Riset Informatika C081

Marchel Adias Pradana
21081010084

PENGGUNAAN TEKNIK *SMOTE-ENN* UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE EXTREME GRADIENT BOOSTING (XGBOOST)

Latar Belakang

Kurangnya **kesadaran diri** masyarakat Indonesia dalam memperhatikan **tingkat kesehatan diri** sendiri seperti kolesterol tinggi, tekanan darah tinggi, merokok, kurangnya aktivitas fisik, dan pola makan tinggi lemak.

Menurut **kemkes.go.id.** “**Satu dari Tiga Kematian Disebabkan oleh Jantung**”. Maka dari itu diperlukan **deteksi dini** pada penyakit jantung di Indonesia

Acuan

A Heart Disease Prediction Model Based on Feature Optimization and Smote-Xgboost Algorithm

ANALISIS PENGGUNAAN TEKNIK OVERSAMPLING PADA XGBOOST UNTUK MENGATASI KETIDAKSEIMBANGAN KELAS PADA KLASIFIKASI PENYAKIT JANTUNG

Rumusan Masalah

1

Bagaimana ketidakseimbangan dataset penyakit jantung berpengaruh pada kinerja algoritma XGBoost?

2

Apakah teknik SMOTE-ENN dapat mengatasi ketidakseimbangan kelas pada dataset penyakit jantung untuk performa model XGBoost

Research Gap

1. Minimnya Studi yang Menggabungkan SMOTE dan ENN pada XGBoost untuk Dataset Penyakit Jantung

2. Belum Maksimalnya Penyesuaian Teknik Balancing terhadap XGBoost

3. Minimnya Studi Evaluasi Mendalam tentang SMOTE-ENN terhadap Noise Data terutama pada Dataset Penyakit Jantung dengan Ketidakseimbangan Tinggi

Rumusan Masalah

1. Bagaimana ketidakseimbangan dataset penyakit jantung berpengaruh pada kinerja algoritma XGBoost?

2. Apakah teknik SMOTE-ENN dapat mengatasi ketidakseimbangan kelas pada dataset penyakit jantung untuk performa model XGBoost

Tujuan Penelitian

1. Menguji ketidakseimbangan dataset penyakit jantung berpengaruh terhadap kinerja algoritma XGBoost

2. Mengetahui teknik SMOTE-ENN dalam mengatasi ketidakseimbangan kelas pada dataset penyakit jantung dalam performa model XGBoost

Manfaat Penelitian

Mengevaluasi kombinasi antara model XGBoost dengan teknik Hybridsampling SMOTE-ENN

Referensi terhadap peneliti atau pengembang pada bidang terkait dengan klasifikasi penyakit jantung

MIND MAPPING

SMOTE-ENN + XGBOOST

Mengevaluasi pengaruh balancing data menggunakan SMOTE-ENN terhadap akurasi dan metrik lainnya pada klasifikasi penyakit jantung dengan XGBoost

1. PREPROCESSING

DATASET

Dataset preprocessing : Dataset cleaning (Missing Value & Duplication), Selection Feature, Transformation Data

2.BALANCING DATA

SMOTE

(Synthetic Minority Oversampling Technique): Membuat data sintetik untuk kelas minoritas berdasarkan interpolasi linier

SMOTE-ENN

Mengintegrasikan oversampling dan undersampling untuk dataset yang lebih bersih dan seimbang

ENN

(Edited Nearest Neighbors): Menghapus noise atau sampel yang tidak sesuai berdasarkan mayoritas tetangga terdekat

3.METODE KLASIFIKASI

XGBOOST

Model boosting yang mengoptimalkan kesalahan residual untuk membangun pohon keputusan yang kuat

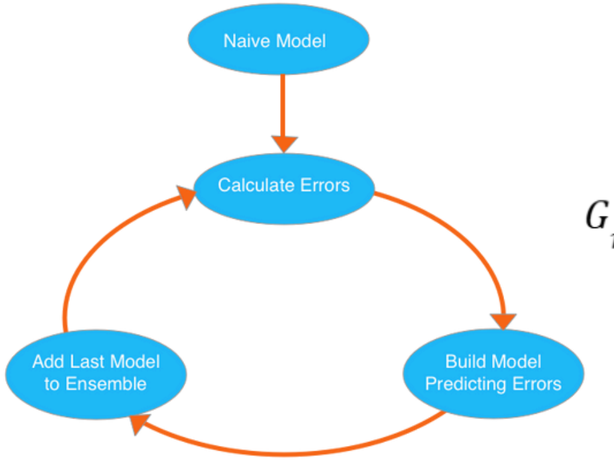
METODE

Balancing Data :

1. SMOTE (Synthetic Minority Oversampling Technique)
- Mengidentifikasi data minoritas
- Membuat KNN antar n data
- Membuat data sintetis antara data dengan neighbors
2. ENN (Edited Nearest Neighbors)
- Menemukan kelas mayoritas
- Identifikasi tetangga terdekat (KNN)
- Bandingkan label mayoritas dengan tetangga, jika label berbeda dari mayoritas, hapus data dari dataset
3. SMOTE-ENN
- Menambahkan data sintetis pada minority class dengan teknik SMOTE
- Mengurangi data dari majority class dengan teknik ENN

XGBOOST Algorithm :

1. Inisialisasi model
2. Hitung Gradien dan Hessian
3. Bangun Decision Tree
4. Pembaruan Prediksi
5. Iterasi Decision Tree
6. Evaluasi Model



Matriks Pengujian:

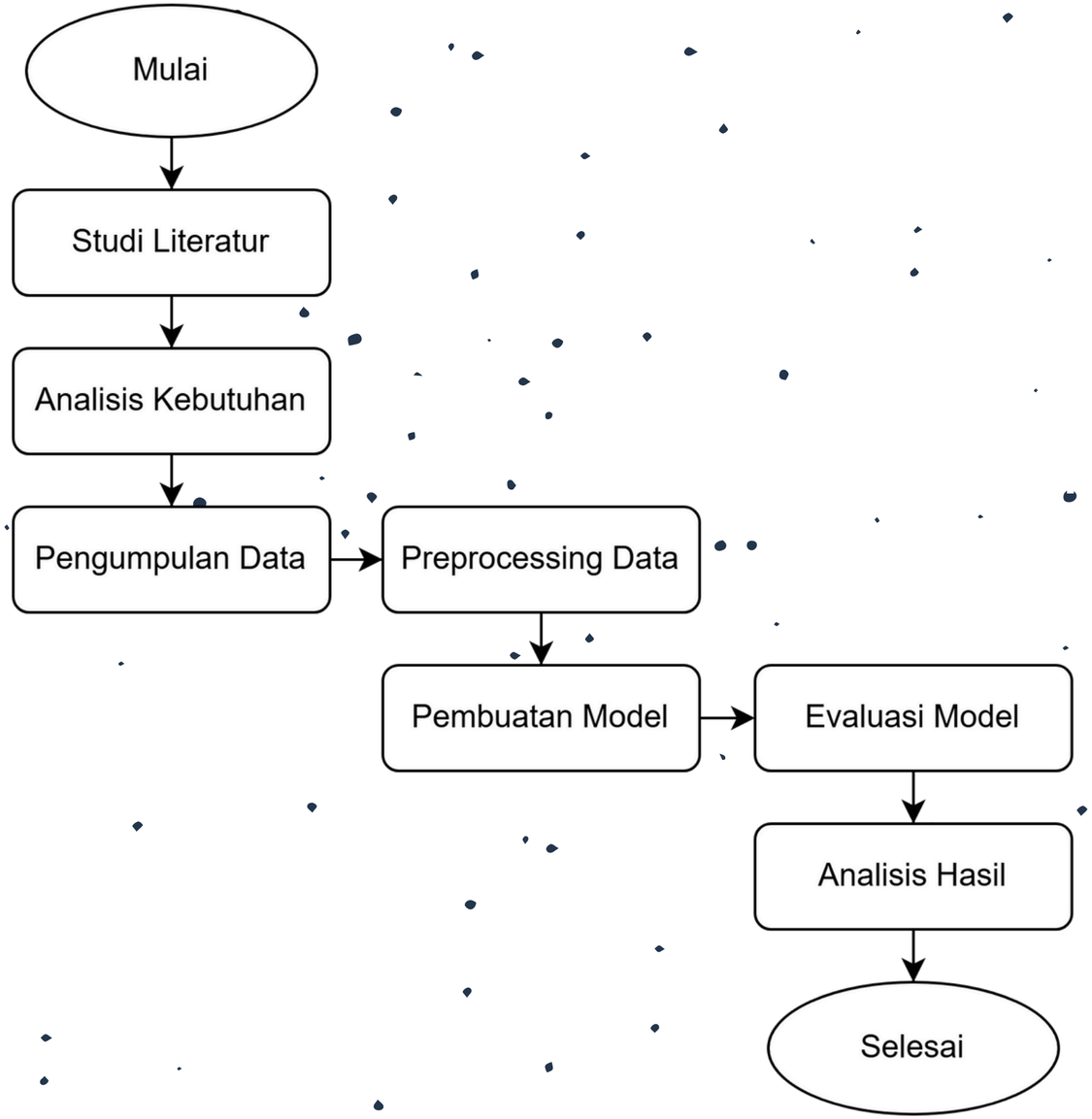
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

$$Sensitivity = \frac{TP}{TP+FN}$$

$$Specificity = \frac{TN}{TN+FP}$$

$$G_{mean} = \sqrt{sensitivity * specificity}$$

Tahapan Penelitian

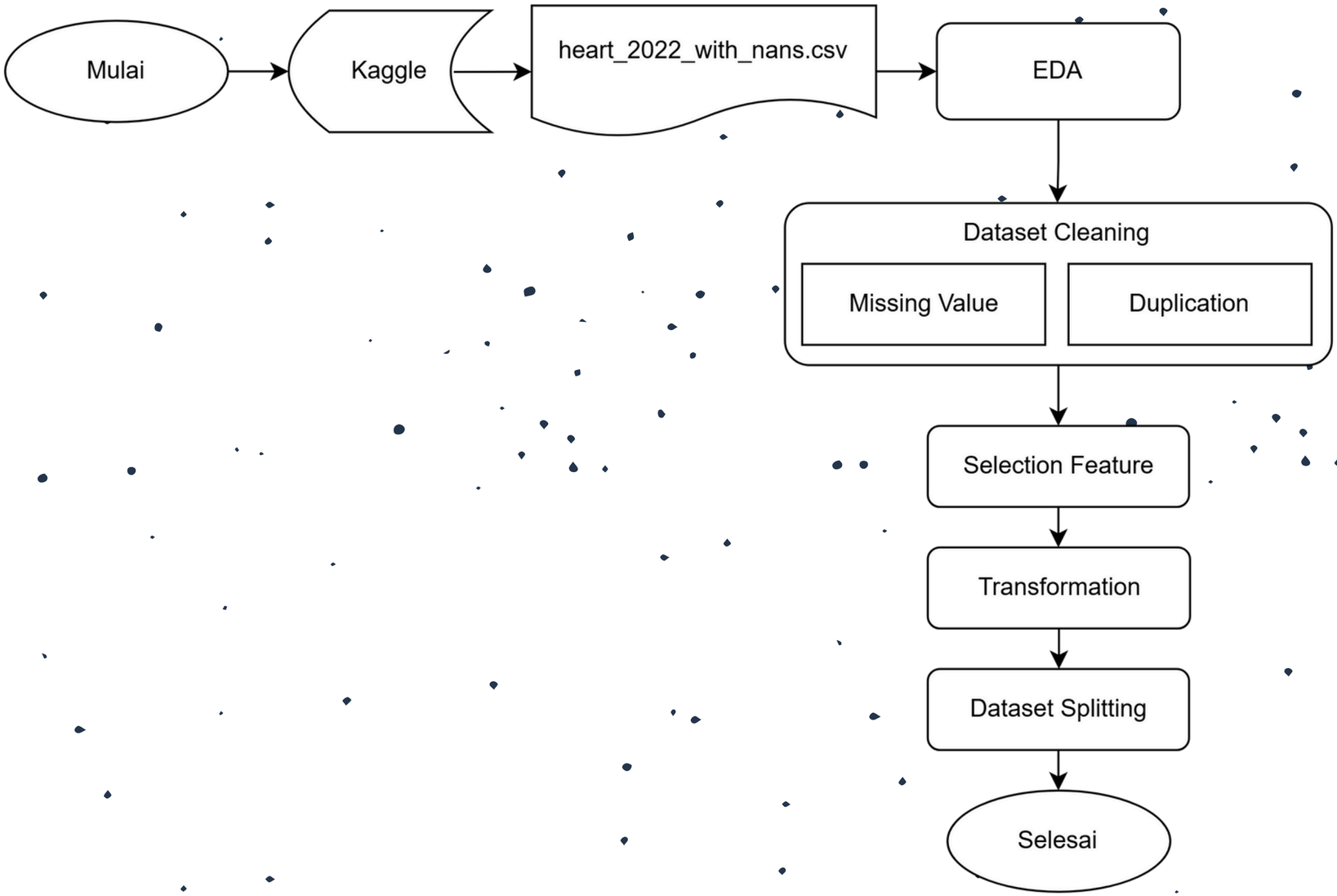


Pengumpulan Data

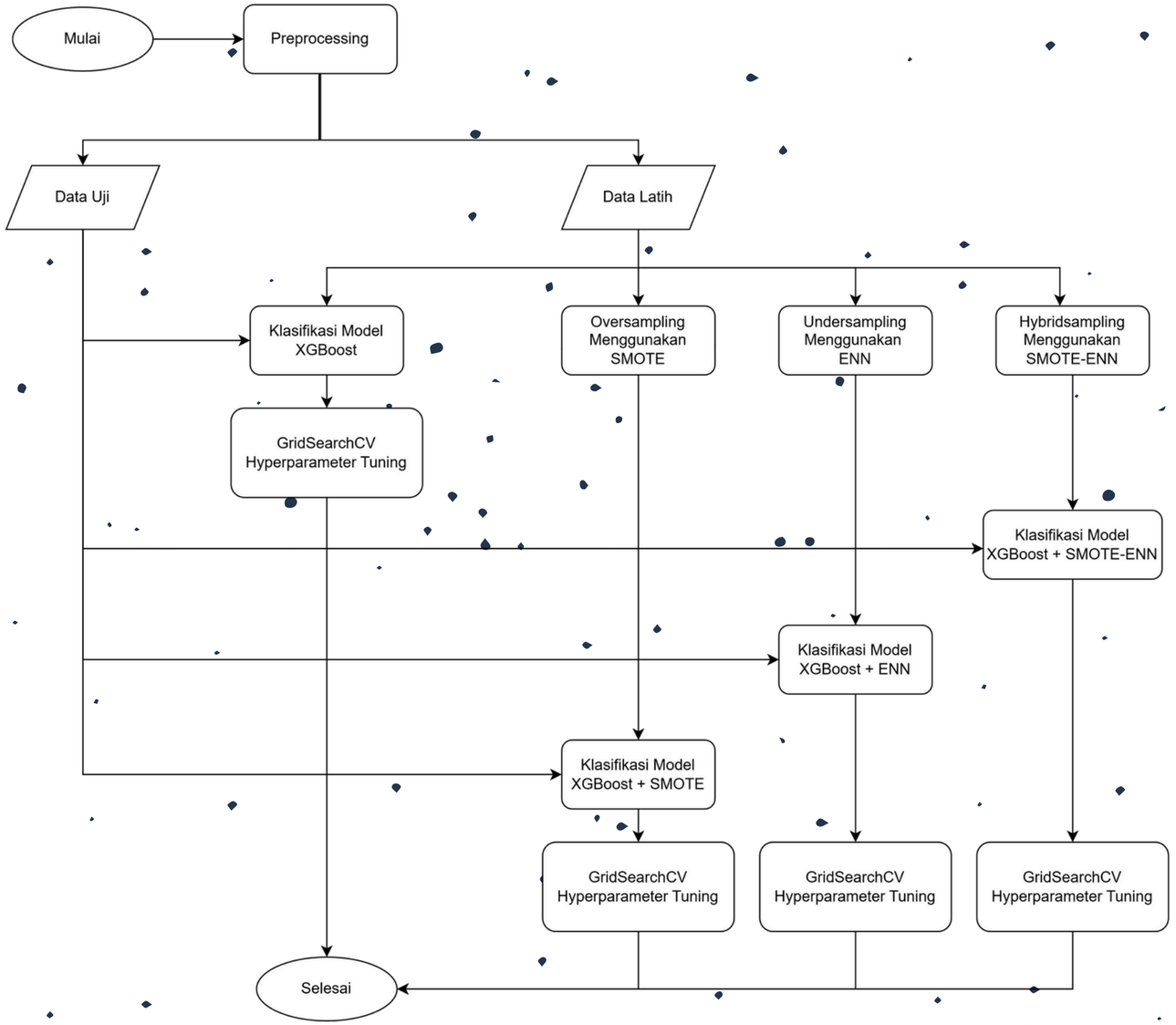
Dataset Kaggle : Indicators of Heart Disease (2022 UPDATE)

https://www.kaggle.com/datasets/kamilpytlak/personal-key-indicators-of-heart-disease

Data Preprocessing

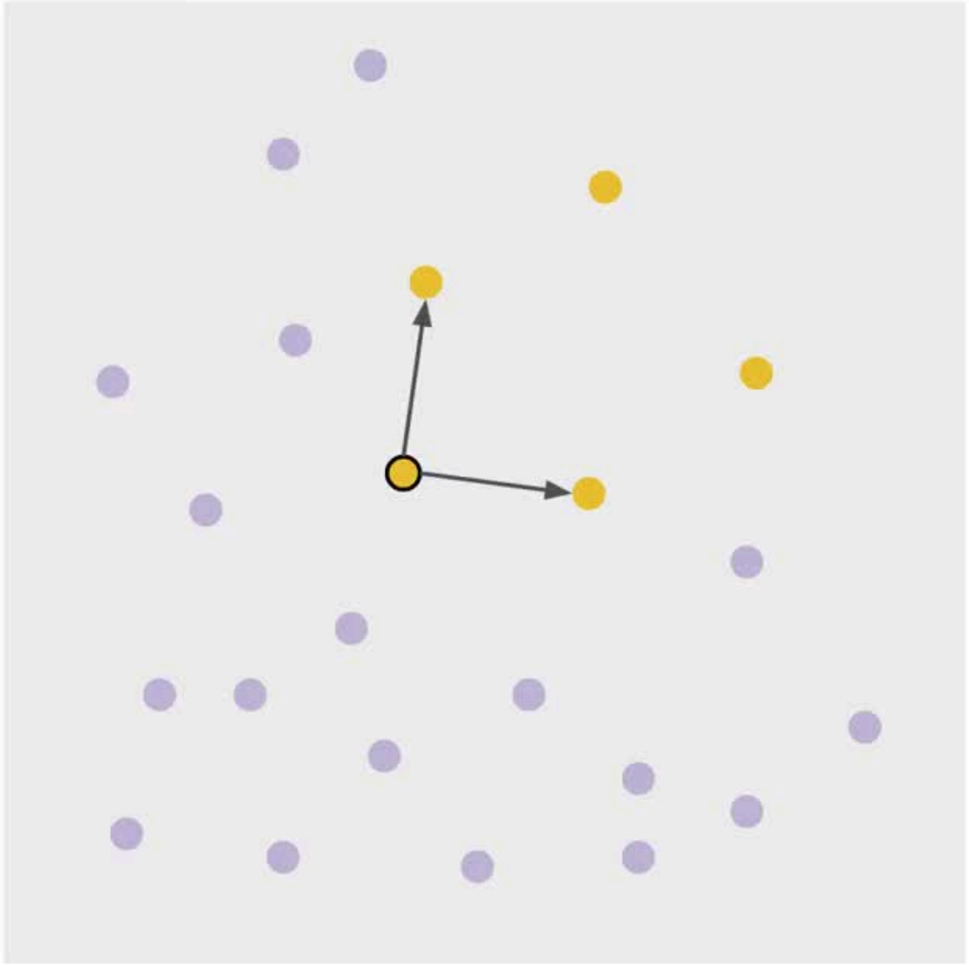


Pembuatan Model



SMOTE

Synthetic Minority Oversampling Technique



The SMOTE Algorithm

FOR OVERSAMPLING THE MINORITY CLASS

1. Identify a data point from the minority class.
2. Find its k nearest neighbor. Here, k is 2.

SMOTE

Synthetic Minority Oversampling Technique

Menghitung jarak tiap data minoritas :

$$dis\,(x_1, x_2) = \sqrt{\sum_{i=0}^{n} (x_{1i} - x_{2i})^2}$$

Dimana;

d(x1,x2) = jarak data x1 dan x2

n = dimensi atribut

x1i = atribut ke i dari data 1

x2i = atribut ke i dari data 2

Pembuatan data sintetis dari setiap data :

$$x_{new} = x_i + (x - x_i) * \delta$$

Dimana;

xnew = data sintetik

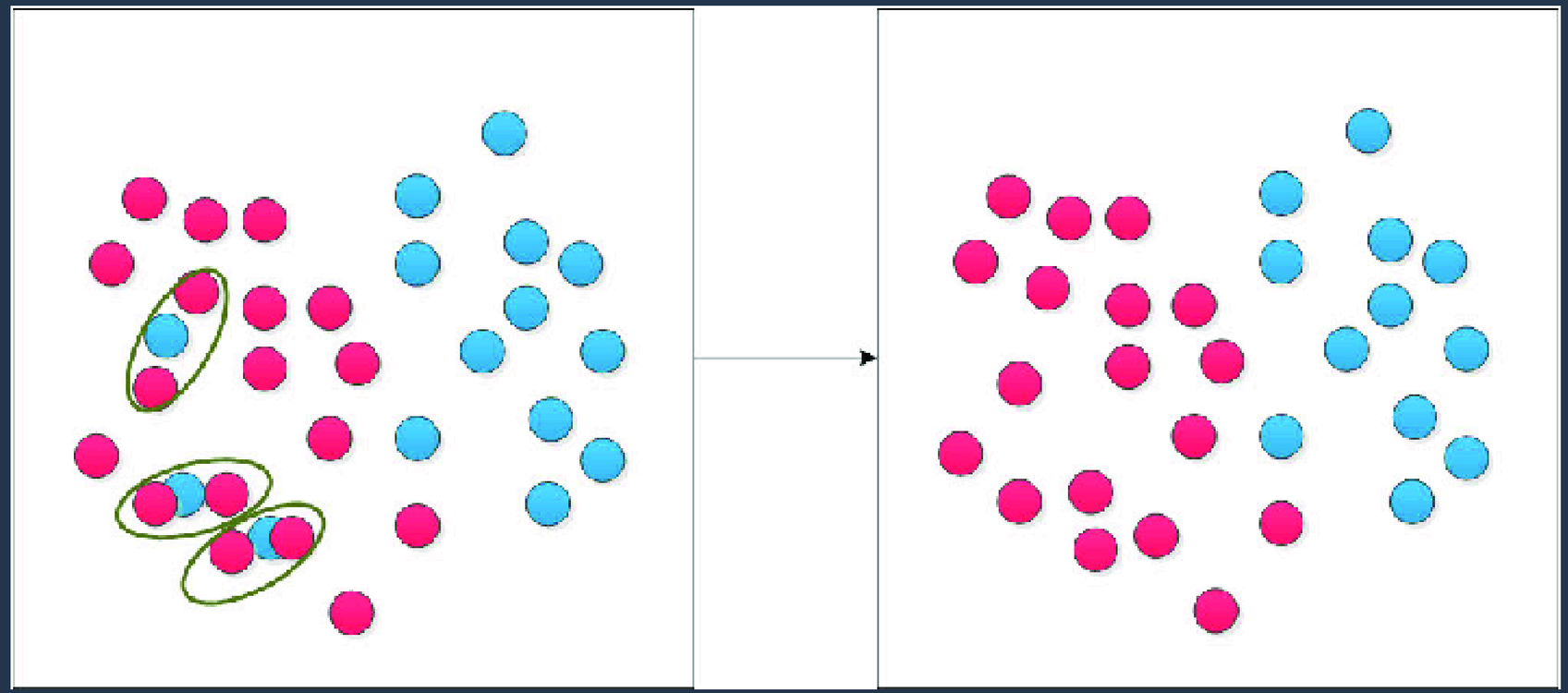
xi = data pada k

x = tetangga terdekat dari xi

δ = bilangan desimal acak antara 0 dan 1

ENN

Edited Nearest Neighbors



ENN

Edited Nearest Neighbors

1. Identifikasi tetangga terdekat (KNN) :

$$dis\ (x_1, x_2) = \sqrt{\sum_{i=0}^{n} (x_{1i} - x_{2i})^2}$$

2. Evaluasi label sampel dengan membandingkan dengan tetangganya
   - Bandingkan label sampel x2i dengan label mayoritas dari tetangganya.
   - Jika label x2i berbeda dari mayoritas tetangga, hapus x1i dari dataset.
3. Ulangi proses tiap sampel

SMOTE-ENN

Hybridsampling

Algorithm 2: The pseudo code of Smote-Enn.

Input : Heart Disease Dataset HDD;
Process:
foreach *sample* s_i *in the minority class of HDD* **do**
    Calculate the K-nearest neighbor samples ks_i of s_i;
    Construct a new data sample $ns = s_i + (\widehat{s_i} - s_i)) + \delta$;
    Add the generated sample ns to HDD;
foreach *sample* h_i *in HDD* **do**
    if h_i *class <> majority class of k-nearest neighbors* **then**
        Remove h_i;
Output : Balanced dataset HDD

A Heart Disease Prediction Model Based on Feature Optimization and Smote-Xgboost Algorithm

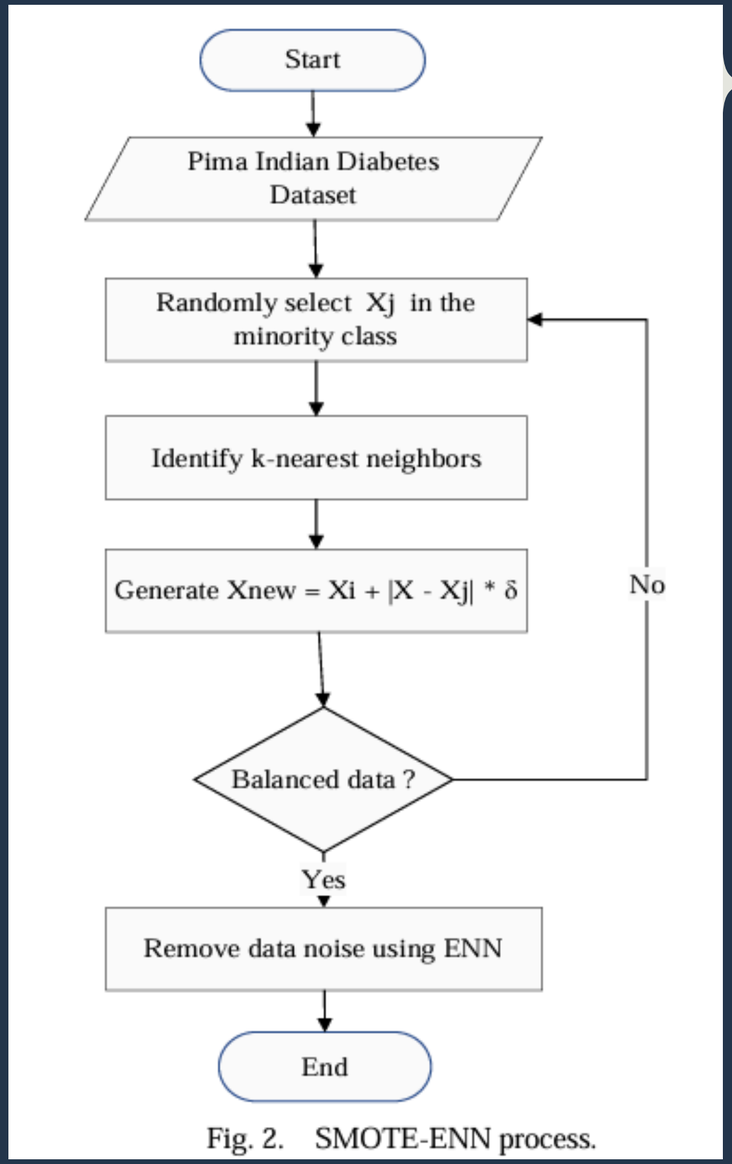
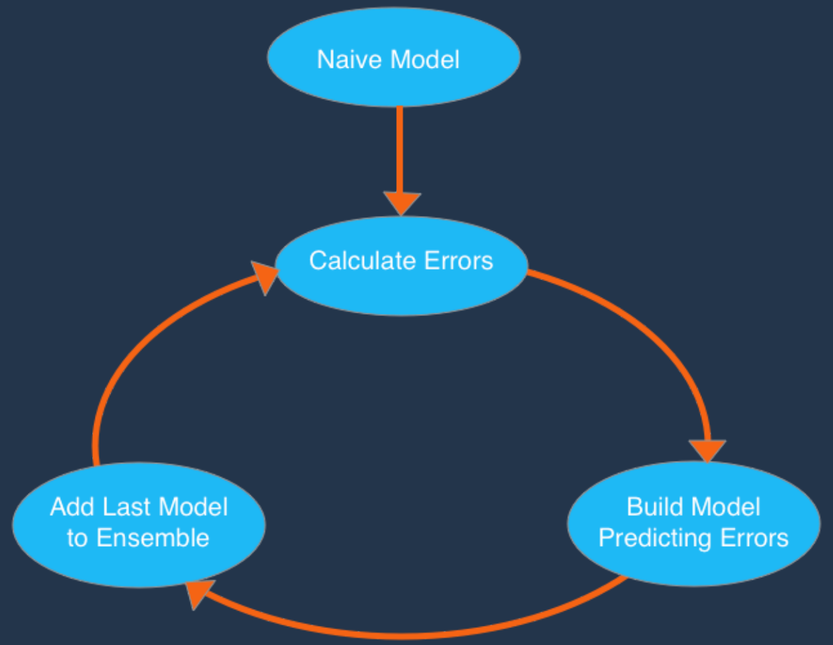


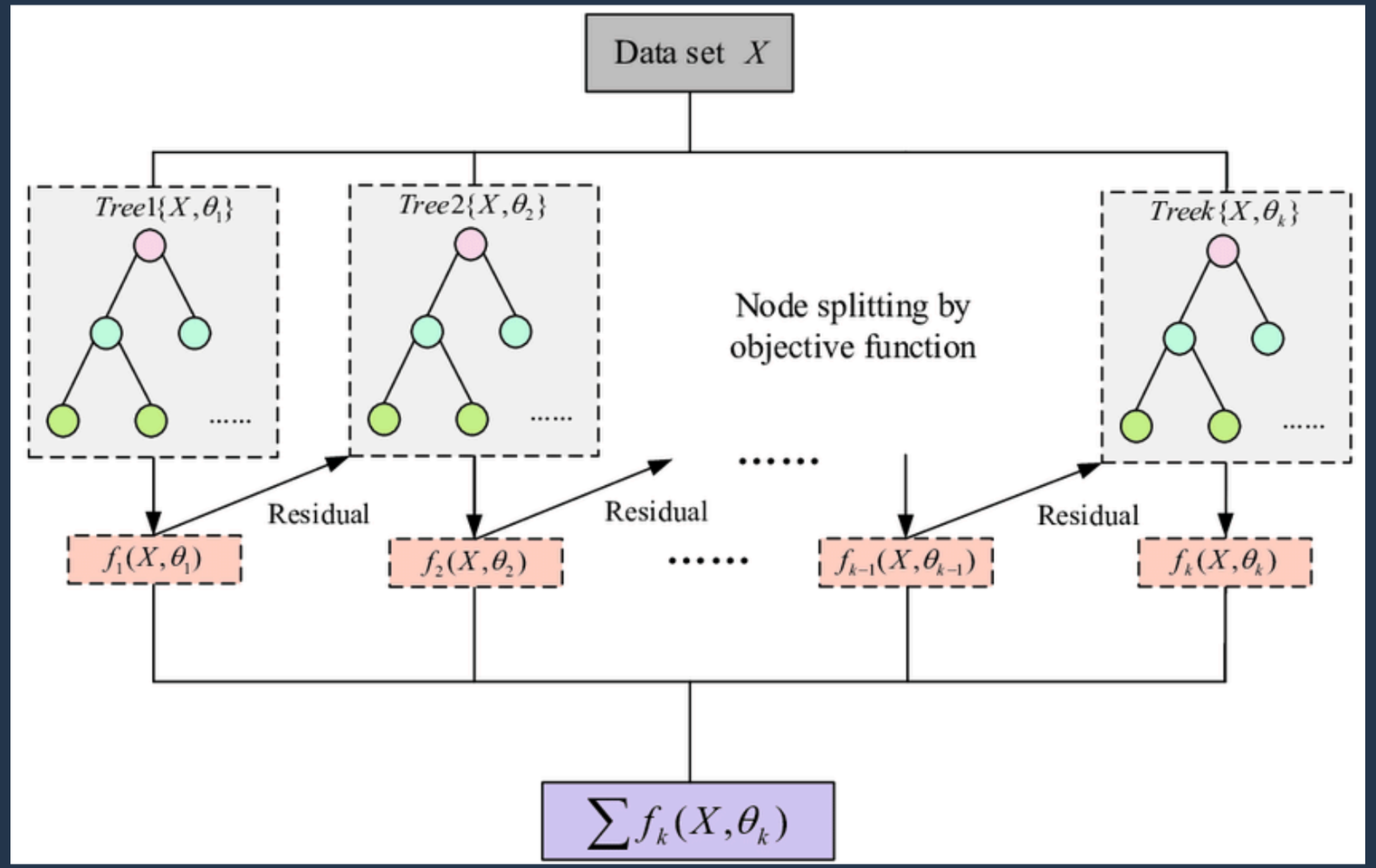
Fig. 2. SMOTE-ENN process.

A New Approach of Hybrid Sampling SMOTE and ENN to the Accuracy of Machine Learning Methods on Unbalanced Diabetes Disease Data

XGBOOST

X Gradient Boost





XGBOOST

X Gradient Boost

Inisialisasi model (klasifikasi biner)

Fungsi loss :

$$\text{Loss} = -\frac{1}{N}\sum_{i=1}^{N}\left[y_i \log(\hat{y}_i) + (1-y_i)\log(1-\hat{y}_i)\right]$$

N: Jumlah total sampel dalam dataset.

y_i: Label aktual dari sampel ke-i (0 untuk kelas negatif, 1 untuk kelas positif).

$\hat{y}_i$: Prediksi probabilitas dari model untuk sampel ke-i bahwa kelas positif benar ($0 \leq \hat{y}_i \leq 1$).

log: Logaritma natural (basis e), digunakan untuk menghitung hubungan logaritmik.

Hitung prediksi awal :

$$\hat{y}_i^{(0)} = \log\frac{p}{1-p}, \quad p = \frac{\text{Jumlah Positif}}{\text{Total Sampel}}$$

p: Probabilitas kelas positif dalam dataset pelatihan

Gradien dan Hessian :

$$g_i = \frac{\partial \text{Loss}}{\partial \hat{y}_i} = \hat{y}_i - y_i$$

g_i: Gradien, yaitu ukuran perubahan fungsi loss terhadap prediksi model pada sampel ke-i.

$\hat{y}_i$: Prediksi probabilitas model untuk sampel ke-i.

y_i: Label aktual dari sampel ke-i.

$$h_i = \frac{\partial^2 \text{Loss}}{\partial \hat{y}_i^2} = \hat{y}_i(1-\hat{y}_i)$$

h_i: Hessian, yaitu ukuran percepatan perubahan fungsi loss terhadap prediksi model.

$\hat{y}_i$: Prediksi probabilitas model untuk sampel ke-i.

$(1-\hat{y}_i)$: Komponen pelengkap dari probabilitas prediksi, memastikan turunan kedua relevan dalam log-loss.

XGBOOST

X Gradient Boost

Petumbuhan Decision Tree :

1.Menentukan split terbaik dengan menghitung Gain :

$$\text{Gain} = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma$$

G_L, G_R: Total gradien di sisi kiri dan kanan split.

H_L, H_R: Total Hessian di sisi kiri dan kanan split.

λ: Parameter regulasi L2.

γ: Penalti untuk menambahkan node baru.

2.Pemangkasan tree (Pruning) :

3.Pembaruan prediksi :

$$\hat{y}_i^{(t+1)} = \hat{y}_i^{(t)} + \eta f_t(x_i)$$

$\hat{y}_i^{(t+1)}$: Prediksi baru untuk sampel ke-i setelah iterasi $t+1$.

$\hat{y}_i^{(t)}$: Prediksi sebelumnya untuk sampel ke-i.

η: Learning rate, mengontrol kontribusi dari setiap pohon.

$f_t(x_i)$: Prediksi pohon keputusan ke-t untuk sampel ke-i.

4.Iterasi hingga n_estimators tercapai

dengan : - Kedalaman maksimum tercapai (ditentukan oleh hyperparameter max_depth).

- Keuntungan tidak signifikan: Jika Gain ≤ 0, split tidak dibuat.

XGBOOST

Matriks Pengujian

Evaluasi Model :

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN}$$

$$Sensitivity = \frac{TP}{TP+FN}$$

$$Specificity = \frac{TN}{TN+FP}$$

TP: True Positive (prediksi 1, label 1).

TN: True Negative (prediksi 0, label 0).

FP: False Positive (prediksi 1, label 0).

FN: False Negative (prediksi 0, label 1).

$$G_{mean} = \sqrt{sensitivity * specificity}$$

Research Progress

Progress yang telah dilakukan :

- **Implementasi Code**
  - **Import Dataset**
  - **Preprocessing Data**
  - **Transformasi Data**
  - **Penggunaan Teknik Resampling (SMOTE, ENN, SMOTEENN)**
  - **Penggunaan Model XGBOOST**
  - **Hyperparameter Tuning Model XGBOOST**
- **Draft Skripsi**
  - **Bab 1-3**
- **Yang Akan Dilakukan Selanjutnya**
  - **Mendapatkan Dataset Primer Terkait Rekam Medis**

Terima Kasih