Differential Evolution Algorithm for Text Sentiment Classification,” *Expert Systems With Applications*, 2016, doi: 10.1016/j.eswa.2016.06.005.
- [7] F. Moreno-seco, “Comparison of Classifier Fusion Methods for Classification in Pattern Recognition Tasks,” pp. 705–713, 2006.
- [8] Mehmet Akif Yaman, “SS symmetry Comparison of Random Subspace and Voting Ensemble Machine Learning Methods for,” *Symmetry 2018*, vol. 10, 2018, doi: 10.3390/sym10110651.
- [9] O. O. Petinrin, F. Saeed, and T. Al-hadhrami, “Voting-Based Ensemble Method for Prediction of Bioactive Molecules,” *2nd International Conference on Knowledge Engineering and Applications Voting-Based*, pp. 118–122, 2017.
- [10] M. R. Smith and T. Martinez, “The robustness of majority voting compared to filtering misclassified instances in supervised classification tasks,” *Artificial Intelligence Review*, 2016, doi: 10.1007/s10462-016-9518-2.
- [11] S. Bashir, U. Qamar, and F. Hassan, “BagMOOV : A novel ensemble for heart disease prediction bootstrap aggregation with multi-objective optimized voting,” *Australas Phys Eng Sci Med*, 2015, doi: 10.1007/s13246-015-0337-6.
- [12] W. Chen *et al.*, “Rail crack recognition based on Adaptive Weighting Multi-classifier Fusion Decision,” *Measurement: Journal of the International Measurement Confederation*, vol. 123, no. December 2017, pp. 102–114, 2018, doi: 10.1016/j.measurement.2018.03.059.
- [13] K. Wang, C. Huang, and W. Zhou, “Pathological automatic classification of hepatocellular carcinoma based on adaptive weighted multi-classifier fusion,” *Proceedings of 2017 IEEE 2nd Advanced Information Technology, Electronic and Automation Control Conference, IAEAC 2017*, pp. 1489–1493, 2017, doi: 10.1109/IAEAC.2017.8054261.