**SKRIPSI**

# PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST

**MARCHEL ADIAS PRADANA**
NPM 21081010084

**DOSEN PEMBIMBING**
-
-

**KEMENTERIAN PENDIDIKAN, KEBUDAYAAN, RISET, DAN TEKNOLOGI**
**UNIVERSITAS PEMBANGUNAN NASIONAL VETERAN JAWA TIMUR**
**FAKULTAS ILMU KOMPUTER**
**PROGRAM STUDI INFORMATIKA**
**SURABAYA**
**2024**

*Halaman ini sengaja dikosongkan*

**LEMBAR PENGESAHAN**

**PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST**

Oleh :
MARCHEL ADIAS PRADANA
NPM. 21081010084

Telah dipertahankan dihadapan dan diterima oleh Tim Penguji Skripsi Prodi xxxxxxx Fakulktas Ilmu Komputer Universitas Pembangunan Nasional Veteran Jawa Timur Pada tanggal.....

| | | |
|---|---|---|
| Dr. Ir. I Gede Susrama Mas , ST. MT. IPU<br>NIP. xxxxxxxx xxxxxx x xxx | …… | (Pembimbing I) |
| Eva Yulia Puspaningrum, S.Kom., M.Kom NIP.<br>xxxxxxxx xxxxxx x xxx | …… | (Pembimbing II) |
| Nama Dosen<br>NIP/NPT | …… | (Pembimbing III)<br>*(Opsional/Tambahan)* |
| Nama Dosen<br>NIP/NPT | …… | (Ketua Penguji) |
| Nama Dosen<br>NIP/NPT | …… | (Penguji I) |

Mengetahui,
Dekan Fakultas Ilmu Komputer

Prof. Dr. Ir. Novirina Hendrasarie, MT
NIP. 19681126 199403 2 001

*Halaman ini sengaja dikosongkan*

## SURAT PERNYATAAN ORISINALITAS

Yang bertandatangan di bawah ini:

Nama Mahasiswa : MARCHEL ADIAS PRADANA
Program Studi : Informatika
Dosen Pembimbing : -

dengan ini menyatakan bahwa isi sebagian maupun keseluruhan disertasi dengan judul:

**PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST**

adalah benar-benar hasil karya intelektual mandiri, diselesaikan tanpa menggunakan bahan-bahan yang tidak diizinkan dan bukan merupakan karya pihak lain yang saya akui sebagai karya sendiri. Semua referensi yang dikutip maupun dirujuk telah ditulis secara lengkap pada daftar pustaka. Apabila ternyata pernyataan ini tidak benar, saya bersedia menerima sanksi sesuai peraturan yang berlaku.

Surabaya, ……
Yang Membuat Pernyataan,

MARCHEL ADIAS PRADANA
NPM. 21081010084

*Halaman ini sengaja dikosongkan*

# ABSTRAK


| | | |
|---|---|---|
| Nama Mahasiswa / NPM | : | MARCHEL ADIAS PRADANA / 21081010084 |
| Judul Skripsi | : | PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST |
| Dosen Pembimbing | : | 1. -<br>2. - |

-



**Kata kunci** : -

*Halaman ini sengaja dikosongkan*

## ABSTRACT

| | | |
|---|---|---|
| Student Name / NPM | : | Marchel Adias Pradana / 21081010084 |
| Thesis Title | : | PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST |
| Advisor | : | 1. -<br>2. - |

## ABSTRACT

-

**Keywords**: -

*Halaman ini sengaja dikosongkan*

# KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas segala rahmat, hidayah dan karunia-Nya kepada penulis sehingga skripsi dengan judul **"PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST"** dapat terselesaikan dengan baik.

Penulis mengucapkan terima kasih kepada - selaku Dosen Pembimbing utama yang telah meluangkan waktunya untuk memberikan bimbingan, nasehat serta motivasi kepada penulis. Dan penulis juga banyak menerima bantuan dari berbagai pihak, baik itu berupa moril, spiritual maupun materiil. Untuk itu penulis mengucapkan terima kasih kepada:

1. Ibu/Bapak..... selaku Dekan Fakultas Ilmu Komputer Universitas Pembangunan Nasional "Veteran" Jawa Timur.
2. Ibu/Bapak ……. selaku Ketua Program Studi xxxx Fakultas Ilmu Sosial Dan Ilmu Komputer Universitas Pembangunan Nasional " Veteran " Jawa Timur.
3. Dosen-dosen Program Studi … dst..

Penulis menyadari bahwa di dalam penyusunan skripsi ini banyak terdapat kekurangan. Untuk itu kritik dan saran yang membangun dari semua pihak sangat diharapkan demi kesempurnaan penulisan skripsi ini. Akhirnya, dengan segala keterbatasan yang penulis miliki semoga laporan ini dapat bermanfaat bagi semua pihak umumnya dan penulis pada khususnya.

Surabaya,_____

Penulis

*Halaman ini sengaja dikosongkan*

# DAFTAR ISI

# DAFTAR GAMBAR

*Halaman ini sengaja dikosongkan*

# DAFTAR TABEL

*Halaman ini sengaja dikosongkan*

# DAFTAR NOTASI

| | | |
|---|---|---|
| I | : | Intensitas |
| $W_R$ | : | *weight factor* |
| H | : | *hue* |
| S | : | *saturation* |
| V | : | *value* |
| *dst* | : | Gambar akumulator |
| *scr* | : | Gambar Input |
| *F* | : | *Foreground* |
| *B* | : | *Background* |
| $f$ | : | *Frame* |
| *SE* | : | *Structuring Element* |
| *ECD* | : | *Equivalent Circular Diameter* |
| b | : | bias |
| $WED(fi \rightarrow ,prototype)$ | : | bobot *euclidian distance* antara vektor fitur $fi \rightarrow$ |
| $f(x)$ | : | Fungsi Vektor Masukan |
| $d(x', x)$ | : | jarak di antara data uji z ke setiap vector data latih |
| $K(x, y)$ | : | fungsi kernel linear |
| $G(x)$ | : | fungsi Gaussian satu dimensi |
| $\sigma$ | : | standard deviasi dari distribusi |
| $G(x, y)$ | : | fungsi Gaussian dua dimensi |

*Halaman ini sengaja dikosongkan*

# BAB I
# PENDAHULUAN

## 1.1. Latar Belakang

Penyakit jantung adalah istilah yang umum digunakan pada seseorang yang mengalami disfungsi pada kerja fungsi jantung. Terdapat banyak jenis nama dari penyakit jantung seperti jantung koroner, kardiovaskuler, bahkan serangan jantung itu sendiri (Utomo & Mesran, 2020). Penyakit jantung mengambil peran sebagai penyebab utama kematian di Indonesia. Menurut data dari *World Health Organization* (WHO) pada tahun 2019, penyakit kardiovasular dapat mencapai 31% dari total penyebab kematian di seluruh belahan dunia, angka kematiannya tersendiri diperkirakan hingga 17,7 juta juta jiwa pada tiap tahunnya. Angka kematian akibat penyakit kardiovaskular diprediksikan mengalami pertumbuhan tiap tahunnya hingga pada tahun 2030 diperkirakan dapat mencapai 23,3 juta kematian (Medayati dkk., 2018). Sedangkan laporan untuk angka kematian yang diakibatkan oleh penyakit jantung di Indonesia menduduki presentase sebesar 12,9% berdasarkan data Kemenkes RI pada tahun 2017 (Hastuti & Mulyani, 2019). Dengan memperhatikan hal ini, maka diperlukannya langkah deteksi dini pada penyakit jantung yang bertujuan untuk mengurangi angka kematian yang disebabkan oleh penyakit jantung di Indonesia.

Penyakit jantung pada dasarnya dapat dicegah dengan menerapkan gaya hidup yang sehat dengan pola makan teratur. Selain itu tak lepas juga deteksi dini diperlukan untuk mencegah kematian pada seorang penderitanya. Cara yang dapat dikatakan efektif untuk saat ini adalah dengan memanfaatkan teknologi informasi sebagai diagnosa dini penyakit jantung, hal ini merupakan situasi yang menantang disebabkan saling ketergantungan dari beberapa faktor (Jain dkk., 2019). Salah satu cara efektif yang dapat digunakan adalah *data mining*, diartikan sebagai alat yang ampuh dalam menemukan pola dan hubungan dalam kumpulan data yang besar (Alhasani dkk., 2023). *Data mining* merupakan proses dengan menggunaan teknik matematika, stastitika, kecerdasan buatan, serta *machine learning* dalam mengekstraksi dan mengidentifikasi informasi bermanfaat dan pengetahuan terkait data dalam jumlah yang cukup banyak (Romli & Firana Puspita Dewi, 2021). Teknik dalam *data mining* sangat bervariasi, diantaranya yang paling sering digunakan yakni klustering, asosiasi,

estimasi, dan klasifiasi (Pristyanto, 2019). Dalam bidang medis, teknik klasifikasi banyak digunakan dalam diagnosis dan analisis guna membantu dalam pengambilan keputusan, untuk itu teknik yang digunakan pada penelitian ini adalah klasifikasi dalam identifikai penyakit jantung.

Algoritma klasifikasi yang sering digunakan terkait *machine learning*, beberapa diantaranya adalah *Decision Tree* (DT), *Neural Network* (NN), *K-Nearest Neighbor* (KNN), *Naïve Bayes* (NB), dan *Support Vector Machine* (SVM) (Pristyanto, 2019). Untuk penelitian ini dipilih algoritma *Extrerme Gradient Boost* (XGBoost) dalam klasifikai penyakit jantung. Algoritma XGBoost adalah salah satu model ensemble dari banyaknya algoritma *machine learning classifier* yang didasari dengan pohon keputusan dengan *Gadient Boost* sebagai inti (Abdurrahman dkk., 2022). Pemilihan model XGBoost ini dikarenakan memiliki ketahanan terhadap *outlier* yang lebih besar, dengan waktu komputasi yang cepat untuk mendapatkan hasil yang akurat (Kurnia dkk., 2023).

Data yang dapat digunakan faktanya tidak selalu dapat digunakan secara langsung, terutama pada bidang kesehatan sering kali ditemukan kasus data yang tidak seimbang (*imbalance*) dikarenakan data yang memiliki hasil negatif biasanya memiliki kelas mayoritas dibandingkan data dengan hasil positif sebagai kelas minoritas sehingga dapat menyebabkan kesalahan dalam klasifikasi (Mutmainah, 2021). Untuk mengatasi ketidakseimbangan kelas pada data, dapat dilakukan dengan pendekatan pada algoritma dan pendekatan pada *data processsing*. Pendekatan pada algoritma berarti dengan fokus pada perbaikan algoritma yang digunakan tanpa mengubah data. Sedangkan untuk penelitian ini menggunakan pendekatan kedua yakni pada *data processing* dengan menggunakan teknik *resampling* untuk menyeimbangkan distribusi pada level data. Teknik *resampling* dikategorikan kedalam tiga jenis, yaitu *oversampling, undersampling,* dan *hybridsampling* atau kombinasi dari keduanya (Sir & Soepranoto, 2022).

Berdasakan penelitian terdahulu yang telah dilakukan oleh Mengyin Lin dkk, pada tahun 2023 yang mengangkat judul penelitian "Detection of Ionospheric Scintillation Based on XGBoost Model Improved by SMOTE-ENN Technique", dilakukan kombinasi dengan mengimprovisasi algoritma *Exteme Gradient Boosting* (XGBoost) dengan menggunakan teknik *Synthetic Minority Oversampling Technique and Edited Nearest Neighbor* (SMOTE-ENN) untuk mendeteksi kejadian yang tidak

seimbang terkait sintilasi ionosfer lemah, sedang, dan kuat. Hasil pengujian untuk metode yang ditingkatkan menunjukkan peningkatan yang signifikan dalam indikator evaluasi, dengan nilai recall relatif stabil di atas 90%, dan F1 skor lebih dari 92% (M. Lin dkk., 2021).

Berdasarkan latar belakang serta literatur yang telak dilakukan oleh , penelitian ini mengangkat judul "PENGGUNAAN TEKNIK SMOTE-ENN UNTUK BALANCING DATA DALAM KLASIFIKASI PENYAKIT JANTUNG DENGAN METODE XGBOOST". Data yang digunakan pada penelitian ini adalah data sekunder berasal dari situs Kaggle dengan total data bejumlah 445.132 data. Adapun tahap – tahap pada penelitian ini meliputi *preprocessing* data, selanjutnya dilakukan teknik *hybridsampling Synthetic Minority Oversampling Technique Edited Nearest Neighbor* (SMOTEENN) yang merupakan kombinasi dari teknik *oversampling Synthetic Minority Oversampling Technique* (SMOTE) dan teknik *undersampling Edited Nearest Neighbor* (ENN), yang akan dibandingkan dari ketiga teknik tersebut dengan tanpa teknik *resampling* apapun pada model algoritma *Extreme Gradient Boosting* (XGBOOST). Setelah itu diakhiri dengan perhitungan metrik evaluasi menggunakan *accuracy, sensitivity, specifity,* dan *g-mean* untuk mengukur kinerja model terhadap klasifikasi data. Tujuan akhir dari penelitian ini adalah untuk menganalisis sejauh mana penggunaan teknik *resampling* untuk *balancing* data dapat meningkatkan perfoma model XGBoost dalam mengatasi ketidakseimbangan pada data kelas penyakit jantung.

### 1.2. Rumusan Masalah

Bedasarkan dari latar belakang sebelumnya, penelitian ini menguraikan beberapa rumusan masalah sebagai berikut:

1. Bagaimana performa algoritma XGBoost sebagai model untuk mengatasi ketidakseimbangan data pada tiap kelas tanpa menggunakan teknik *hybridsampling* pada dataset penyakit jantung?
2. Apakah kombinasi teknik *hybridsampling* SMOTEENN untuk mengatasi ketidakseimbangan pada dataset penyakit jantung dapat meningkatkan performa dari model klasifikasi XGBoost?

3. Pengorbanan apa yang perlu untuk dipertimbangkan terkait penerapan teknik *hybridsampling* SMOTEENN dalam model XGBoost yang digunakan untuk mengatasi rasio ketidakseimbangan kelas dalam klasifikasi pada data penyakit jantung?

### 1.3. Tujuan Penelitian

Berdasarkan perumusan masalah yang dijelaskan sebelumnya, tujuan yang ingin dicapai pada penelitian ini adalah sebagai berikut ini:

1. Menguji performa algoritma XGBoost sebagai model untuk mengatasi ketidakseimbangan tanpa menggunakan teknik *hybridsampling* pada data penyakit jantung.
2. Mengevaluasi apakah kombinasi dari teknik *hybridsampling* pada dataset penyakit jantung dapat meningkatkan performa dari model klasifikasi XGBoost.
3. Menganalisis apakah terdapat pengorbanan yang perlu dipertimbangkan terkait penerapan teknik *hybridsampling* dengan SMOTEENN dalam model XGBoost untuk mengatasi rasio ketidakseimbangan kelas dalam klasifikasi pada data penyakit jantung.

### 1.4. Batasan Masalah

Penelitian ini memiliki beberapa barasan masalah yaitu meliputi:

1. Dataset yang digunakan untuk penelitian ini meggunakan dataset sekunder yang didapat dari situs Kaggle dengan format CSV yang memiliki data cukup besar, yakni berjumlah 445.132 data.
2. Algoritma yang digunakan klasifikasi penyakit jantung pada penelitian ini adalah Extreme Gradient Boosting (XGBoost).
3. Penelitian ini menggunakan teknik *hybridsampling* SMOTEENN dengan kombinasi antara teknik *oversampling* SMOTE dan *undersampling* ENN, dalam mengatasi kelas yang tidak seimbang.
4. Pengukuran kinerja model menggunakan matriks evaluasi dalam mengukur seberapa baik performa model yang terdiri dari akurasi untuk mengukur prediksi benar, sensitivity sebagai recall kelas positif, specifity sebagai recall kelas negatif, dan G-mean untuk mengevaluasi dataset tidak seimbang.

## 1.5. Manfaat Penelitian

Penelitian yang dilakukan diharapkan mampu memberikan manfaat, yaitu:

1. Hasil dari uji coba dengan mengkombinasikan antara XGBoost dengan metode *hybridsampling*, penelitian ini diharapkan mampu menambahkan wawasan baru terkait seberapa efektif pendekatan yang telah digunakan dalam meningkatkan akurasi pada data yang memiliki kelas yang tidak seimbang.
2. Penelitian ini juga diharapkan memberikan referensi untuk peneliti maupun pengembang selanjutnya dengan bidang terkait. Dengan tujuan dan pemilihan metode yang baik, penelitian ini dapat menjadi acuan untuk penggunaan klasifikasi terhadap penyakit jantung.

*Halaman ini sengaja dikosongkan*

# BAB II
# TINJAUN PUSTAKA

## 2.1. Penelitian Terdahulu

Penelitian dilakukan tidak lepas dari acuan terhadap penelitian yang telah ada sebelumnya. Acuan ini diharapkan menjadi dasar dan pembanding dalam mendukung penelitian yang akan dilakukan serta untuk menghindari dari duplikasi. Melalui tinjauan pustaka terhadap penelitian sebelumnya, penulis mengevaluasi hubungan dan relevansi terhadap penelitian yang akan dilakukan. Penelitian-penelitian yang telah dilakukan oleh para peneliti sebelumnya terkait dengan penelitian yang akan dilakukan diantaranya sebagai berikut:

1. Pada penelitian yang dilakukan oleh (Hairani & Priyanto, 2023) dengan judul "A New Approach of Hybrid Sampling SMOTE and ENN to the Accuracy of Machine Learning Methods on Unbalanced Diabetes Disease Data". Penelitian ini membahas perbandingan penggunaan algoritma *Support Vector Machine* (SVM) dan *Random Forest Classification* melalui pendekatan *balancing* data SMOTE-ENN pada dataset penyakit diabetes. Hasil penelitian ini menggunakan *k-fold cross validation* 10 menghasilkan komparasi terbaik dengan *Random Forest* melalui pendekatan SMOTE-ENN pada *accuracy* 95.8%, *sensitivity* 98.3%, dan *specifity* 92.5%.
2. Pada penelitian yang dilakukan oleh (M. Lin dkk., 2021) dengan judul "Detection of Ionospheric Scintillation Based on XGBoost Model Improved by SMOTE-ENN Technique". Peneliti melakukan kombinasi dengan mengimprovisasi algoritma *Exteme Gradient Boosting* (XGBoost) menggunakan teknik *Synthetic Minority Oversampling Technique and Edited Nearest Neighbor* (SMOTE-ENN) untuk mendeteksi kejadian yang tidak seimbang terkait sintilasi ionosfer lemah, sedang, dan kuat. Hasil pengujian untuk metode yang ditingkatkan menunjukkan peningkatan yang signifikan dalam indikator evaluasi, dengan nilai *recall* relatif stabil di atas 90%, dan *f1-score* lebih dari 92% . Dengan beragam testing pada *imbalance* data, terdapat peningkatan sekitar 10% sampai 20% pada nilai *recall* dan peningkatan sebesar 6% sampai 11% pada *f1-score* dengan peningkatan akurasi pada testing dengan rentang 90.42% hingga 96.04%.

3. Pada penelitian oleh (Han & Joe, 2024) dengan judul penelitian "Enhancing Machine Learning Models Through PCA, SMOTE-ENN, and Stochastic Weighted Averaging". Penelitian ini memprediksi hasil kelangsungan hidup pada kecelakaan kritis dengan studi membahas beberapa keterbatasan metode yang ada, termasuk *imbalance* data, kurangnya penekanan pada *hyperparmeter tuning*, dan kecenderungan untuk *overfitting*. Dengan mengintegrasikan *Principal Component Analysis* (PCA), *hyperparameter tuning*, dan metode *resampling*, serta menggabungkan *Edited Nearest Neighbors* (ENN) dengan *Synthetic Minority Oversampling Technique* (SMOTE). Terdapat tujuh model yang dibandingkan secara langsung untuk untuk menghasilkan prediksi yaitu *Logistic Regression*, *Support Vector Machine*, *KNN*, *Random Forest*, *XGBoost*, *LightGBM*, dan *CatBoost.* Penelitian juga menerapkan Stochastic Weighted Averaging (SWA) untuk mengurangi overfitting dan meningkatkan generalisasi. Akurasi meningkat dari 91.97% menjadi 94.89% setelah SWA diterapkan dengan model akhir menunjukkan kinerja yang sangat baik, untuk nilai Area Under the Curve (AUC) dan Average Precision (AP) yang keduanya mencapai 0.98, yang menunjukkan akurasi dan presisi yang tinggi.
4. Pada penelitian oleh (Kartina Diah Kusuma Wardani & Memen Akbar, 2023) dengan judul penelitian "*Diabetes Risk Prediction using Feature Importance Extreme Gradient Boosting* (XGBoost)". Penelitian ini membahas terkait deteksi dini pada penyakit diabetes menggunakan algoritma model XGBoos dengan menggunakan dataset dari *UCI Machine Learning* dengan total 520 rekord dan 16 atribut. Prediksi model XGBoost yang ditampilkan dalam bentuk *tree* dengan membandingkan dua penggunaan fitur yaitu 16 atribut dan 10 atribut terbaik. Model akurasi menghasilkan 98.71% untuk *accuracy* dan 98.18% untuk *f1-score* pada 16 atribut, sementara untuk 10 atribut menghasilkan model dengan 98.72% *accuracy.*
5. Pada penelitian lain yang dilakukan oleh (Aditya Gumilar dkk., 2022) dengan judul penelitian "Performance Analysis of Hybrid Machine Learning Methods on Imbalanced Data (Rainfall Classification)". Penelitian mengusulkan beberapa metode dalam menganalisis kinerja metode hybrid pada *machine learning* menggunakan *voting* dan *stacking* pada data curah hujan kota Bandung tahun 2005 sampai dengan tahun 2021. Kedua metode tersebut dibentuk dari lima metode

klasifikasi yaitu *Logistic Regression, Support Vector Machine, Random Forest, Artificial Neural Network*, dan *eXtreme Gradient Boosting*. Hasil penelitian ini menunjukkan bahwa dengan menggabungkan lima metode *machine learning* pada dataset yang tidak seimbang terbukti bahwa algoritma *voting* memiliki kelemahan pada data tidak seimbang, sementara algoritma *stacking* memperoleh nila *accuracy* sebesar 99.60%. Setelah itu pengujian dibuktikan kembali dengan menambahkan teknik *oversampling* SMOTE yang menghasilkan akurasi yang meningkat hingga 99.71%.

## 2.2. Penyakit Jantung

Penyakit Jantung dapat disebabkan oleh berbagai resiko yang tidak dapat diubah seperti usia, jenis kelamin, dan genetik. Selain itu juga terdapat faktor risiko lain yang dapat diubah seperti kebiasaan merokok, hipertensi, aktifitas fisik, obesitas, dislipdemia, diabetes mellitus, stres, alkohol, dan juga diet yang tidak sehat (Naomi dkk., 2021).

Seseorang yang mengalami penyakit jantung seringkali dapat terjadi karena rusaknya sel otot-otot pada organ jantung yang bertugas memompa aliran darah keseluruh tubuh. Selain itu, kurangnya oksigen yang dapat dibawa darah pada pembuluh darah kedalam jantung juga dapat mengakibatkan kejang pada otot jantung sehingga menyebabkan kegagalan dalam memompa darah pada organ jantung, hal inilah yang dapat menyebabkan organ jantung tidak dapat melaksanakan fungsinya dengan baik (Wahyudi & Hartati, 2017). Terdapat beberapa jenis penyakit jantung antara lain sebagai berikut:

1. Penyakit Jantung Koroner

   Penyakit Jantung Koroner (PJK) penyakit pada arteri jantung yang diakibatkan oleh penyempitan atau tersumbatnya pembuluh darah yang mengakibatkan kurangnya suplai darah yang diberikan kepada otot jantung. PJK ditandai dengan adanya rasa nyeri yang ada di dada atau juga terasa tertekan saat berjalan pada area dada (Saragih, 2020).

2. Gagal Jantung

   Gagal jantung merupakan sindrom klinis yang kompleks akibat gangguan pada struktur atau fungsi ventrikel, baik dalam proses pengisian maupun pengeluaran darah. Gejala utama yang ditimbulkan adalah sesak napas dan rasa

lelah, yang sering menyebabkan keterbatasan aktivitas fisik. Selain itu, kondisi ini dapat memicu retensi cairan, yang berujung pada kongesti paru atau pembengkakan (edema) di bagian tubuh tertentu, terutama perifer.Aritmia Jantung (Amanah & Herawati, 2022).

3. Penyakit Jantung Bawaan

   Penyakit Jantung Bawaan (PJB) dikenal sebagai kelainan jantung sejak lahir, merupakan istilah umum dalam menggambarkan gangguan pada struktur jantung dan pembuluh darah besar yang terjadi dari lahir. Kondisi ini menjadi salah satu kelainan bawaan yang paling umum ditemukan dan menjadi penyebab utama kematian di antara berbagai jenis kelainan bawaan lainnya. Secara umum, PJB terjadi dengan angka kejadian sekitar 8 hingga 10 kasus per 1.000 kelahiran hidup (Ain dkk., 2015).

## 2.3. Data Mining

*Data mining* diartikan sebagai metode untuk pencarian, pemilihan, dan pemodelan data dalam jumlah besar yang menghasilkan pola – pola tersembunyi yang diamati untuk mendapatkan pembuktian untuk menyimpulkan cara baru pada dan bermanfaat bagi pemilik data (Amin dkk., 2022). Metode yang digunakan dalam menganalisis data umumnya menggunakan *Knowledge Discovery in Database* (KDD) yang meliputi beberapa proses sebagai berikut (Ameliana dkk., 2024):

1. *Data Selection* : sebagai tahap awal yang merupakan proses pemilihan dan penggalian informasi yang dibutuhkan, proses ini menghasilkan data yang telah diseleksi dengan mengacu pada data yang relevan dan sesuai dengan kebutuhan selanjutnya.
2. *Preprocessing* : tahap kedua ini akan membersihkan semua data yang tidak relevan, data yang hilang (*missing value*), dan duplikasi data. Tahapan ini menghasilkan data bersih agar seleksi data lebih terstruktur dalam memulai *data mining*.
3. *Transformation* : tahapan selanjutnya bertujuan mengubah data yang telah diseleksi untuk nantinya memudahkan dalam proses *data mining.* Hasil dari proses ini bergantung pada format data yang ada serta bagaimana jenis pola informasi yang akan dicari nantinya.

4. *Data Mining* : proses ini mencari pola dan informasi didalam data yang dilakukan dengan teknik atau metode tertentu. Hasil dari proses ini adalah pola yang ditampilkan sesuai dengan teknik yang digunakan.
5. *Evaluation* : pada tahap akhir akan dilihat hasil performa serta pola atau informasi dan pengetahuan yang ditampilkan dalam bentuk yang dapat dimengerti oleh pihak yang membutuhkan.

### 2.4. Seleksi Fitur Chi-Square

Seleksi fitur bertujuan untuk melakukan pengurangan dimensi dari suatu data dengan menghapus kolom yang dirasa tidak penting atau kurang memiliki hubungan dengan target klasifikasi sehingga pengklasifikasian dapat lebih akurat dan efektif (Amrullah dkk., 2020). *Chi-Square* testing merupakan metode pada ilmu statistika pengujian korelasi antar suatu variabel dengan target yang akan dievaluasi pada data diskrit untuk melihat apakah variabel tersebut saling berkaitan atau tidak (Suharno dkk., 2017). Berikut merupakan perhitungan nilai *chi-square* yang terdapat pada persamaan 2.1.

$$X^2 = \sum \frac{(O-E)^2}{E} \tag{2.1}$$

Keterangan:

$X^2$ : nilai *chi-square*

$O$ : frekuensi observasi

$E$ : frekuensi ekspektasi

Nilai dari $E$ didapat dari penggunaan bantuan tabel kontigensi dengan ukuran baris dikalikan kolom dapat dilihat pada Tabel 2.1

**Tabel 2.1 Tabel Kontigensi**

| | $Kolom_1$ | $Kolom_2$ | ... | $Kolom_j$ | Total |
|---|---|---|---|---|---|
| $Baris_1$ | $O_{11}$ | $O_{12}$ | ... | $O_{1j}$ | $T_{b1}$ |
| $Baris_2$ | $O_{21}$ | $O_{22}$ | ... | $O_{2j}$ | $T_{b2}$ |
| ... | ... | ... | ... | ... | ... |
| $Baris_i$ | $O_{i1}$ | $O_{i2}$ | ... | $O_{ij}$ | $T_{bi}$ |
| Total | $T_{k1}$ | $T_{k2}$ | ... | $T_{kj}$ | $N$ |

Nilai ekspektasi kemudian dihitung dengan persamaan 2.2.

$$E = \frac{T_{bi}\, x\, T_{ki}}{N} \tag{2.2}$$

Keterangan:

$E$ : frekuensi ekspektasi

$T_{bi}$ : total nilai baris ke-i

$T_{ki}$ : total nilai kolom ke-i

$N$ : total keseluruhan nilai observasi

Setelah semua nilai ekspektasi dihitung, selanjutnya adalah memasukkan ke dalam rumus *chi-square* satu persatu untuk semua sel dalam tabel kontingensi. Kemudian tentukan derajat kebebasan dihitung untuk menentukan distribusi *chi-square* yang relevan melalui persamaan 2.3.

$$dk = (T_b - 1)\, x\, (T_k - 1) \tag{2.3}$$

Keterangan:

$dk$ : derajat kebebasan

$T_b$ : jumlah baris

$T_k$ : jumlah kolom

Dalam uji *chi-square*, hipotesis berfungsi untuk menentukan hasil pengujian untuk menjawab apakah fitur tetap dipertahankan atau tidak. Formula hipotesis pada *chi-square* adalah:

$H_o$ : Fitur yang diuji tidak memiliki hubungan dengan target.

$H_1$ : Fitur yang diuji memiliki hubungan signifikan dengan target.

Selanjutnya menentukan niai signifikansi (α) = 0.05 yang berarti (tingkat kepercayaan 95%), maka jika hasil uji berada dalam 5% distribusi teratas, dengan kata lain ($H_o$) ditolak. Langkah terakhir menghitung nilai $X^2$ dibandingkan dengan nilai kritis pada tabel pada tabel distribusi dengan signifikansi (α) dan derajat kebebasan ($dk$) dengan pengambilan keputusan sebagai berikut:

- Jika $X^2_{hitung} \leq X^2_{tabel}$, maka tidak cukup bukti untuk menolak ($H_o$) yang berarti tidak ada hubungan signifikan antara fitur dan target sehingga fitur dianggap tidak relevan.
- Jika $X^2_{hitung} > X^2_{tabel}$, maka cukup untuk menolak ($H_o$) yang diartikan sebagai adanya hubungan signifikan antara fitur dan target sehingga fitur dapat dipertahankan.

## 2.5. Imbalance Data

Istilah dari ketidakseimbangan data atau *imbalance data* merupakan kondisi dimana dataset memiliki jumlah data yang tidak seimbang antar kelas, sehingga menyebabkan adanya kelas mayoritas dan kelas minoritas. Sebagai kasus yang seringkali mengalami banyak ketidakseimbangan data yakni pada data kesehatan dimana terdapat kelas mayoritas untuk negatif dan minoritas untuk hasil yang positif sehingga merupakan faktor yang paling umum untuk menyebabkan *imbalance data* (Rifqi Fitriadi & Deni Mahdiana, 2023).

Pada sebagian besar dari model algoritma *data mining* dapat bekerja dengan performa terbaiknya pada saat jumlah sampel setiap kelas relatif seimbang. Ketidakseimbangan pada dataset yang digunakan untuk *data mining* merupakan masalah yang serius dan harus ditangani (Gumelar & Al Fatta, 2023). Terdapat dua pendekatan untuk menanganni masalah *imbalance data*, yakni pendekatan pada level data dan pendekatan pada level algoritma (Sir & Soepranoto, 2022):

1. Pendekatan level data

   Metode pendekatan pertama adalah pada level data yang fokus pada distribusi kelas menjadi seimbang menggunakan berbagai macam teknik resampling pada level data. Metode ini dibagi menjadi tiga kategori, yaitu teknik *oversampling* untuk penambahan data pada kelas minoritas, teknik *undersampling* untuk penambahan data pada kelas mayoritas, serta kombinasi keduanya.

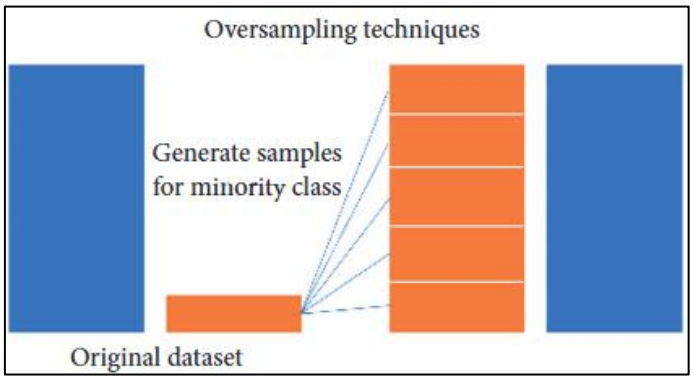
2. Pendekatan level algoritma

   Metode pendekatan kedua ialah pada level algoritma yang lebih difokuskan untuk perbaikan dari model algoritma yang digunakan tanpa mengubah penyebaran data. Pendekatan ini diharapkan dapa mengurangi bias terhadap kelompok mayoritas untuk membuat model prediksi yang lebih akurat.

Masalah *imbalance data* lebih sering diatasi menggunakan pendekatan level data baik dengan teknik *oversampling*, *undersampling*, maupun kombinasi keduanya (*hybrid*). Namun perlu diingat bahwa *resampling* hanya dilakukan pada data pelatihan, bukan data validasi atau pengujian. Pendekatan-pendekatan ini bertujuan untuk menyeimbangkan distribusi data antar kelas, sehingga performa model prediksi dapat ditingkatkan.

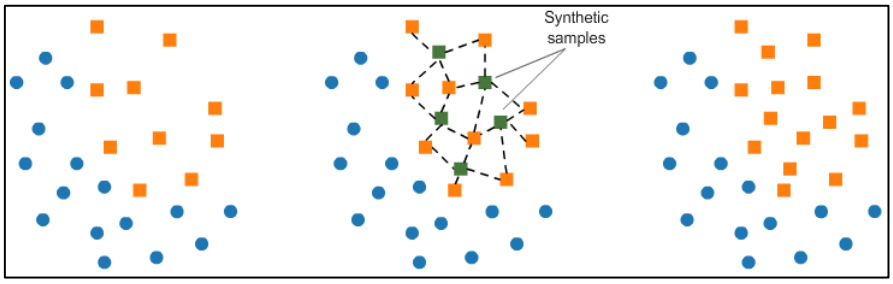
### 2.5.1 SMOTE (*Synthetic Minority Oversampling Technique*)

SMOTE atau *Synthetic Minority Oversampling Technique* diperkenalkan pertama kali oleh Chawla dkk., merupakan teknik yang biasa digunakan dalam metode *oversampling* dengan tujuan untuk membuat distribusi kelas lebih proporsional sehingga model dapat menentukan hasil berdasarkan tingkat keseimbangan antar kelas yang diilustrasikan pada gambar 2.1 (Le dkk., 2019).




**Gambar 2.1 Ilustrasi Teknik *Oversampling***

SMOTE merupakan metode dengan memperbanyak kelas minoritas melalui duplikasi data menggunakan data sintetik. Cara kerja *oversampling* yang dilakukan oleh SMOTE adalah mengambil *instance* pada kelas minoritas, dilanjut dengan menemukan *k-nearest neighbor* pada setiap *instance*, selanjutnya *instance* sintetik direplikasi pada kelas minoritas. Dengan kata lain, masalah *overfiting* yang berlebihan dapat diatasi (Sutoyo & Fadlurrahman, 2020).




**Gambar 2.2 Ilustrasi SMOTE**

Proses yang dilakukan SMOTE yang ditampilkan pada gambar 2.2 dimulai dengan menentukan kelas minoritas, yakni kelas yang memiliki jumlah sampel lebih sedikit dibandingkan kelas mayoritas. Selanjutnya mengidentifikasi jumlah tetangga terdekat dengan metode *k-nearest neighbor*,

kemudian mengukur jarak dari tiap data minoritas menggunakan e*uclidean distance* melalui persamaan 2.4 berikut.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (2.4)$$

Keterangan :

$d(x,y)$ : jarak data *x* dengan data *y*

$n$ : jumlah dimensi atribut

$x_i$ : atribut ke-i pada data *x*

$y_i$ : atribut ke-i pada data y

Apabila hasil dari pengukuran *euclidean distance* ditemukan pada tiap data, langkah setelahnya adalah dengan memilih nilai terkecil pada tetangga (*k*) yang telah ditentukan. Kemudian membuat data sintetik melalui persamaan 2.5 berikut.

$$x_{baru} = x_i + (x_{knn} - x_i)\delta \quad (2.5)$$

Keterangan :

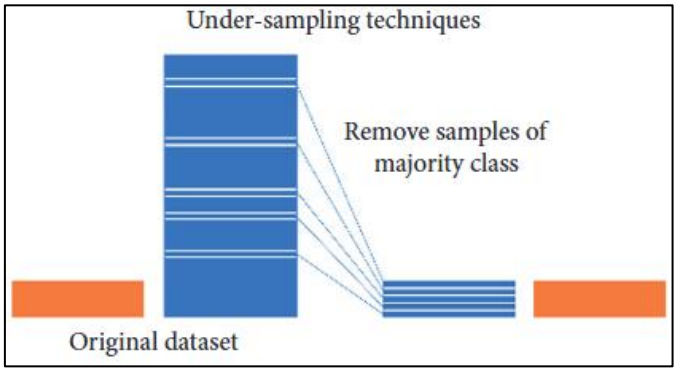
$x_{baru}$ : data sintetik yang dihasilkan

$x_{knn}$ : tetangga terdekat dari $x_i$

$\delta$ : nilai acak antara 0 dan 1

### 2.5.2 ENNs (*Edited Nearest Neighbors*)

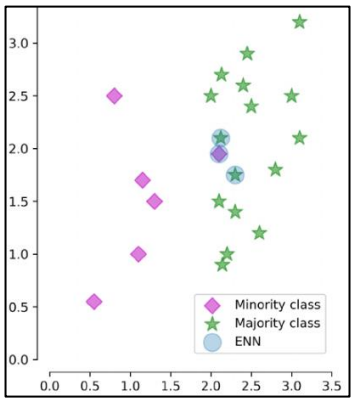
*Edited Nearest Neighbors* (ENN) merupakan salah satu dari metode *undersampling* yang bertujuan menyeimbangkan distribusi data dengan menghilangkan data di kelas mayoritas (Le dkk., 2019). Metode *undersampling* digambarkan pada ilustrasi pada gambar 2.3.




**Gambar 2.3 Ilustrasi Teknik *Undersampling***

ENN diperkenalkan oleh Wilson pada penelitiannya di tahun 1972, metode ini menghitung tetangga terdekat (secara defaut menggunakan nilai *k*=3) pada setiap sampel data. Jika suatu sampel berasal dari kelas mayoritas

dan salah diklasifikasi oleh tiga tetangga terdekatnya, maka sampel tersebut dihapus dari dataset. Sebaliknya, jika suatu sampel berasal dari kelas minoritas dan salah diklasifikasi oleh tiga tetangga terdekatnya, maka tiga sampel dari kelas mayoritas yang mengelilinginya akan dihapus. Dengan cara ini, ENN membantu mengurangi ketidakseimbangan data dan *noise* yang mungkin ada di kelas mayoritas, yang pada gilirannya dapat meningkatkan kinerja model (Imani & Arabnia, 2023).




**Gambar 2.4 Ilustrasi ENN**

Proses dari metode ENN diilustrasikan pada gambar 2.4 dimulai dengan menentukan *k-nearest neighbor* (*k*=3 secara default) berdasarkan jarak untuk setiap sampel data, dihitung dengan *euclidean distance* pada persamaan 2.6 berikut.

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2} \quad (2.6)$$

Keterangan :

$d(x,y)$ : jarak data *x* dengan data *y*

$n$ : jumlah dimensi atribut

$x_i$ : atribut ke-i pada data *x*

$y_i$ : atribut ke-i pada data y

Selanjutnya identifikasi label sampel dengan membandingkan dengan tetangganya dengan evaluasi:

- Jika sampel dari kelas mayoritas dan mayoritas tetangga terdekatnya berasal dari kelas minoritas (lebih dari separuh tetangga terdekat adalah dari kelas minoritas), maka sampel mayoritas tersebut dianggap salah klasifikasi dan dihapus dari dataset.

- Jika sampel dari kelas minoritas dan sampel minoritas dikelilingi oleh mayoritas tetangga yang salah klasifikasi, maka tetangga mayoritas yang menyebabkan kesalahan klasifikasi dihapus dari dataset.

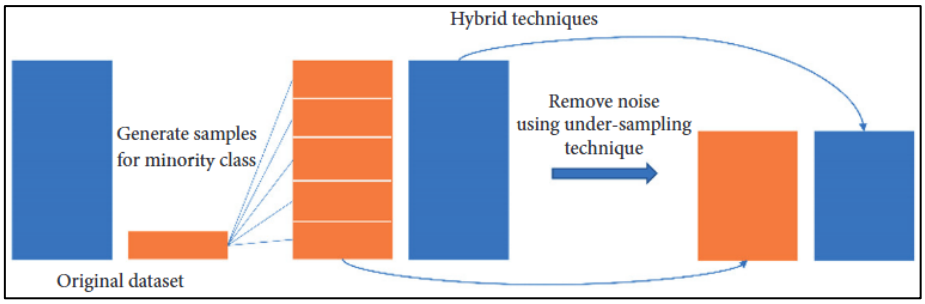
Proses tersebut akan terus berlanjut hingga semua data dalam dataset telah dievaluasi sehingga menghasilkan dataset yang lebih bersih dengan pengambilan keputusan lebih mulus. Berikut contoh penggambaran dari hasil teknik penghapusan dengan metode ENN melalui Tabel 2.2.

**Tabel 2.2 Contoh Penghapusan ENN**

| Sampel | Kelas | *knn* | Kelas *knn* | Keputusan |
|---|---|---|---|---|
| A | Mayoritas | B, C, D | Minoritas, Mayoritas, Mayoritas | Dihapus |
| B | Minoritas | A, C, E | Mayoritas, Mayoritas, Minoritas | Tetangga Dihapus |
| C | Mayoritas | A, B, E | Minoritas, Minoritas, Mayoritas | Dihapus |
| D | Minoritas | A, B, C | Mayoritas, Minoritas, Mayoritas | Tetangga Dihapus |
| E | Mayoritas | B, C, D | Minoritas, Minoritas, Minoritas | Dihapus |

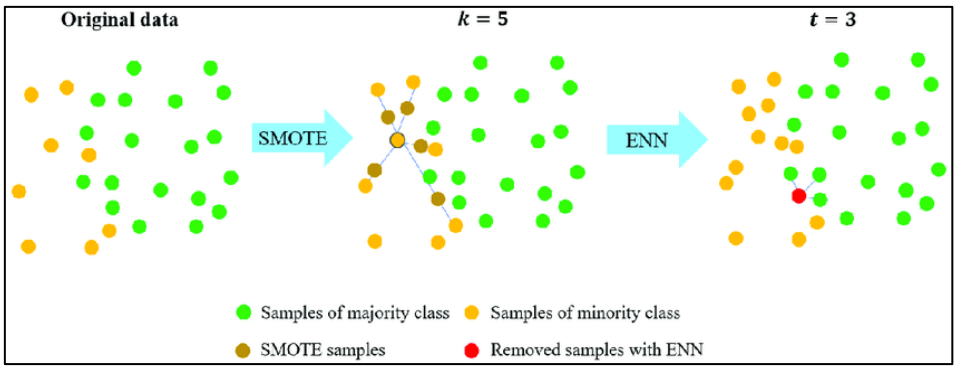
### 2.5.3 SMOTEENN

SMOTEENN merupakan penggabungan dari teknik *oversampling* SMOTE dengan teknik *undersampling* ENN, yang menawarkan perbaikan untuk mendapatkan hasil yang lebih unggul karena kombinasi dari pengambilan rasio *oversampling* dan rasio *undersampling*, menyempurnakan batasan keputusan, dan meningkatkan untuk mengklasifikasikan data (Imani & Arabnia, 2023). Ilustrasi yang digambarkan pada metode *hybrid* terdapat pada gambar 2.5 dibawah ini.




**Gambar 2.5 Ilustrasi Teknik *Hybridsampling***

Pada dasarnya, penggunaan kombinasi sama halnya dengan penggunaan dua metode secara berurutan. SMOTE untuk *oversampling* kelas minoritas dan ENN untuk *undersampling* kelas mayoritas. Gabungan ini bertujuan untuk meningkatkan representasi kelas minoritas sambil mengurangi noise pada kelas mayoritas, menciptakan dataset yang lebih seimbang dan lebih bersih.




**Gambar 2.6 Ilustrasi SMOTEENN**

Proses dalam metode SMOTEENN terlihat pada gambar 2.6 diatas, dimulai dengan menggunakan teknik SMOTE untuk *oversampling* untuk menambah sampel pada kelas minoritas. Proses ini akan menghasilkan data sintetis baru berdasarkan interpolasi antara data minoritas yang ada dan tetangga terdekatnya. Dengan cara ini, distribusi kelas menjadi lebih seimbang. Setelah dataset minoritas berhasil ditambah dengan data sampel sintetis, langkah berikutnya adalah menggunakan teknik *undersampling* untuk mengurangi noise pada kelas mayoritas menggunakan ENN (G.-M. Lin & Zeng, 2021).

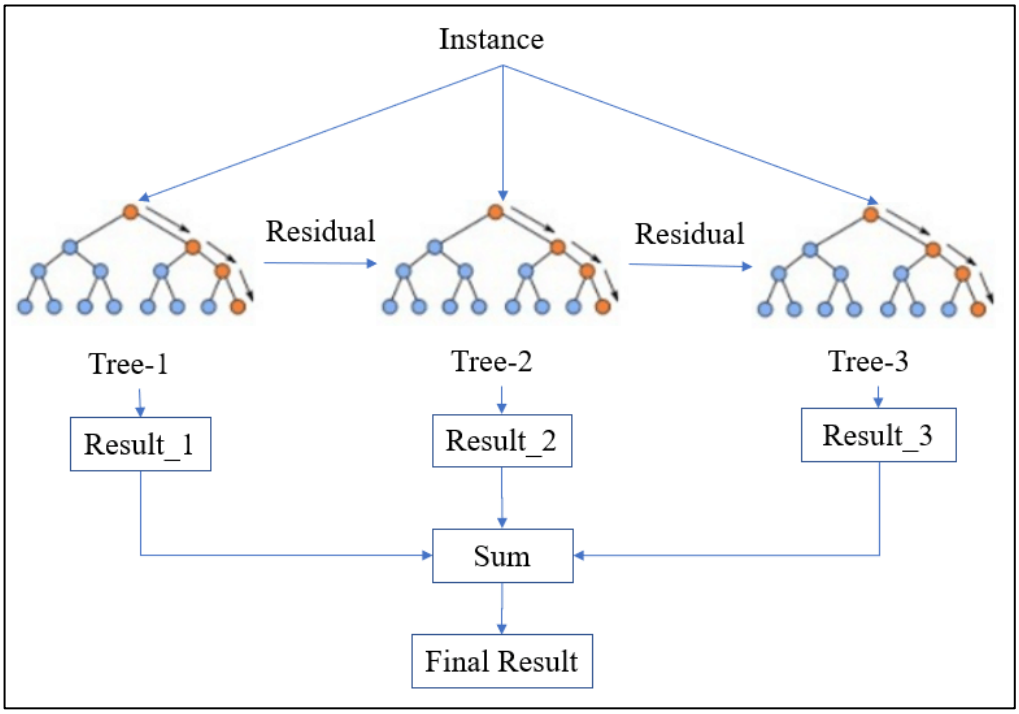
Setelah kedua proses dijalankan, maka SMOTEENN akan menghasilkan dataset yang lebih bersih dan seimbang dimana:

- Kelas minoritas telah memiliki lebih banyak sampel, meningkatkan representasi mereka dalam dataset.

- Kelas mayoritas telah dibersihkan dari sampel yang salah klasifikasi, mengurangi noise dan meningkatkan kualitas data.

### 2.5. XGBoost (*Extreme Gradient Boosting*)

*Extreme Gradient Boosting* atau biasa disebut dengan istilah XGBoost diperkenalkan oleh Chen Tianqi pada tahun 2016, yang menghadirkan kompleksitas komputasi rendah dengan akurasi yang cukup tinggi. Dasar dari model merupakan algoritma boosting yang bertujuan menggabungkan banyak model sederhana (*weak classifier*) menjadi model kuat (*strong classifier*). Selain itu, boosting gradien berupaya untuk meningkatkan ketahanan dengan membuat fungsi kerugian algoritma turun sepanjang arah gradiennya dalam proses iterasi (M. Lin dkk., 2021).




**Gambar 2.7 Ilustrasi Struktur XGBoost**

Langkah perhitungan dari XGBoost mencakup dasar *boosting*, g*radient boosting*. Algoritma XGBoost menggunakan fungsi objektif untuk menilai seberapa baik model pada penggunaan model latih. Fungsi ini terdiri dari dua komponen yaitu *loss function* dan *regularization terms* yang ditampilkan pada persamaan 2.7.

$$Obj = L + \Omega \tag{2.7}$$

Keterangan :

$L$ : *Loss function* (mengukur kesalahan antara prediksi dan nilai sebenarnya)

$\Omega$ : *Regularization term* (Mengontrol kompleksitas model untuk mencegah overfitting)

Dengan demikian, fungsi objektif pada iterasi *t* terdapat pada persamaan 2.8 dengan regularisasi pada fungsi 2.9.

$$L = \sum_{i=1}^{n}[L\left(y_i, \hat{y}_i^{(t-1)}\right) + f_t(x_i)] \quad (2.8)$$

Keterangan :

$\hat{y}_i^{(t-1)}$ : prediksi hingga iterasi $t-1$

$y_i$ : nilai aktual

$f_t(x_i)$ : pohon baru yang ditambahkan pada iterasi *t*

$$\Omega = \gamma T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \quad (2.9)$$

Keterangan :

$\gamma$ : parameter penalti terhadap jumlah *T*.

$T$ : jumlah node daun dalam pohon keputusan

$\lambda$ : Parameter yang mengontrol penalti pada bobot node

$w$ : bobot pada daun ke-j

Sedangkan perhitungan dari nilai pembobotan (*weight*) dari setiap leaf node terhadap prediksi akhir adalah dengan persamaan 2.10.

$$w_j = -\frac{G_j}{H_j + \lambda} \quad (2.10)$$

Keterangan :

$G_j$ : total *gradien* pada leaf node j

$H_j$ : total *hessian* pada leaf node j

$\lambda$ : parameter regularisasi

XGBoost menghitung perbaikan prediksi model dengan fungsi *gradien* dan *hessian* (untuk turunan kedua) untuk setiap iterasi yang terdapat pada fungsi 2.11 dan 2.12.

$$g_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial(\hat{y}_i)} \quad (2.11)$$

$$h_i = \frac{\partial L(y_i, \hat{y}_i)}{\partial(\hat{y}_i^2)} \quad (2.12)$$

Keterangan :

$g_i$ : *gradien* untuk sampel *i*

$h_i$ : *hessian* untuk sampel *i*

$y_i$ : nilai aktual

$\hat{y}_i$ : prediksi model

Selanjutnya lakukan perhitungan *gain* untuk mengukur kualitas dari pemecahan (split) node dalam pohon keputusan yang dihitung berdasarkan *gradien* dan *hessian* untuk mengukur keuntungan yang diperoleh dengan persamaan 2.12.

$$Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L+\lambda} + \frac{G_R^2}{H_R+\lambda} - \frac{(G_L+G_R)^2}{H_L+H_R+\lambda}\right] - \gamma \quad (2.12)$$

Keterangan :

$G_L$ dan $G_R$ : *gradien* untuk leaf node kiri dan kanan

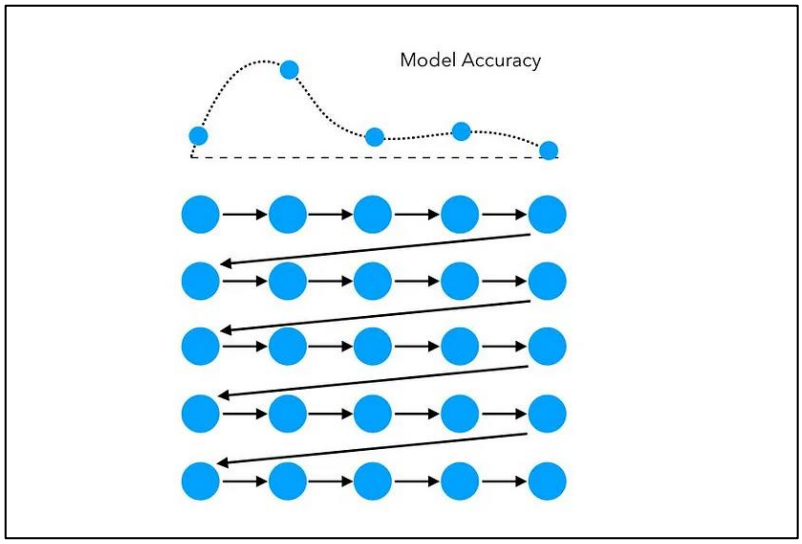
$H_L$ dan $H_R$ : *gradien* untuk leaf node kiri dan kanan

$\lambda$ : *regularisasi* untuk mengurangi overfitting

$\gamma$ : penalti untuk jumlah leaf node

## 2.6. GridSearchCV

Dalam metode *machine learning* terdapat beberapa nilai yang dapat meningkatkan kinerja model disebut dengan *hyperparameter*. *Hyperparameter* dapat digunakan untuk meningkatkan kinerja dan mempengaruhi berbagai uji model, pencarian *hyperparameter* dilakukan secara manual atau dengan menguji pada parameter yang sebelumnya telah ditentukan (Herni Yulianti dkk., 2022). Metode *hyperparameter* yang paling sering diaplikasikan adalah *grid search cross validation* (CV) yang diilustrasikan pada Gambar 2.8.




**Gambar 2.8 Ilustrasi *Gris Search Cross Validation***

*GridSearchCV* merupakan metope paling umum yang menggunakan *cross validation* dalam melatih model secara otomatis dengan memvalidasi setiap kombinasi *hyperparameter* sehingga dapat membuat model lebih efisien dengan menghemat waktu pemrosesan. Perhitungan *GridSearchCV* dengan menghitung jumlah kombinasi apabila terdapat *k hyperparameter* dengan masing – masing jumlah n1, n2, n3, …, nk.

Proses ini dihitung pada dengan persamaan 2.13 dan 2.14.

$$Jumlah\ Kombinasi = n_1\ x\ n_2\ x\ n_3\ x\ \dots\ x\ n_k \quad (2.13)$$

$$Total\ Evaluasi = Jumlah\ Kombinasi\ x\ Jumlah\ CV \quad (2.14)$$

Keterangan :

*k* : jumlah *hyperparameter*

*n* : jumlah nilai setiap *hyperparameter* ke-i yang diuji

## 2.7. Confusion Matrix

Confusion matrix merupakan tabel silang yang digunakan untuk mencatat jumlah kejadian berdasarkan dua kategori yakni klasifikasi sebenarnya (true/actual calssification) dan klasifikasi yang di prediksi (predicted classification). Matrix ini digunakan untuk mengevaluasi kinerja model machine learning khususnya pada masalah klasifikasi. Tabel dari confusion matix ditampilkan pada Tabel 2.3 dibawah.

**Tabel 2.3 Tabel *Confusion Matrix***

| **Aktual** | **Prediksi** | |
|---|---|---|
| | **Negatif (0)** | **Positif (1)** |
| **Negatif (0)** | *True Negative* (TN) | *False Positive* (FP) |
| **Positif (1)** | *False Negative* (FN) | *True Positive* (TP) |

Didapat dari Tabel 2.3 diatas, menghasilkan empat kategori pada *confusion matrix*, dengan penjelasan sebagai berikut:

1. *True Negative* (TN) : Kasus di mana model memprediksi kelas negatif, dan prediksi tersebut benar (sesuai dengan nilai sebenarnya yaitu negatif).
2. *False Negative* (FN) : Kasus dimana model memprediksi hasil kelas negatif, tetapi prediksi tersebut salah (nilai sebenarnya positif).
3. *True Positive* (TP) : Kasus di mana model memprediksi kelas positif, dan prediksi tersebut benar (sesuai dengan nilai sebenarnya yaitu positif).
4. *False Positive* (FP) : Kasus di mana model memprediksi kelas positif, tetapi prediksi tersebut salah (nilai sebenarnya negatif).

Berdasarkan empat komponen kategori tersebut memberikan evaluasi performa model klasifikasi dengan memberikan metrik seperti:

1. *Accuracy*

   *Accuracy* atau akurasi digunakan untuk mengukur kinerja dari model,

merupakan rasio jumlah kejadian yang benar. Persamaan dari perhitungan terdapat pada persamaan 2.15.

$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \quad (2.15)$$

2. *Sensitivity*

   *Sensitivity* atau sensitivitas adalah ukuran seberapa akurat prediksi positif pada suatu model. Presisi diartikan sebagai rasio prediksi positif yang benar terhadap jumlah total prediksi positif yang dibuat oleh model dikenal sebagai *True Positive Rate* atau *Recall*. Persamaan dari sensitivitas dapat dilihat pada persamaan 2.16.

   $$Sensitivitas = \frac{TP}{TP+FN} \quad (2.16)$$

3. *Specify*

   *Specify* atau Spesifitas merupakan evaluasi pengukuran kemampuan model untuk mengidentifikasi hasil negatif dengan benar. Spesifitas biasa dikenal sebagai *True Negative Rate* dengan persamaan 2.17 sebagai berikut.

   $$Spesifitas = \frac{TN}{TN+FP} \quad (2.17)$$

4. *G-mean*

   *G-mean* adalah rata-rata geometris antara sensitivitas dan spesifitas. *G-mean* bertujuan untuk melihat keseimbangan performa model pada kedua kelas (positif dan negatif), terutama pada dataset yang tidak seimbang. Nilai *g-mean* yang tinggi menunjukkan bahwa model bekerja dengan baik untuk kedua kelas. Persamaan dari perhitungan *g-mean* dapat dilihat pada persamaan 2.18.

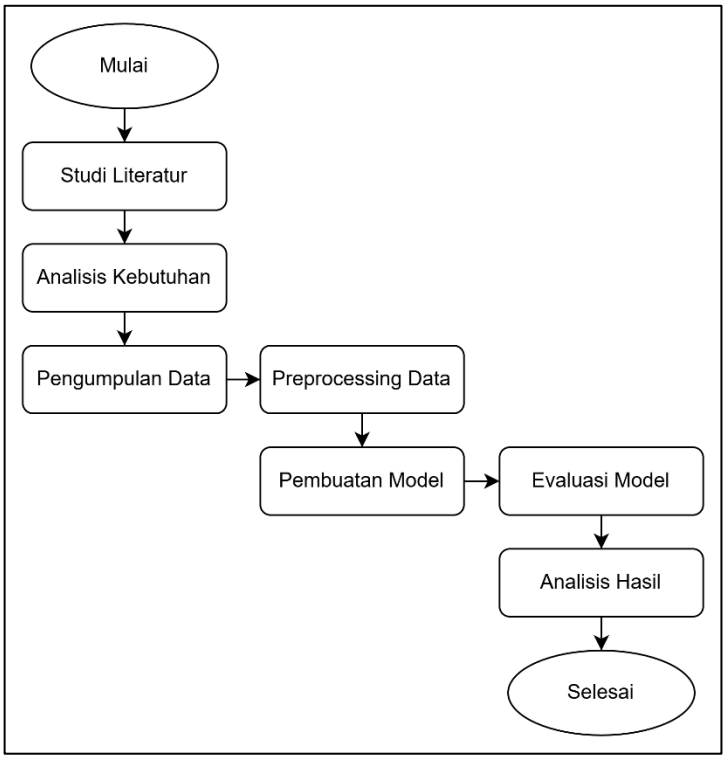
   $$G - mean = \sqrt{sensitifitas\ x\ spesifitas} \quad (2.18)$$

*Halaman ini sengaja dikosongkan*

# BAB III
# METODOLOGI

## 3.1. Tahapan Penelitian

Tahapan penelitian meliputi alur atau panduan dalam menentukan kegiatan pada penelitian dengan tujuan agar penelitian dapat berjalan sesuai dengan proses dari awal hingga akhir. Adapun penelitian yang dilakukan oleh penulis digambarkan melalui tahapan secara sistematis, terdapat pada Gambar 3.1.




**Gambar 3.1 Alur penelitian**

## 3.2. Studi Literatur

Pada tahap studi literatur ini penulis mengumpulkan berbagai literatur yang dapat membantu dalam menyelesaikan penelitian meliputi jurnal, artikel, serta referensi lainnya. Penulis juga mencari referensi utama pada jurnal atau artikel terkait klasifikasi pada penyakit jantung serta penggunaan teknik *resampling*.

## 3.3. Analisis Kebutuhan

Tahapan analisis kebutuhan yang dilakukan oleh penulis yakni dengan melakukan analisis kebutuhan untuk menentukan apa saja yang diperlukan pada penelitian, hal ini melibatkan perangkat keras atau *hardware* dan perangkat lunak atau *software*.

### 3.3.1. Spesifikasi Perangkat Keras

Pada penelitian ini menggunakan *device* berupa laptop dengan spesifikasi tertera pada Tabel 3.1. Laptop digunakan pada seluruh tahapan penelitian, mulai dari awal memasukkan dataset, pembuatan model klasifikasi, hingga penulisan laporan akhir.

**Tabel 3.1 Spesifikasi Perangkat Keras**

| No. | Perangkat Keras | Spesifikasi |
|---|---|---|
| 1 | Model Perangkat | Lenovo Legion 5 15ACH6H |
| 2 | CPU | AMD Ryzen 5 5600H (6 cores/12 threads, up to 4.2GHz, 16MB) |
| 3 | GPU | NVIDIA GeForce RTX 3050 4GB GDDR6 (dedicated, 95W TGP) |
| 4 | RAM | 16 GB |

### 3.3.2. Spesifikasi Perangkat Lunak

Dalam mendukung analisis dan komputasi pada penelitian dari awal hingga akhir, penelitian ini menggunakan beberapa perangkat lunak dengan rincian yang dibutuhkan dapat dilihat pada Tabel 3.2 berikut ini.

| No. | Perangkat Lunak | Spesifikasi |
|---|---|---|
| 1 | Aplikasi | Jupyter Notebook |
| 2 | Bahasa Pemrograman | Python |
| 3 | Library | pandas<br>numpy<br>matplotlib<br>scipy<br>scikit-learn<br>imblearn<br>xgboost |

## 3.4. Pengumpulan Data

Dalam penelitian ini menggunakan data sekunder yang didapat dari situs yang menyediakan dataset bernama Kaggle pada https://www.kaggle.com/dataset dengan judul “Indicators of Heart Disease (2022 UPDATE)”. Dataset didapat dari survei yang

dilakukan oleh Behavioral Risk Factor Surveillance System (BRFSS). Dataset berisi data sebanyak 445.132 baris dengan 40 atribut berformat CSV dapat dilihat pada rincian lengkap terkait atribut pada Tabel 3.3.

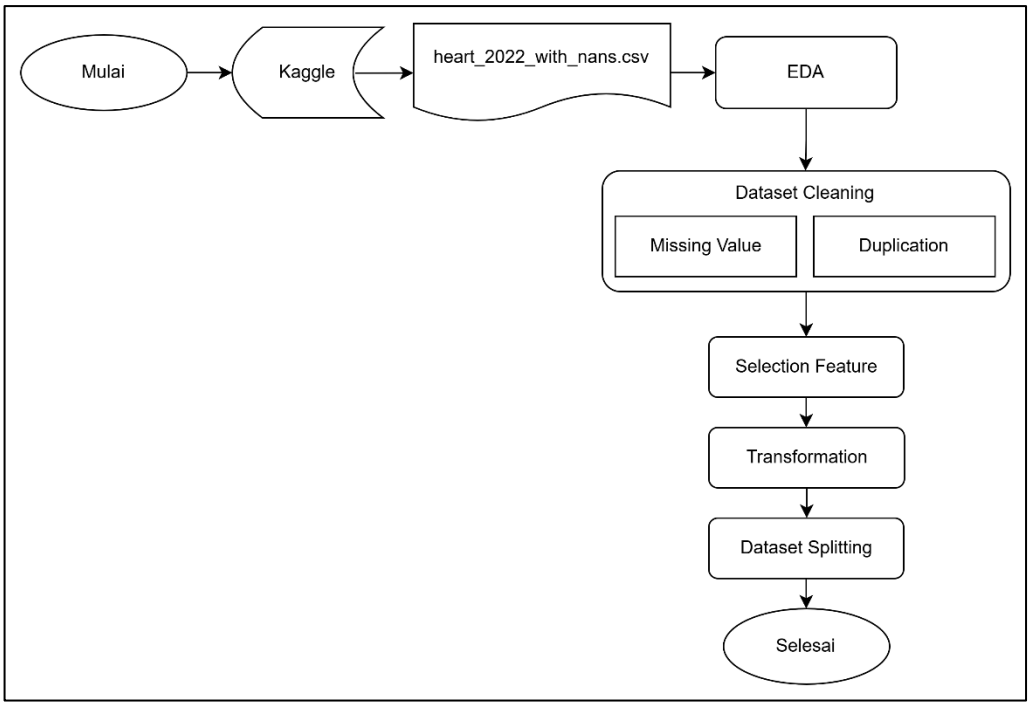
**Tabel 3.3. Rincian Atribut pada Dataset**

| No. | Nama Atribut | Keterangan |
|---|---|---|
| 1 | State | Negara bagian tempat responden tinggal |
| 2 | Sex | Jenis kelamin responden (pria/wanita) |
| 3 | GereralHealth | Penilaian umum terhadap kondisi kesehatan (baik/buruk) |
| 4 | PhysicalHealthDays | Jumlah hari kesehatan fisik yang buruk dalam sebulan terakhir |
| 5 | MentalHealthDays | Jumlah hari kesehatan mental yang buruk dalam sebulan terakhir |
| 6 | LastCheckupTime | Waktu terakhir melakukan pemeriksaan kesehatan |
| 7 | PhysicalActivities | Aktivitas fisik yang dilakukan responden |
| 8 | SleepHours | Rata-rata jam tidur per malam |
| 9 | RemovedTeeth | Jumlah gigi yang telah dicabut |
| 10 | HadHeartAttack | Pernah mengalami serangan jantung |
| 11 | HadAngina | Pernah mengalami angina (nyeri dada) |
| 12 | HadStroke | Pernah mengalami stroke |
| 13 | HadAsthma | Pernah didiagnosis asma |
| 14 | HadSkinCancer | Pernah didiagnosis kanker kulit |
| 15 | HadCOPD | Pernah didiagnosis dengan Penyakit Paru Obstruktif Kronis (COPD) |
| 16 | HadDepressiveDisorder | Pernah didiagnosis gangguan depresi |
| 17 | HadKidneyDisease | Pernah mengalami penyakit ginjal |
| 18 | HadArthritis | Pernah didiagnosis radang sendi |
| 19 | HadDiabetes | Pernah didiagnosis diabetes |
| 20 | DeafOrHardOfHearing | Mengalami gangguan pendengaran |
| 21 | BlindOrVisionDifficulty | Mengalami gangguan penglihatan |
| 22 | DifficultyConcentrating | Kesulitan berkonsentrasi |

| No. | Nama Atribut | Keterangan |
|---|---|---|
| 23 | DifficultyWalking | Kesulitan berjalan |
| 24 | DifficultyDressingBathing | Kesulitan berpakaian atau mandi |
| 25 | DifficultyErrands | Kesulitan melakukan tugas sehari-hari |
| 26 | SmokerStatus | Status merokok (aktif/tidak) |
| 27 | ECigaretteUsage | Penggunaan rokok elektronik |
| 28 | ChestScan | Riwayat pemindaian dada |
| 29 | RaceEthnicityCategory | Kategori ras atau etnis |
| 30 | AgeCategory | Kategori usia responden |
| 31 | HeightInMeters | Tinggi badan dalam meter |
| 32 | WeightInKilograms | Berat badan dalam kilogram |
| 33 | BMI | Indeks Massa Tubuh (Body Mass Index) |
| 34 | AlcoholDrinkers | Kebiasaan konsumsi alkohol |
| 35 | HIVTesting | Riwayat tes HIV |
| 36 | FluVaxLast12 | Pernah menerima vaksin flu dalam 12 bulan terakhir |
| 37 | PneumoVaxEver | Pernah menerima vaksin pneumonia |
| 38 | TetanusLast10Tdap | Riwayat vaksin tetanus dalam 10 tahun terakhir |
| 39 | HighRiskLastYear | Risiko tinggi terkena penyakit dalam setahun terakhir |
| 40 | CovidPos | Status pernah terinfeksi COVID-19 |

### 3.5. Preprocessing Data

Data yang telah didapat pada penelitian ini masih dalam bentuk *raw* atau data mentah, sehingga diperlukan adanya tahapan *preprocessing* terlebih dahulu sebelum dilakukan tahapan pembuatan model. Tujuan adanya tahapan ini adalah untuk meningkatkan kualitas pada data dan mempermudah model dalam pengklasifikasian. Tahapan yang dilakukan pada *preprocessing* ditampilkan pada Gambar 3.2.




**Gambar 3.2 Flowchart *Preprocessing* Data**

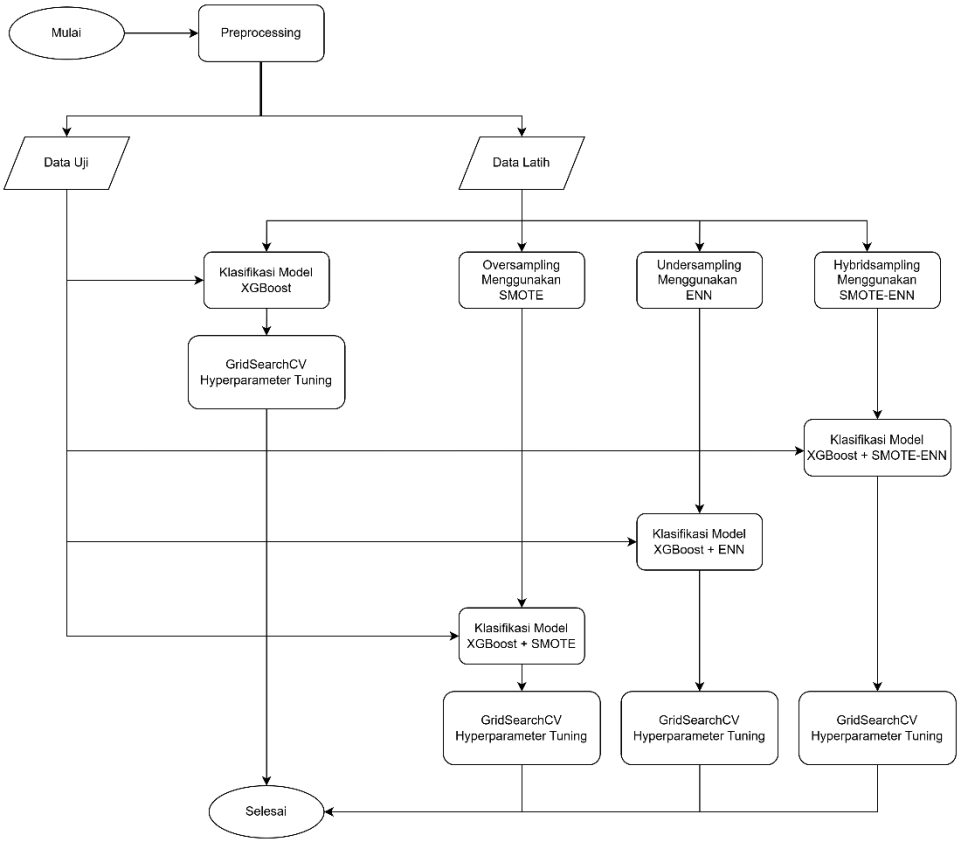
Berdasarkan gambar 3.2 menjelaskan mengenai alur tahap *preprocessing* sebagai berikut:

1. EDA (*Exploratory Data Analysis*) merupakan tahap awal dari *preprocessing* data, pada tahapan ini penulis melakukan pemahaman terkait karakteristik data secara mendalam. Penulis memeriksa beberapa hal dimulai dari melihat setiap tipe data pada atribut, kemudian memeriksa apakah terdapat *missing value* dan duplikasi data, dan terakhir melihat ketidakseimbangan pada dua kelas penyakit dan tanpa penyakit pada atribut penyakit jantung. Proses ini menggunakan visualisasi guna memudahkan pemahaman terkait hubungan target dengan atribut lainnya.
2. *Dataset cleaning* atau pembersihan data merupakan tahap kedua, pada tahap ini akan dilakukan pembersihan pada *missing value* dan duplikasi data. Pada data yang terindikasi duplikasi akan dihapus, sedangkan pada *missing value* dilakukan penghapusan data pada nilai hilang untuk fitur kategorikal dan penggantian data dengan *mean* atau nilai rata-rata pada fitur numerik.
3. *Selection feature* atau seleksi fitur diposisikan pada tahap ketiga, pada tahap ini dilakukan proses pemilihan sebagian fitur dari keseluruhan yang ada pada dataset. Pemilihan fitur bertujuan untuk mengurangi waktu dan sumber daya selama komputasi pelatihan modul dengan hanya menggunakan fitur yang dianggap relevan atau informatif terhadap model.

4. *Transformation* atau transformasi merupakan tahap selanjutnya setelah dilakukannnya seleksi fitur. Pada tahap ini penulis melakukan beberapa cara untuk transformasi pada data kategorikal yaitu transformasi *binary encoding* untuk data dengan jawaban "yes" atau "no", *ordinal encoding* untuk data dengan jawaban rentang, dan *one-hot encoding* untuk data dengan jawaban beberapa kategori. Hasil dari transformasi adalah bentuk numerik agar dapat diproses oleh model.
5. *Dataset splitting* atau pemisahan data merupakan tahap terakhir. Pada proses ini dilakukan pembagian data untuk pelatihan terhadap model dan data untuk pengujian setelah proses pelatihan selesai dilakukan.

### 3.6. Pembuatan Model

Setelah usai melakukan tahap *preprocessing* pada dataset, maka dataset siap digunakan untuk pembuatan model klasifikasi. Model yang dibuat memiliki tujuan dalam mengklasifikasikan apakah seseorang memiliki potensi penyakit jantung atau tidak. Model klasifikasi yang dibuat memiliki proses yang dapat dilihat pada Gambar 3.3.

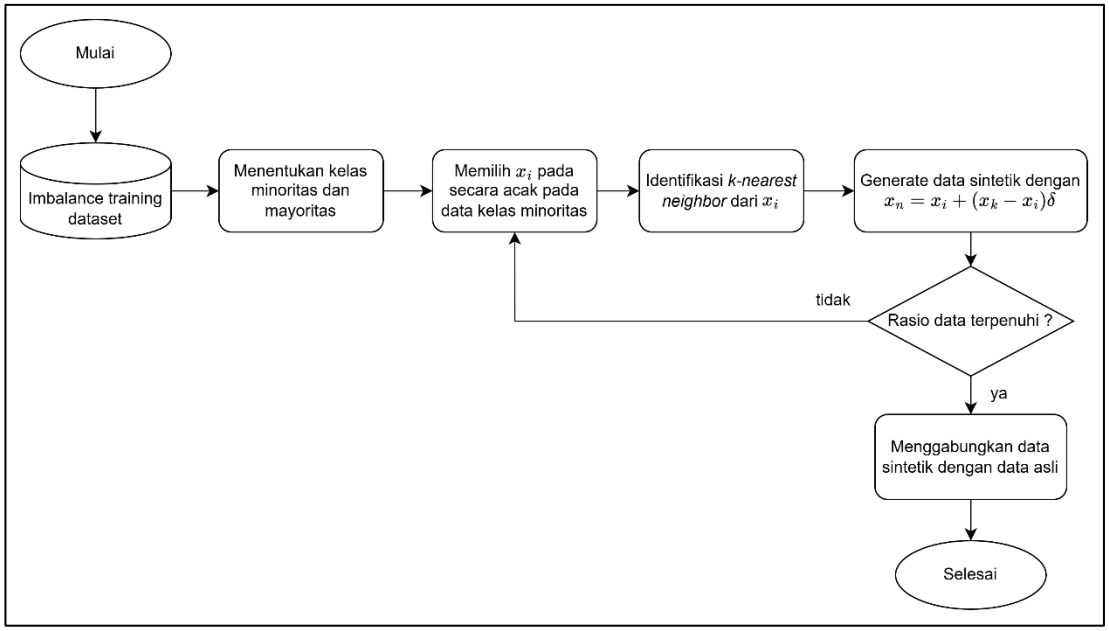



**Gambar 3.3 Flowchart Pembuatan Model Klasifikasi**

Gambar 3.3 menjelasakan tahapan pembuatan model, model XGBoost digunakan dalam mengklasifikasikan data pelatihan yang didapat dari tahapan *preprocessing*. Selanjutnya penelitian ini akan membandingkan dengan penggunaan teknik *resampling* sebelum dilakukan pelatihan model agar data dapat lebih seimbang. Perbandingan ini dimulai dari penggunaan *oversampling* menggunakan teknik SMOTE, kemudian dilanjut dengan *undersampling* menggunakan teknik ENN, dan diakhiri dengan kombinasi keduanya atau *hybrid* menggunakan teknik SMOTEENN. Kemudian semua model yang berhasil dilatih akan digunakan dalam memprediksi data pengujian yang berbeda dari data pelatihan dan belum pernah dilihat pada model sebelumnya. Setelah *baseline* dari tiap model selesai dibangun, selanjutnya adalah melakukan *tuning hyperparameter* menggunakan *GridSearchCV* agar parameter pada model dapat dioptimalkan terhadap hasil dari masing-masing pendekatan balancing data.

### 3.6.1. Teknik *Oversampling* (SMOTE)

Teknik SMOTE atau *Syntethic Minority Oversampling Technique* adalah teknik dalam pada metode *oversampling* yang membuat data sintetik berdasarkan tetangga terdekat (*neirest neighbor*) pada tiap data minoritas. Proses dalam metode ini ditampilkan pada Gambar 3.5.



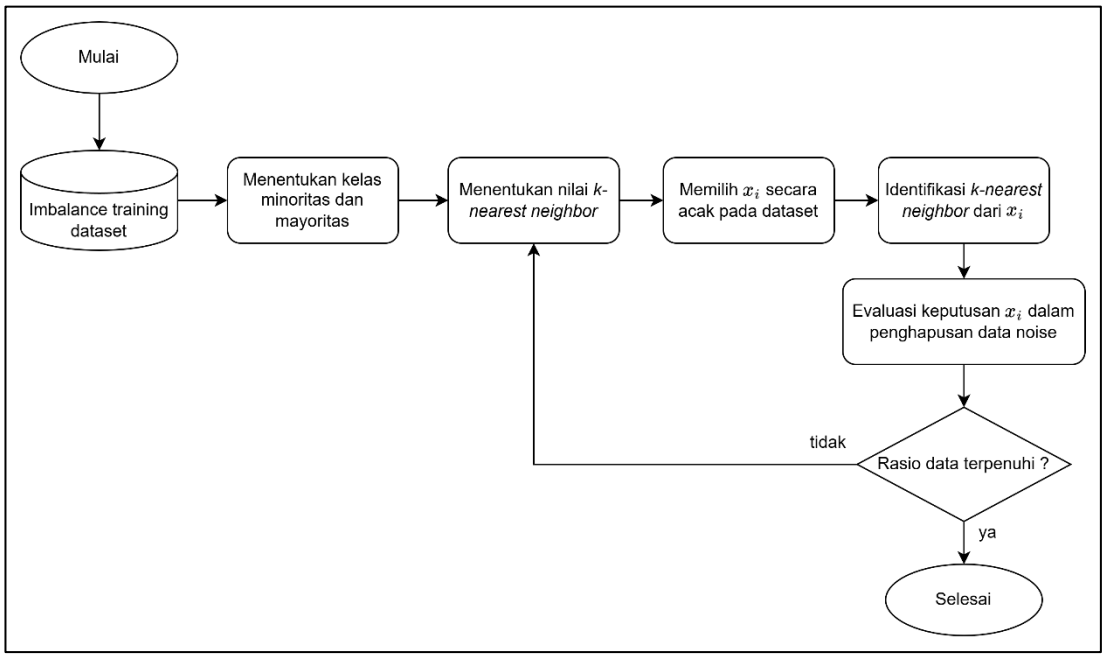

**Gambar 3.5 Flowchart Teknik SMOTE**

Mengacu pada Gambar 3.5, metode SMOTE dilakukan dari data yang tidak seimbang, kemudian menentukan kelas target berdasarkan kelas minoritas dan kelas mayoritas. Selanjutnya memilih sampel dari kelas minoritas ($x_i$) secara acak,

kemudian dilanjutkan dengan mengidentifikasi *k-neirest neighbor* dengan jarak euclidean antara sampel dengan *k* tetangga terdekatnya. Setelah itu barulah data sintetik dibuat dengan menambahkan bilangan desimal acak antara 0 hingga 1. Proses tersebut akan terus dilakukan sampai jumlah rasio *oversampling* terpenuhi sesuai dengan yang diharapkan. Langkah terakhir adalah menggabungkan data sintetik dengan data asli pada kelas minoritas sehingga data memiliki rasio yang sama dengan kelas mayoritas.

### 3.6.2. Teknik *Undersampling* (ENN)

Teknik ENN (*Edited Nearest Neighbor*) merupakan teknik distribusi data dengan menyeimbangkan rasio dengan menghilangkan sebagaian data pada kelas mayoritas sehingga memiliki jumlah yang sama antara data kelas minoritas dengan kelas mayoritas. Proses dalam metode ENN ditampilkan pada Gambar 3.6.



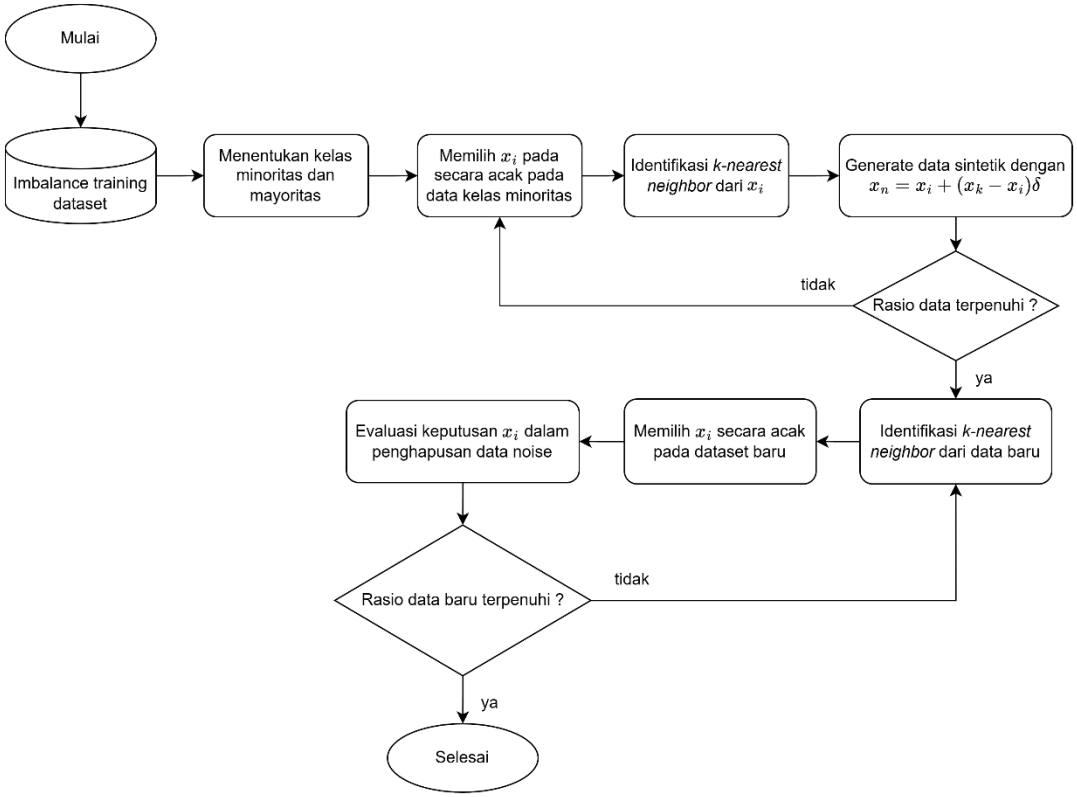

**Gambar 3.6 Flowchart Teknik ENN**

Berdasarkan Gambar 3.6, metode ENN dilakukan berdarkan data yang tidak seimbang, setelah itu menentukan kembali kelas minoritas dan mayoritas. Teknik ini berbeda dengan SMOTE karena menggunakan *undersampling* yang berarti pengurangan data pada kelas mayoritas. Selanjutnya menentukan nilai *k* dengan tetangganya, kemudian dilanjut dengan memilih sampel data acak pada kelas mayoritas. Setelah itu identifikasi sampel dengan *k* tetangga terdekat dan evaluasi hasil keputusan, pada tahap ini ENN akan menghapus sampel jika label dari $x_i$ berbeda

dengan mayoritas label dari tetangganya. Ulangi proses hingga rasio data terpenuhi, maka proses ENN selesai dilakukan.

### 3.6.3. Teknik *Hybridsampling* (SMOTEENN)

Metode SMOTEENN merupakan kombinasi dari teknik *oversampling* SMOTE dengan teknik *undersampling* ENN. Metode kombinasi atau *hybrid* ini bertujuan untuk meningkatkan representasi kelas minoritas juga untuk mengurangi noise pada kelas mayoritas, sehingga dapat menciptakan dataset yang lebih seimbang dan lebih bersih. Proses pada teknik SMOTEENN terdapat pada Gambar 3.7.




**Gambar 3.7. Flowchart Teknik SMOTEENN**

Berdasarkan pada Gambar 3.7, pada dasarnya teknik SMOTEENN tidak jauh berbeda dari penggunaan teknik SMOTE dan ENN secara terpisah yang telah dijelaskan sebelumnya. Teknik SMOTEENN dimulai dari proses SMOTE dengan menambahkan data sintetik kedalam kelas minoritas agar data dapat lebih seimbang, disinilah proses SMOTE berakhir. Selanjutnya adalah proses ENN dengan mengidentifikasi data baru hasil dari SMOTE kemudian mengevaluasi data dengan menghapus *noise* pada dataset dengan teknik ENN. Hal ini akan terus berulang hingga rasio data terpenuhi sesuai dengan keinginan, maka proses SMOTEENN berakhir dengan data seimbang dan bebas *noise*.

### 3.6.4. Model XGBoost

Pada tahap pembuatan model klasifikasi, penulis menggunakan *hyperparameter tuning* sebagai optimasi untuk model XGBoost, model akan dilakukan optimasi dengan *GridSearchCV* baik pada model XGBoost dengan teknik *resampling* maupun tidak. Acuan untuk menilai seberapa baik optimasi *hyperparameter* menggunakan evaluasi *g-mean*. Rincian optimasi ditampilkan pada Tabel 3.4.

**Tabel 3.4 Nilai *Hyperparameter* yang Diuji**

| Parameter | Rentang Nilai Uji | Keterangan |
|---|---|---|
| n_estimators | 100 – 300 | Parameter yang digunakan untuk menentukan jumlah pohon dalam proses ensemble. Semakin banyak pohon yang digunakan, model akan menjadi lebih kompleks dan memiliki kapasitas lebih besar untuk menangkap pola dalam data. |
| max_depth | 5 – 9 | Parameter yang digunakan untuk mengatur kedalaman maksimum setiap pohon dalam ensemble. Semakin besar nilai *max_depth*, semakin kompleks model karena mampu menangkap pola yang lebih rumit dalam data. Namun, nilai yang terlalu tinggi dapat menyebabkan overfitting, sementara nilai yang terlalu rendah dapat menyebabkan underfitting. |
| gamma | 0 – 1 | Parameter yang menentukan ambang batas untuk pembagian node. Semakin besar nilai *gamma*, semakin tinggi kriteria untuk membagi node, sehingga menghasilkan model yang lebih sederhana dan mengurangi risiko overfitting. |

| Parameter | Rentang Nilai Uji | Keterangan |
|---|---|---|
| learning_rate | 0.01 – 0.3 | Parameter yang mengatur langkah pembaruan pada setiap iterasi. Nilai yang lebih kecil membuat model belajar lebih lambat tetapi lebih akurat, karena setiap iterasi memberikan kontribusi kecil pada hasil akhir. Nilai yang lebih besar mempercepat pelatihan tetapi berisiko kehilangan detail pola. |
| reg_lambda | 10 – 100 | Parameter yang mengontrol regularisasi L2 untuk mencegah *overfitting*. Nilai yang lebih besar meningkatkan penalti untuk bobot yang besar, sehingga menghasilkan model yang lebih sederhana dan tahan terhadap *overfitting*. |
| scale_pos_weight | 10 – 19 | Parameter yang digunakan untuk menangani ketidakseimbangan kelas dengan menyesuaikan bobot kelas positif. Nilai yang lebih tinggi memberikan penekanan lebih besar pada kelas minoritas, sehingga membantu meningkatkan performa model pada data yang tidak seimbang. |

### 3.7. Evaluasi Model

Evaluasi model dilakukan secara menyeluruh setelah mendapatkan hasil dari model klasifikasi. Evaluasi dengan membandingkan hasil klasifikasi dengan data pengujian, kemudian juga mengidentifikasi apakah terjadi *overfitting* atau *underfitting* pada model klasifikasi. Penggunaan *confusion matrix* untuk melihat informasi berupa *accuracy*, *sensitivity*, *specifity*, dan *g-mean* yang dihitung dari tiap kolom *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). Metriks tersebut akan digunakan dalam menganalisis dan membandingkan dengan skenario lainnya.

### 3.8. Analisis Hasil

Setelah semua proses dalam evaluasi model telah dilakukan yakni dengan menguji matriks berupa *accuracy*, *sensitivity*, *specifity*, dan *g-mean*, selanjutnya dilakukan analisis dengan membandingkan kinerja model klasifikasi sesuai skenario. Analisis dilakukan dengan mengidentifikasi waktu eksekusi dari tiap pelatihan model yang diukur dalam satuan menit untuk melihat efisiensi komputasi dari beberapa skenario. Selain itu juga menggunakan diagram dari *confusion matrix* untuk memvisualisasikan perbandingan antar model skenario, yang diharapkan dapat mempermudah dalam membandingkan dan menganalisis performa tiap klasifikasi model.

### 3.9. Skenario Pengujian

Beberapa skenario uji coba dilakukan bertujuan untuk memperoleh pemahaman yang lebih baik dan efisien terkait performa masing – masing model pada berbagai skenario. Beberapa uji coba skenario yang dilakukan dalam merancang model secara keseluruhan sebagai berikut:

1. Perubahan rasio *splitting* data latih dan data uji

   Pada skenario pengujian ini dilakukan dengan tujuan memperoleh berbagai pengaruh dalam beberapa variasi proporsi pada data latih dan data uji terhadap performa model. Skenario ini melakukan beberapa variasi dalam membagi data latih dan data uji antara lain 90:10, 80:20, dan 70:30. Model dikatakan baik apabila mampu bekerja secara stabil dan konsisten terhadap berbagai perubahan rasio data, sehingga diperlukan pengujian dengan mengubah variasi proporsi data yang bertujuan untuk memastikan bahwa model tidak mengalami *underfitting* atau *overfitting* karena proporsi pembagian data yang tidak seimbang.

2. Perubahan rasio *sampling*

   Pada skenario perubahan ini, dilakukan perubahan pada rasio *sampling* untuk melihat berbagai perbandingan dalam rasio data pada teknik *resampling*. Rasio yang akan digunakan dalam pengujian antara lain adalah 0.1, 0.3, 0.5, 0.7, dan 0.9. Rasio sampling berarti semakin besar nilainya, maka jumlah kelas minoritas dan kelas mayoritas akan semakin seimbang nilainya. Tujuan dalam skenario ini adalah untuk menguji dan menemukan rasio sampling yang paling optimal dalam menghasilkan performa model klasifikasi terbaik.

3. Perubahan nilai *k-neighbor*

   Pada skenario pengujian ini, dilakukan variasi nilai parameter *k-neighbor* dalam algoritma *resampling* untuk memahami pengaruhnya terhadap performa model. Nilai *k-neighbor* menentukan jumlah tetangga terdekat yang dipertimbangkan saat menghasilkan data baru dalam proses *resampling*. Nilai-nilai *k-neighbor* yang diuji mencakup 3 dan 5. Tujuan dari skenario ini adalah untuk mengidentifikasi nilai *k* yang memberikan distribusi data terbaik dan meningkatkan kemampuan generalisasi model terhadap data baru.

*Halaman ini sengaja dikosongkan*

# BAB IV
# PENGUJIAN DAN ANALISA

## 4.1. Metode Pengujian

-

## 4.2. Hasil Pengujian

-

*Halaman ini sengaja dikosongkan*

# BAB V
# PENUTUP

## 5.1. Kesimpulan

Dari keseluruhan pengujian sistem, hasil dari penelitian ini dapat disimpulkan antara lain sebagai berikut:

1.

2.

3.

## 5.2. Saran Pengembangan

1.

2.

3.

*Halaman ini sengaja dikosongkan*

# DAFTAR PUSTAKA


Abdurrahman, G., Oktavianto, H., & Sintawati, M. (2022). Optimasi Algoritma XGBoost Classifier Menggunakan Hyperparameter Gridesearch dan Random Search Pada Klasifikasi Penyakit Diabetes. *INFORMAL: Informatics Journal*, *7*(3), 193. https://doi.org/10.19184/isj.v7i3.35441

Aditya Gumilar, Sri Suryani Prasetiyowati, & Yuliant Sibaroni. (2022). Performance Analysis of Hybrid Machine Learning Methods on Imbalanced Data (Rainfall Classification). *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *6*(3), 481–490. https://doi.org/10.29207/resti.v6i3.4142

Ain, N., Hariyanto, D., & Rusdan, S. (2015). Karakteristik Penderita Penyakit Jantung Bawaan pada Anak di RSUP Dr. M. Djamil Padang Periode Januari 2010 – Mei 2012. *Jurnal Kesehatan Andalas*, *4*(3). https://doi.org/10.25077/jka.v4i3.388

Alhasani, A. T., Alkattan, H., Subhi, A. A., El-Kenawy, E.-S. M., & Eid, M. M. (2023). A Comparative Analysis of Methods for Detecting and Diagnosing Breast Cancer Based on Data Mining. *Journal of Artificial Intelligence and Metaheuristics*, *4*(2), 08–17. https://doi.org/10.54216/JAIM.040201

Amanah, D. A., & Herawati, T. (2022). Pengaruh Telenursing terhadap Quality of Life (QoL) Pada Pasien Gagal Jantung: Literature Review. *JHCN Journal of Health and Cardiovascular Nursing*, *2*. https://doi.org/10.36082/jhcnv2i1.408

Ameliana, N., Suarna, N., & Prihartono, W. (2024). ANALISIS DATA MINING PENGELOMPOKKAN UMKM MENGGUNAKAN ALGORITMA K-MEANS CLUSTERING DI PROVINSI JAWA BARAT. *JATI (Jurnal Mahasiswa Teknik Informatika)*, *8*(3), 3261–3268. https://doi.org/10.36040/jati.v8i3.9655

Amin, F., Anggraeni, D. S., & Aini, Q. (2022). Penerapan Metode K-Means dalam Penjualan Produk Souq.Com. *Applied Information System and Management (AISM)*, *5*(1), 7–14. https://doi.org/10.15408/aism.v5i1.22534

Amrullah, A. Z., Sofyan Anas, A., Adrian, M., & Hidayat, J. (2020). Analisis Sentimen Movie Review Menggunakan Naive Bayes Classifier Dengan Seleksi Fitur Chi Square. *Jurnal*, *2*(1). https://doi.org/10.30812/bite.v2i1.804

Gumelar, G., & Al Fatta, H. (2023). Kombinasi Algoritma Klasifikasi Dengan Algoritma Oversampling Untuk Menangani Ketidakseimbangan Kelas Pada Level Data. *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*, *10*(2), 29–39. http://jurnal.mdp.ac.id

Hairani, H., & Priyanto, D. (2023). A New Approach of Hybrid Sampling SMOTE and ENN to the Accuracy of Machine Learning Methods on Unbalanced Diabetes Disease Data. *International Journal of Advanced Computer Science and Applications*, *14*(8). https://doi.org/10.14569/IJACSA.2023.0140864

Han, Y., & Joe, I. (2024). Enhancing Machine Learning Models Through PCA, SMOTE-ENN, and Stochastic Weighted Averaging. *Applied Sciences*, *14*(21), 9772. https://doi.org/10.3390/app14219772

Hastuti, Y. D., & Mulyani, E. D. (2019). Kecemasan Pasien dengan Penyakit Jantung Koroner Paska Percutaneous Coronary Intervention. *Jurnal Perawat Indonesia*, *3*(3), 167. https://doi.org/10.32584/jpi.v3i3.427

Herni Yulianti, S. E., Oni Soesanto, & Yuana Sukmawaty. (2022). Penerapan Metode Extreme Gradient Boosting (XGBOOST) pada Klasifikasi Nasabah Kartu Kredit. *Journal of Mathematics: Theory and Applications*, 21–26. https://doi.org/10.31605/jomta.v4i1.1792

Imani, M., & Arabnia, H. R. (2023). Hyperparameter Optimization and Combined Data Sampling Techniques in Machine Learning for Customer Churn Prediction: A Comparative Analysis. *Technologies*, *11*(6), 167. https://doi.org/10.3390/technologies11060167

Jain, A., Ahirwar, M., & Pandey, R. (2019). A Review on Intutive Prediction Of Heart Disease Using Data Mining Techniques. *International Journal of Computer Sciences and Engineering*, *7*(7), 109–113. https://doi.org/10.26438/ijcse/v7i7.109113

Kartina Diah Kusuma Wardani, & Memen Akbar. (2023). Diabetes Risk Prediction using Feature Importance Extreme Gradient Boosting (XGBoost). *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, *7*(4), 824–831. https://doi.org/10.29207/resti.v7i4.4651

Kurnia, D., Itqan Mazdadi, M., Kartini, D., Adi Nugroho, R., & Abadi, F. (2023). Seleksi Fitur dengan Particle Swarm Optimization pada Klasifikasi Penyakit Parkinson Menggunakan XGBoost. *Jurnal Teknologi Informasi dan Ilmu Komputer*, *10*(5), 1083–1094. https://doi.org/10.25126/jtiik.20231057252

Le, T., Vo, M. T., Vo, B., Lee, M. Y., & Baik, S. W. (2019). A Hybrid Approach Using Oversampling Technique and Cost-Sensitive Learning for Bankruptcy Prediction. *Complexity*, *2019*(1). https://doi.org/10.1155/2019/8460934

Lin, G.-M., & Zeng, H.-C. (2021). Electrocardiographic Machine Learning to Predict Mitral Valve Prolapse in Young Adults. *IEEE Access*, *9*, 103132–103140. https://doi.org/10.1109/ACCESS.2021.3098039

Lin, M., Zhu, X., Hua, T., Tang, X., Tu, G., & Chen, X. (2021). Detection of Ionospheric Scintillation Based on XGBoost Model Improved by SMOTE-ENN Technique. *Remote Sensing*, *13*(13), 2577. https://doi.org/10.3390/rs13132577

Medayati, N., Ridwan, A., Russeng, S., & Stang. (2018). KARAKTERISTIK DAN PREVALENSI RISIKO PENYAKIT KARDIOVASKULAR PADA TUKANG MASAK WARUNG MAKAN DI WILAYAH KERJA PUSKESMAS TAMALANREA. *Jurnal Kesehatan*, *11*(1), 30–38. https://doi.org/10.24252/kesehatan.v11i1.5029

Mutmainah, S. (2021). PENANGANAN IMBALANCE DATA PADA KLASIFIKASI KEMUNGKINAN PENYAKIT STROKE. *Jurnal Sains, Nalar, dan Aplikasi Teknologi Informasi*, *1*(1), 10–16. https://doi.org/10.20885/snati.v1i1.2

Naomi, W. S., Picauly, I., & Toy, S. M. (2021). Faktor Risiko Kejadian Penyakit Jantung Koroner. *Media Kesehatan Masyarakat*, *3*(1), 99–107. https://doi.org/10.35508/mkm.v3i1.3622

Pristyanto, Y. (2019). PENERAPAN METODE ENSEMBLE UNTUK MENINGKATKAN KINERJA ALGORITME KLASIFIKASI PADA IMBALANCED DATASET. *Jurnal Teknoinfo*, *13*(1), 11. https://doi.org/10.33365/jti.v13i1.184

Rifqi Fitriadi, & Deni Mahdiana. (2023). SYSTEMATIC LITERATURE REVIEW OF THE CLASS IMBALANCE CHALLENGES IN MACHINE LEARNING. *Jurnal Teknik Informatika (Jutif)*, *4*(5), 1099–1107. https://doi.org/10.52436/1.jutif.2023.4.5.970

Romli, I., & Firana Puspita Dewi, R. (2021). PENERAPAN DATA MINING MENGGUNAKAN ALGORITMA K-MEANS UNTUK KLASIFIKASI PENYAKIT ISPA. *Indonesian Journal of Business Intelligence (IJUBI)*, *4*(1), 10. https://doi.org/10.21927/ijubi.v4i1.1727

Saragih, A. D. (2020). Terapi Dislipidemia untuk Mencegah Resiko Penyakit Jantung Koroner. *Indonesian Journal of Nursing and Health Sciences*, *1*(1), 15–24. https://doi.org/10.37287/ijnhs.v1i1.223

Sir, Y. A., & Soepranoto, A. H. H. (2022). Pendekatan Resampling Data Untuk Menangani Masalah Ketidakseimbangan Kelas. *Jurnal Komputer dan Informatika*, *10*(1), 31–38. https://doi.org/10.35508/jicon.v10i1.6554

Suharno, C. F., Fauzi, M. A., & Perdana, R. S. (2017). Klasifikasi Teks Bahasa Indonesia Pada Dokumen Pengaduan Sambat Online Menggunakan Metode K-Nearest Neighbors Dan Chi-square. *Systemic: Information System and Informatics Journal*, *3*(1), 25–32. https://doi.org/10.29080/systemic.v3i1.191

Sutoyo, E., & Fadlurrahman, M. A. (2020). Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Television Advertisement Performance Rating Menggunakan Artificial Neural Network. *Jurnal Edukasi dan Penelitian Informatika (JEPIN)*, *6*(3), 379. https://doi.org/10.26418/jp.v6i3.42896

Utomo, D. P., & Mesran, M. (2020). Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, *4*(2), 437. https://doi.org/10.30865/mib.v4i2.2080

Wahyudi, E., & Hartati, S. (2017). Case-Based Reasoning untuk Diagnosis Penyakit Jantung. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *11*(1), 1. https://doi.org/10.22146/ijccs.15523