Jurnal Matematika UNAND
Vol. **12** No. **2** Hal. 176 – 193
ISSN : 2303–291X
e-ISSN : 2721–9410

PERFORMA KLASIFIKASI DATA TIDAK SEIMBANG DENGAN PENDEKATAN MACHINE LEARNING (STUDI KASUS: DIABETES INDIAN PIMA)

MASJIDIL AQSHA, NURTITI SUNUSI*
Departemen Statistika, Fakultas MIPA, Universitas Hasanuddin
email : masjidilaqsha15@gmail.com, nurtitisunusi@unhas.ac.id

Diterima 22 Desember 2022 Direvisi 24 Maret 2023 Dipublikasikan 21 Oktober 2023

Abstrak. Diabetes merupakan suatu penyakit atau gangguan metabolisme kronis dengan multi etiologi yang ditandai dengan tingginya kadar gula darah disertai dengan gangguan metabolisme karbohidrat, lipid, dan protein sebagai akibat insufisiensi fungsi insulin. Faktor risiko diabetes berhubungan dengan status diabetes sesorang. Berbagai pendekatan machine learning menjadi alternatif dalam memprediksi status diabetes. Namun, dalam banyak kasus, data yang tersedia tidak cukup seimbang dalam kelas datanya. Adanya ketidakseimbangan data dapat menyebabkan hasil prediksi menjadi tidak akurat. Tujuan penelitian dalam paper ini adalah untuk mengatasi masalah ketidakseimbangan data dan membandingkan kinerja model dalam memprediksi status diabetes. Secara umum, metode seperti *Synthetic Minority Over-sampling Technique* (SMOTE) dan *Adaptive Synthetic* (ADASYN) dapat digunakan untuk menyeimbangkan data. Data Diabetes Indian Pima yang telah diseimbangkan kemudian diprediksi dengan metode *machine learning* seperti metode *Bagging, Random Forest*, dan XGBoost. Penelitian ini diharapkan memberikan model terbaik yang dapat mengklasifikasikan status diabetes perempuan keturunan Indian Pima.

Abstract. *Diabetes is a chronic metabolic disease or disorder with various etiologies characterized by high blood sugar levels accompanied by disturbances in carbohydrate, lipid, and protein metabolism as a result of insufficient insulin function. Diabetes risk factors are related to a person's diabetes status. Various machine learning approaches have become an alternative in predicting diabetes status. However, in many cases, the available data is not sufficiently balanced within the data classes. The existence of data imbalance can cause prediction results to be inaccurate. The research objective of this paper is to address the problem of data imbalance and compare the performance of the model in predicting diabetes status. In general, methods such as the Synthetic Minority Over-sampling Technique (SMOTE) and Adaptive Synthetic (ADASYN) can be used to balance the data. The balanced Pima Indian Diabetes data is then predicted using machine learning methods such as the Bagging, Random Forest, and XGBoost methods. This research is expected to provide the best model that can classify the diabetes status of women of Pima Indian descent.*

Kata Kunci: Diabetes, Data Tidak Seimbang, *Machine Learning*

*penulis korespondensi

1. Pendahuluan

Diabetes merupakan suatu kelompok penyakit metabolik dengan karakteristik hiperglikemia yang terjadi karena kelainan ginjal, syaraf, jantung, dan pembuluh darah. Diabetes dapat menyerang beberapa organ tubuh yang mengakibatkan berbagai macam keluhan. Penyakit diabetes adalah penyakit yang ditandai oleh kadar glukosa darah yang melebihi batas normal, yang disebabkan oleh kurangnya hormon insulin yang dihasilkan oleh pankreas sehingga dapat menurunkan kadar gula darah [1].

Perkembangan teknologi komputasi dan algoritma *Machine Learning* (ML) berimplikasi pada pesatnya perkembangan teknologi di berbagai bidang, termasuk bidang penelitian biogenetika. Teknologi ML memanfaatkan komputer untuk melakukan proses pembelajaran dari data dan menghasilkan model prediksi. Salah satu metode yang umum digunakan dalam ML adalah *Decision Tree* (DT). DT mampu mengekstraksi informasi dari kumpulan data menjadi pengetahuan yang intuitif dan mudah dipahami [2].

Hambatan dari DT adalah nilai prediksi yang lemah yang disebabkan oleh ketersediaan data latih. Metode ensemble adalah algoritma pembelajaran yang dapat digunakan untuk mengatasi hal tersebut. Dalam beberapa penelitian menunjukkan bahwa klasifikasi dan prediksi dengan algoritma ensemble yang ditangani dengan baik umumnya dapat menghasilkan akurasi dan stabilitas yang lebih tinggi daripada menggunakan satu algoritma saja [3,4,5].

Namun dalam hal klasifikasi, terkadang terjadi masalah ketidakseimbangan data, di mana informasi pada satu kelas tidak sebanding (data minoritas) dari kelas lainnya (data mayoritas). Metode *Synthetic Minority Over-sampling Technique* (SMOTE) dan *Adaptive Synthetic* (ADASYN) merupakan salah satu teknik untuk mengatasi masalah ketidakseimbangan data. SMOTE merupakan algoritma dengan pendekatan *over-sampling*, yaitu data buatan untuk kelas data minoritas dibangkitkan agar terjadi keseimbangan proporsi kelas data mayor dan minor [6]. Sedangkan, ADASYN merupakan metode pendekatan pengambilan sampel yang menggunakan bobot distribusi data pada kelas minoritas berdasarkan tingkat kesulitan belajarnya, sehingga data sintesis yang dihasilkan dari kelas minoritas memiliki tingkat kesulitan belajar yang lebih tinggi dibandingkan dengan data minoritas itu sendiri [7].

2. Metode Penelitian

2.1. *Sumber Data*

Data yang digunakan dalam penelitian ini adalah data sekunder, yaitu data diabetes perempuan berusia minimal 21 tahun keturunan Indian Pima, yang berasal dari National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK). Prediktor yang digunakan pada studi ini bertipe numerik. Prediktor-prediktor tersebut terdiri dari Pregnancies (X1), Glucose (X2), BloodPressure (X3), Triceps SkinThickness (X4), Insulin (X5), BMI (X6), DiabetesPedigree (X7), dan Age (X9). Proporsi kelas dalam data yang digunakan adalah 65.10% atau 500 pengamatan yang Tidak Diabetes (*Not Diabetes*) dan 34.90% atau 268 pengamatan yang terkena Diabetes.

2.2. *Data Preparasi*

Pada tahap preparasi ini dilakukan penyaringan data untuk memilih data yang mempunyai nilai atau memilih data sesuai dengan kebutuhan. Pada data diabetes Indian Pima, terdapat data yang bernilai 0 yang diasumsikan sebagai nilai NA (kosong) sehingga dilakukan estimasi terhadap nilai kosong menggunakan titik pusatnya. Hal ini dilakukan untuk menghindari hilang informasi dalam data di mana data sedari awal terbilang cukup kecil.

Selain itu, tingginya dimensi suatu data juga menjadi salah satu masalah yang perlu diperhatikan. Oleh karena itu, reduksi dimensi dapat dilakukan dengan melakukan seleksi fitur (prediktor). Dalam paper ini, pendekatan analisis korelasi dan uji t statistik digunakan. Prediktor yang memiliki korelasi yang tinggi dengan prediktor lainnya akan dikeluarkan salah satunya dengan mempertimbangkan hasil pengujian t statistik. Prediktor yang paling informatif dan berkontribusi dalam proses klasifikasi status diabetes diukur berdasarkan nilai p-value dari uji t statistik. Prediktor yang digunakan diharapkan akan memberikan kontribusi maksimal dalam proses.

2.3. *Data Tidak Seimbang*

Data tidak seimbang (*Imbalance Data*) terjadi ketika ada satu atau lebih kelas yang mendominasi keseluruhan data sebagai kelas mayoritas dan kelas lainnya merupakan kejadian langka sebagai kelas minoritas. Data tidak seimbang akan menghasilkan suatu akurasi prediksi klasifikasi yang baik terhadap kelas mayoritas, sedangkan pada kelas minoritas akurasi yang dihasilkan kurang baik. Sulitnya mendapatkan model prediksi yang baik dan bermakna pada kelas minor karena adanya ketidakcukupan informasi [8].

2.3.1. *Synthetic Minority Over-sampling Technique*

Salah satu metode yang dapat digunakan dalam menangani pengaruh dari sedikitnya informasi mengenai kelas minoritas dalam suatu gugus data adalah *Synthetic Minority Over-sampling Technique* (SMOTE). Metode ini diperkenalkan oleh [9]. SMOTE merupakan algoritma dengan pendekatan over-sampling yaitu menambah jumlah data pengamatan pada kelas minoritas [10]. Teknik SMOTE ini dapat digunakan dalam penanganan data tidak seimbang. Data buatan untuk kelas data minoritas dibangkitkan untuk menyeimbangkan proporsi kelas data mayor dan minor [11]. Dalam membangkitkan data buatan tersebut, k-tetangga terdekat digunakan [9].

Berikut algoritma dalam metode SMOTE:

(1) Menentukan K tetangga terdekat (*nearest neighbors*) yang digunakan.
(2) Menghitung jarak antar setiap amatan minoritas menggunakan Euclidean dengan formula sebagai berikut:

$$d(x_i, y_j) = \sqrt{\Sigma_{j=1}^{n}(y_j - x_i)^2}. \tag{2.1}$$

Nilai $d(x_i, y_i)$ adalah jarak antara amatan minoritas x ke-i dan amatan minoritas y ke-j, n adalah jumlah amatan minoritas.

(3) Memilih secara acak amatan minoritas y_k dari K amatan terdekat dengan amatan minoritas x ke-i.

(4) Membangkitkan sebanyak jumlah kelipatan (persentase) yang telah ditentukan sampel sintetik baru menggunakan formula berikut:

$$S_i = x_i + (y_k - x_i)\lambda. \tag{2.2}$$

Dalam persamaan di atas, λ adalah bilangan acak antara $0-1$, S_i adalah sampel sintetik baru, x_i dan y_k adalah dua sampel dalam lingkungan ketetanggaan yang sama.

2.3.2. *Adaptive Synthetic*

Adaptive Synthetic (ADASYN) merupakan metode pendekatan untuk sampling pada kasus data yang tidak seimbang [7]. Ide utama dari ADASYN adalah menggunakan bobot distribusi untuk data pada kelas minoritas berdasarkan pada tingkat kesulitan pemahamannya, sehingga data sintesis dihasilkan dari kelas minoritas yang sulit untuk dipahami dibandingkan dengan data minoritas yang lebih mudah untuk dipahami. ADASYN meningkatkan pemahaman dengan dua cara. Pertama, mengurangi bias yang diakibatkan oleh ketidakseimbangan kelas dan kedua secara adaptif menggeser batas keputusan klasifikasi terhadap kesulitan data.

Berikut algoritma dalam metode ADASYN:

(1) Data keseluruhan D dengan m sampel $\{X_i, y_i\}$, $i = 1, 2, \cdots, m$, dimana m adalah jumlah sampel keseluruhan, X_i adalah sampel pada ruang fitur dimensi n, $y_i \in Y = \{1, -1\}$ adalah label kelas identitas terikat dengan X_i dan n adalah jumlah fitur pada data.

(2) Data minoritas $D_{minor} \in D$ dengan m_s sampel $\{X_{minor_i}, y_{minor_i}\}$, $i = 1, 2, \cdots, m_s$ dimana m_s adalah jumlah sampel minoritas, m_l adalah jumlah sampel mayoritas, X_{minor_i} adalah sampel kelas minoritas pada ruang fitur dimensi n dan y_{minor_i} adalah label kelas menoritas. Oleh karena itu, $m_s \leq m_l$ dan $m_s + m_l = m$.

(3) Hitung derajat kelas tidak seimbang:

$$d = \frac{m_s}{m_l}, \tag{2.3}$$

dimana $d \in (0, 1]$.

(4) Jika $d < d_{th}$ maka (d_{th} adalah batas toleransi maksimum derajat kelas tidak seimbang):

(a) Hitung jumlah sampel data sintetik (G) yang dibutuhkan untuk dibangkitkan pada kelas minoritas:

$$G = (m_l - m_s) \times \beta, \tag{2.4}$$

dimana $\beta \in [0, 1]$ adalah parameter yang digunakan untuk menentukan level keseimbangan yang diinginkan setelah pembangkitan data sintetik. Jika $\beta = 1$ maka data sepenuhnya seimbang dibuat setelah proses generalisasi.

(b) Untuk setiap sampel X_{minor_i}, tentukan K tetangga terdekat (*nearest neighbors*) dan hitung r_i. Setelah langkah ini, setiap sampel minoritas akan terkait dengan tetangga yang berbeda.

$$r_i = \frac{\#majority}{k}, \tag{2.5}$$

dimana r_i adalah tingkat kandungan mayoritas dari tetangga sampel minoritas ke-i dan k adalah jumlah ketetanggaan. Nilai r_i yang lebih tinggi, mengandung banyak tetangga sampel kelas mayoritas dan semakin sulit untuk dipahami (*learn*). Perhatikan gambar dari visualisasi untuk langkah ini.

(c) Normalisasi nilai r_i sehingga jumlah dari seluruh r_i adalah 1.

$$\hat{r}_i = \frac{r_i}{\Sigma r_i} \tag{2.6}$$

$$\Sigma \hat{r}_i = 1, \tag{2.7}$$

dimana $\hat{r}_i$ adalah hasil normalisasi dari tingkat kandungan data mayoritas. Langkah ini dilakukan untuk memudahkan langkah selanjutnya.

(d) Hitung jumlah sampel sintetik yang akan dibangkitkan setiap sampel minoritas.

$$G_i = G\hat{r}_i, \tag{2.8}$$

dengan r_i bernilai tinggi untuk tetangga sekitar yang didominasi oleh sampel kelas mayoritas, banyak sampel sintetik kelas minoritas yang akan dibangkitkan.

(e) Bangkitkan G_i data untuk setiap data minoritas. Pertama, ambil sampel minoritas, X_{minor_i}. Kemudian, pilih secara acak tetangga dengan label kelas minoritas dari X_{minor_i}, $X_{minor_{z_i}}$. Sampel sintetik baru akan dihitung menggunakan:

$$S_i = X_{minor_i} + (X_{minor_{z_i}} - X_{minor_i})\lambda. \tag{2.9}$$

Pada persamaan diatas, λ adalah bilangan acak antara $0 - 1$, S_i adalah sampel sintetik baru, X_{minor_i} dan $X_{minor_{z_i}}$ adalah dua sampel minoritas dalam lingkungan ketetanggan yang sama.

2.4. *Metode Machine Learning Ensemble*

2.4.1. *Metode Bagging*

Bagging atau *bootstrap aggregating* adalah salah satu metode ensemble yang digunakan untuk meningkatkan performa klasifikasi [12]. *Bagging* menggunakan sebagian dari data untuk menghasilkan kumpulan data pelatihan untuk proses pembelajaran (*learning*). *Bagging* baik digunakan untuk kasus klasifikasi dan regresi. Dalam kasus regresi, untuk menjadi lebih kuat dapat mengambil rata-rata ketika menggabungkan hasil prediksi, sedangkan dalam kasus klasifikasi mengambil kelas yang populer ketika menggabungkan hasil prediksi.

Algoritma *Bagging* untuk klasifikasi yaitu:

(1) Data D dengan n sampel $\{X_i, y_i\}$, $i = 1, 2, \cdots, n$, dimana n adalah jumlah sampel keseluruhan, X_i adalah sampel pada ruang fitur dimensi p, $y_i \in Y = \{1, -1\}$ adalah label kelas identitas terikat dengan X_i dan p adalah jumlah fitur pada data.
(2) Menentukan nilai B dimana B adalah jumlah pohon.
(3) Untuk $b = 1$ sampai perulangan ke-B dilakukan:
    (a) Lakukan pengambilan secara acak sampel bootstrap $Z^* \in D$ dengan ukuran n dari data latih.
    (b) Buat pohon (*tree*) T_b dari data sampel bootstrap dengan cara melakukan langkah dibawah ini secara rekursif untuk setiap simpul dari pohon (*tree*), sampai paling tidak ukuran minimal simpul dicapai atau sampai data sudah tidak bisa dibagi lagi.
        i. Ambil variabel dengan penyekat terbaik di antara semua variabel.
        ii. Menentukan simpul akhir.
(4) Keluaran dari model Bagging adalah label yang ditentukan dengan memilih label kelas terbanyak (sistem voting) dari B pohon yang telah dibentuk.

2.4.2. *Metode Random Forest*

Random Forest merupakan sebuah metode ensemble. Metode ensemble merupakan cara untuk meningkatkan akurasi metode klasifikasi dengan cara mengkombinasikan metode klasifikasi [13]. *Random Forest* diawali dengan teknik dasar *data mining* yaitu pohon keputusan (*decision tree*). Pada pohon keputusan data dimasukkan pada bagian atas (*root*) kemudian turun ke bagian bawah (*leaf*) untuk menentukan data kelasnya. *Random Forest* adalah pengklasifikasi yang terdiri dari kumpulan pengklasifikasi pohon terstruktur di mana masing-masing pohon memberikan kelas, lalu memilih kelas yang populer [14].

Algoritma *Random Forest* untuk klasifikasi yaitu:

(1) Data D dengan n sampel $\{X_i, y_i\}$, $i = 1, 2, \cdots, n$, dimana n adalah jumlah sampel keseluruhan, X_i adalah sampel pada ruang fitur dimensi p, $y_i \in Y = \{1, -1\}$ adalah label kelas identitas terikat dengan X_i dan p adalah jumlah fitur pada data.
(2) Menentukan nilai B dan m, dimana B adalah jumlah pohon dan m adalah jumlah variabel yang akan dipilih secara acak saat pembuatan pohon.
(3) Untuk $b = 1$ sampai perulangan ke-B dilakukan:
    (a) Lakukan pengambilan secara acak sampel bootstrap $Z^* \in D$ dengan ukuran n dari data latih.
    (b) Buat pohon (*tree*) T_b dari data sampel bootstrap dengan cara melakukan langkah dibawah ini secara rekursif untuk setiap simpul dari pohon (*tree*), sampai paling tidak ukuran minimal simpul dicapai atau sampai data sudah tidak bisa dibagi lagi.
        i. Pilih m variabel secara acak.
        ii. Ambil variabel dengan penyekat terbaik di antara m variabel.

iii. Menentukan simpul akhir.

(4) Keluaran dari model adalah *Random Forest* $\{T_b\}_1^B$, dimana $\{T_b\}_1^B$ adalah kumpulan dari pohon-pohon yang telah dibangun yang dimulai dari pohon pertama sampai pohon ke-B. Untuk melakukan prediksi pada titik baru, yaitu dengan memilih label kelas terbanyak (sistem voting) dari B pohon yang telah dibentuk.

2.4.3. *Metode Extreme Gradient Boosting*

Extreme Gradient Boosting (XGBoost) adalah salah satu bentuk implementasi dari *boosting*, yaitu kumpulan pohon keputusan, di mana setiap pohon keputusan saling bergantung dengan pohon keputusan sebelumnya. Sama halnya dengan *Gradient Boosting*, XGBoost memanfaatkan pohon keputusan sebagai model utama dan membangun ekspansi aditif dari objective function untuk meminimalkan kesalahan prediksi (*loss function*). Namun, XGBoost memiliki skalabilitas yang lebih baik dan mampu melakukan optimasi 10 kali lebih cepat dibandingkan dengan *Gradient Boosting* [15].

Objective function digunakan untuk mengukur tingkat kemampuan model dalam mengenali data latih. Dalam *objective function* terdapat dua karakteristik, yaitu *training loss* dan *regularization term* sebagai berikut:

$$obj(\theta) = L(\theta) + \Omega(\theta). \tag{2.10}$$

Nilai L menunjukkan *training loss function* yang mengukur kemampuan model dalam memprediksi data latih dan Ω merupakan *regularization term* yang akan mengatur tingkat kompleksitas model untuk menghindari terjadinya *over fitting*. Secara umum *training loss* didefinisikan sebagai berikut:

$$L(\theta) = \Sigma_{i=1}^{n} l(y_i, \hat{y}_i). \tag{2.11}$$

Nilai y_i merupakan data aktual dan $\hat{y}_i$ merupakan hasil prediksi model. Dalam paper ini diterapkan *cross entropy loss* sebagai berikut:

$$L(\theta) = -[y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)]. \tag{2.12}$$

Adapun *regularization term* yang digunakan untuk mengontrol kompleksitas pohon keputusan adalah sebagai berikut:

$$\Omega(\theta) = \gamma T + \frac{1}{2}\lambda ||w||^2. \tag{2.13}$$

Nilai T merupakan jumlah daun (*leaf*) dari pohon keputusan *(tree)* dan w merupakan nilai luaran (output) dari *leaf*. Nilai γ akan mengatur nilai menimum dari *loss reduction gain* pada saat membuat simpul (*node*) yang mengakibatkan pohon yang terbentuk akan semakin sederhana apabila γ memiliki nilai yang besar.

3. Hasil dan Pembahasan

3.1. *Deskripsi Data Diabetes Indian Pima*

Data yang digunakan dalam paper ini adalah data sekunder, yaitu data diabetes perempuan berusia minimal 21 tahun keturunan Indian Pima yang berasal dari

National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK). Data tersebut dapat diakses melalui Kaggle dengan link `https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database`.

Tabel 1. Deskripsi Data Diabetes Indian Pima

Fitur (Variabel)	Keterangan	Tipe Fitur
Pregnancies (X1)	Jumlah berapa kali pasien melahirkan	Numerik
Glucose (X2)	Konsentrasi glukosa plasma 2 jam dalam tes toleransi glukosa oral	Numerik
Blood Pressure (X3)	Tekanan darah diastolik (mm Hg)	Numerik
Skin Thickness (X4)	Ketebalan lipatan kulit trisep (mm)	Numerik
Insulin (X5)	Insulin serum dalam 2 jam (mu U/ml)	Numerik
BMI (X6)	Indeks Massa Tubuh	Numerik
Diabetes Pedigree (X7)	Fungsi perhitungan kemungkinan diabetes berdasarkan keturunan	Numerik
Patient Age (X8)	Umur pasien	Numerik
Status Diabetes (Y)	Status diabetes pasien	Kategorik

Dalam Tabel 1 ditunjukkan bahwa data terdiri dari 9 variabel yang meliputi 1 variabel dependen dan 8 variabel independen. Variabel-variabel independen tersebut selanjutnya akan digunakan untuk analisis lebih lanjut dalam melakukan pemodelan *Machine Learning*.

3.2. *Statistik Deskriptif Sederhana*

Statistik deskriptif sederhana dapat dilihat dari nilai minimum, maksimum, rata-rata dan standar deviasi dari masing-masing variabel dengan tujuan eksplorasi data.

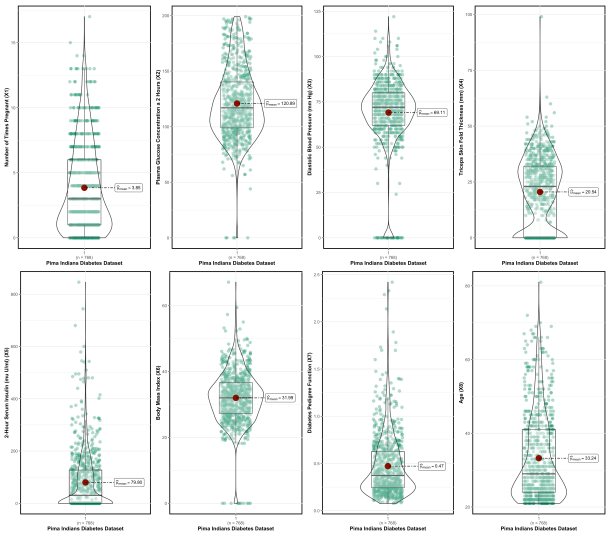
Tabel 2. Deskripsi Data Diabetes Indian Pima

Fitur (Variabel)	Min.	Max.	Mean	Std.
Pregnancies (X1)	0.00	17.00	3.85	3.37
Glucose (X2)	0.00	199.00	120.90	31.97
Blood Pressure (X3)	0.00	122.00	69.11	19.36
Skin Thickness (X4)	0.00	99.00	20.54	15.95
Insulin (X5)	0.00	846.00	79.80	115.24
BMI (X6)	0.00	67.10	31.99	7.88
Diabetes Pedigree (X7)	0.08	2.42	0.47	0.33
Patient Age (X8)	21.00	81.00	33.24	11.76
Status Diabetes (Y)	N: 768, D: 268 (34.90%), TD: 500 (65.10%)			

Pada Tabel 2 terlihat bahwa kelas dalam data diabetes adalah 65.10% atau 500 pengamatan yang Tidak Diabetes dan 34.90% atau 268 pengamatan yang terkena Diabetes. Dari Tabel 2 juga terlihat bahwa terdapat fitur (variabel) yang memiliki nilai yang tidak logis sehingga perlu dilakukan eksplorasi yang lebih jauh.

3.3. *Pendeteksian Data Anomali*

Data yang dikumpulkan terkadang tidak konsisten atau tidak sesuai dengan harapan, sehingga data yang akan diolah harus dieksplorasi terlebih dahulu untuk mengetahui apakah terdapat masalah dalam data.



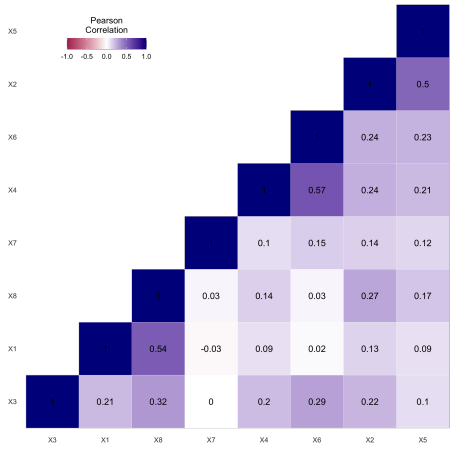
Gambar 1. Violin Plot dan Boxplot Distribusi Data

Pada Gambar 1 terlihat bahwa prediktor Glucose (X2), Blood Pressure (X3), Skin Thickness (X4), Insulin (X5), dan BMI (X6) mengandung nilai yang tidak sesuai, yaitu 0 (nol). Sehingga diperlukan perlakuan khusus untuk menangani hal tersebut. Salah satu teknik untuk menangani hal tersebut adalah dengan menganggap nilai 0 sebagai nilai kosong (NA) yang diasumsikan diakibatkan oleh kesalahan dalam pengumpulan data, kemudian melakukan estimasi menggunakan titik pusatnya.

3.4. *Seleksi Fitur (Prediktor)*

Dalam banyak kasus, sering ditemukan prediktor yang mengandung informasi yang hampir sama dengan prediktor lain. Hal ini dapat menyebabkan kinerja dari model yang digunakan tidak dapat bekerja dengan maksimal dan memberikan hasil yang kurang memuaskan. Oleh karena itu, prediktor yang memiliki informasi yang sama harus dikeluar sebelum masuk ke tahap pemodelan untuk mengurangi dampak dari multikolinieritas.

Salah satu teknik yang digunakan untuk mengukur tingkat kesamaan informasi dalam prediktor adalah matriks korelasi. Matriks korelasi bertujuan untuk melihat hubungan antar prediktor, di mana prediktor yang memiliki hubungan yang kuat diasumsikan memiliki terkaitan yang kuat. Dengan kata lain, prediktor yang memiliki hubungan yang kuat secara statistik dapat dikatakan memiliki informasi yang sama.

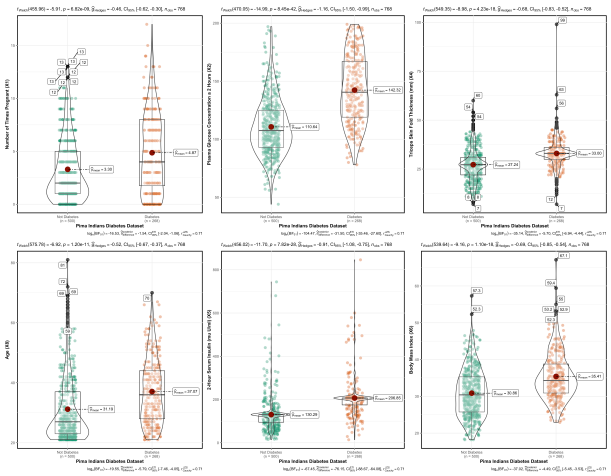


Gambar 2. Heatmap Matriks Korelasi Prediktor

Gambar 2 menunjukkan bahwa terdapat beberapa prediktor yang memiliki hubungan yang cukup kuat. Pregnancies (X1) dan Patient Age (X8) memiliki hubungan yang kuat positif yaitu 0.54, di mana apabila salah satu dari kedua prediktor meningkatkan maka prediktor lain juga ikut meningkat. Hal ini jelas secara logis di mana semakin bertambah umur seorang wanita maka peluang semakin sering hamil juga meningkat dan begitupun sebaliknya. Hal ini juga berlaku untuk prediktor yang memiliki hubungan kuat positif lainnya seperti Skin Thickness (X4) dan BMI (X6) yang memiliki nilai korelasi positif sebesar 0.57, serta Glucose (X2) dan Insulin (X5) yang memiliki nilai korelasi positif sebesar 0.5.

Setelah mengetahui adanya korelasi yang terjadi antar prediktor, dilakukan uji perbedaan rataan menggunakan uji t statistik untuk melihat perbedaan rataan antar kelas (status diabetes). Apabila terdapat perbedaan yang signifikan antar kelas maka terdapat pengaruh signifikan yang disebabkan oleh prediktor terkait. Hal ini akan diterapkan pada prediktor yang saling berkorelasi yaitu Pregnancies (X1) dan Patient Age (X8), Skin Thickness (X4) dan BMI (X6), serta Glucose (X2) dan Insulin (X5).

Pada Gambar 3 terlihat bahwa ketiga pasang prediktor yaitu memiliki pengaruh yang signifikan terhadap Status Diabetes (Y) seseorang data Diabetes Indian Pima. Berdasarkan ketiga prediktor yang saling berkorelasi, maka dipilih prediktor Insulin (X5), BMI (X6) dan Patient Age (X8) untuk digunakan ke dalam analisis selanjutnya. Sehingga, prediktor-prediktor yang digunakan pada tahap selanjutnya adalah

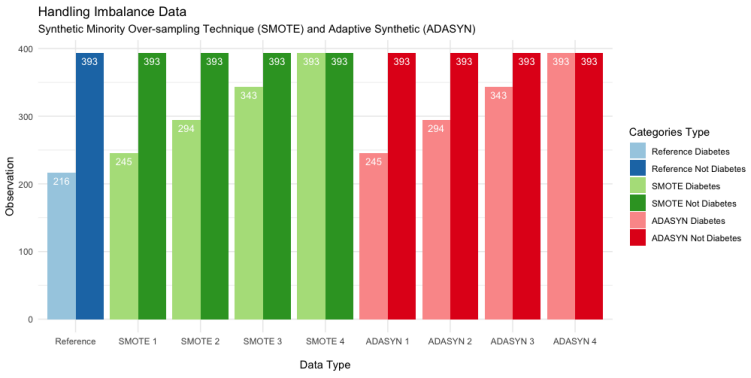


Gambar 3. Violin Plot dan Boxplot Distribusi Data Berdasarkan Status Diabetes.

Blood Pressure (X3), Insulin (X5), BMI (X6), Diabetes Pedigree (X7) dan Patient Age (X8).

3.5. *Penanganan Data Tidak Seimbang*

Selanjutnya, melalukan penangan ketidakseimbangan data dalam data Diabetes Indian Pima. Pada paper ini, data diseimbangkan menggunakan pendekatan oversampling SMOTE dan ADASYN dengan nilai persentase pembangkitan data sebesar 25%, 50%, 75% dan 100%. Hal ini dilakukan agar banyaknya data yang dibangkitkan dapat diketahui secara optimal dan terhindar dari pembangkitan data yang berlebihan. Proses pembangkitan data dalam paper ini mengikuti prosedur pembangkitan data metode SMOTE dan ADASYN dengan dukungan aplikasi pemrograman R.



Gambar 4. Penanganan Data Tidak Seimbang.

Pada Gambar 4 terlihat bahwa jumlah pengamatan dari masing-masing kelas dalam data Diabetes Indian Pima sebelum dan setelah diterapkannya metode SMOTE dan ADASYN. Dalam Gambar 4 juga terlihat bahwa total data yang akan dimodelkan sebagai 9 jenis yang terdiri dari 1 data asli, 4 data hasil pembangkitan menggunakan metode SMOTE dengan persentase bangkitan yang berbeda dan 4 jenis data hasil pembangkitan menggunakan metode SMOTE dengan persentase bangkitan yang berbeda pula.

3.6. *Pemodelan Machine Learning*

Pada tahap Pemodelan *Machine Learning*, metode yang digunakan adalah metode *Bagging*, *Random Forest* dan XGBoost. Ketiga metode *machine learning* digunakan untuk melihat metode yang paling cocok dalam mengklasifikan data Diabetes Indian Pima sebelum dan setelah data diseimbangkan. Performa model akan diukur menggunakan metrik akurasi, sensitivitas, dan spesifikasi. Namun perlu diketahui bahwa paper ini akan berfokus pada nilai spesifikasi dikarenakan adanya pertimbangan resiko kesalahan diagnosa pada pasien yang benar-benar terkena diabetes lebih tinggi.

3.6.1. *Metode Bagging*

Metode Bagging yang digunakan dalam paper ini memiliki dua parameter, yaitu max_depth dan m_final, dimana max_depth adalah batas maksimum kedalaman setiap pohon keputusan akan yang dibangun dan m_final adalah jumlah pohon keputusan (perulangan) yang akan dibangun.

Tabel 3. Spesifikasi Model Bagging dengan Jenis Data dan Kombinasi Parameter Berbeda

md	mf	Ref	IS25	IS50	IS75	IS100	IA25	IA50	IA75	IA100
3	50	84.62%	84.62%	86.54%	84.62%	84.62%	84.62%	84.62%	86.54%	86.54%
5	50	82.69%	84.62%	88.46%	80.77%	86.54%	82.69%	88.46%	84.62%	96.15%
7	50	84.62%	84.62%	86.54%	82.69%	84.62%	84.62%	88.46%	86.54%	94.23%
3	75	82.69%	84.62%	86.54%	84.62%	84.62%	84.62%	84.62%	88.46%	92.31%
5	75	82.69%	84.62%	86.54%	84.62%	84.62%	82.69%	86.54%	84.62%	92.31%
7	75	84.62%	82.69%	86.54%	82.69%	86.54%	82.69%	86.54%	84.62%	96.15%
3	100	84.62%	84.62%	86.54%	86.54%	84.62%	84.62%	86.54%	88.46%	88.46%
5	100	84.62%	82.69%	88.46%	82.69%	84.62%	82.69%	84.62%	84.62%	94.23%
7	100	82.69%	84.62%	88.46%	84.62%	84.62%	84.62%	86.54%	86.54%	92.31%
3	125	82.69%	82.69%	88.46%	84.62%	84.62%	84.62%	84.62%	88.46%	88.46%
5	125	82.69%	84.62%	84.62%	82.69%	84.62%	84.62%	86.54%	84.62%	92.31%
7	125	84.62%	82.69%	84.62%	82.69%	84.62%	82.69%	86.54%	84.62%	96.15%

Tabel 3 adalah kombinasi dari parameter yang digunakan dalam paper ini beserta performa dari model yang dibangun berdasarkan kombinasi parameter tersebut. Dalam Tabel 3 terlihat bahwa model Bagging dengan parameter max_depth

sebesar 5 dan m_final sebesar 50 yang dibangun berdasarkan data hasil pembangkitan dengan metode ADASYN 100% (sepenuhnya seimbang) memiliki performa terbaik dibandingkan model Bagging lainnya. Hal ini ditunjukkan pada nilai spesifikasi yang sangat tinggi yaitu 96.15%.

3.6.2. *Metode Random Forest*

Metode *Random Forest* yang digunakan dalam paper ini juga memiliki dua parameter, yaitu m_try dan $ntree$, dimana m_try adalah jumlah fitur (variabel) acak yang akan digunakan dalam membentuk pohon keputusan dan $ntree$ adalah jumlah pohon keputusan (perulangan) yang akan dibangun. Berikut adalah kombinasi dari parameter yang digunakan dalam paper ini berserta performa dari model yang dibangun berdasarkan kombinasi parameter tersebut:

Tabel 4. Spesifikasi Model Random Forest dengan Jenis Data dan Kombinasi Parameter Berbeda

mt	nr	Ref	IS25	IS50	IS75	IS100	IA25	IA50	IA75	IA100
3	50	82.69%	82.69%	80.77%	82.69%	88.46%	82.69%	84.62%	80.77%	84.62%
5	50	82.69%	82.69%	82.69%	82.69%	84.62%	80.77%	80.77%	82.69%	84.62%
7	50	82.69%	80.77%	86.54%	84.62%	82.69%	82.69%	82.69%	84.62%	88.46%
3	75	80.77%	82.69%	82.69%	82.69%	84.62%	84.62%	82.69%	84.62%	84.62%
5	75	82.69%	86.54%	82.69%	82.69%	84.62%	84.62%	82.69%	82.69%	84.62%
7	75	82.69%	82.69%	82.69%	80.77%	82.69%	82.69%	80.77%	86.54%	84.62%
3	100	82.69%	80.77%	86.54%	82.69%	86.54%	80.77%	84.62%	84.62%	86.54%
5	100	80.77%	84.62%	84.62%	82.69%	84.62%	84.62%	80.77%	80.77%	86.54%
7	100	82.69%	82.69%	84.62%	80.77%	82.69%	82.69%	84.62%	82.69%	86.54%
3	125	82.69%	82.69%	84.62%	84.62%	84.62%	84.62%	82.69%	84.62%	84.62%
5	125	80.77%	86.54%	86.54%	84.62%	82.69%	82.69%	82.69%	84.62%	86.54%
7	125	82.69%	82.69%	82.69%	82.69%	82.69%	82.69%	82.69%	82.69%	88.46%

Tabel 4 adalah kombinasi dari parameter yang digunakan dalam paper ini berserta performa dari model yang dibangun berdasarkan kombinasi parameter tersebut. Dalam Tabel 4 terlihat bahwa model *Random Forest* dengan parameter m_try sebesar 3 dan $ntree$ sebesar 50 yang dibangun berdasarkan data hasil pembangkitan dengan metode SMOTE 100% (sepenuhnya seimbang) memiliki performa terbaik dibandingkan model *Random Forest* lainnya. Hal ini ditunjukkan pada nilai spesifikasi yang sangat tinggi yaitu 88.46%.

3.6.3. *Metode Extreme Gradient Boosting (XGBoost)*

Metode XGBoost yang digunakan dalam paper ini memiliki tiga parameter, yaitu eta, max_depth dan $nround$, dimana eta adalah *learning rate* atau tingkat pembelajaran yang digunakan dalam setiap pohon keputusan yang dibangun, max_depth adalah batas maksimum kedalaman setiap pohon keputusan akan yang dibangun

dan *nround* adalah jumlah pohon keputusan (perulangan) yang akan dibangun. Berikut adalah kombinasi dari parameter yang digunakan dalam paper ini berserta performa dari model yang dibangun berdasarkan kombinasi parameter tersebut.

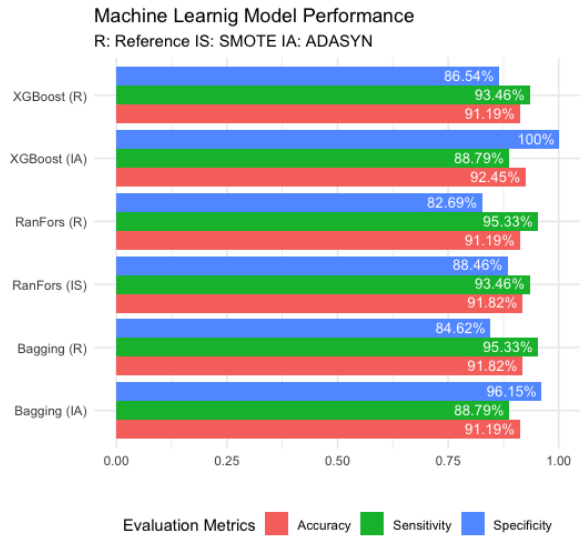
Tabel 5. Spesifikasi Model Extreme Gradient Boosting dengan Jenis Data dan Kombinasi Parameter Berbeda

lr	md	nr	Ref	IS25	IS50	IS75	IS100	IA25	IA50	IA75	IA100
0.01	3	50	84.62%	84.62%	86.54%	86.54%	88.46%	84.62%	88.46%	88.46%	100%
0.05	3	50	84.62%	86.54%	86.54%	86.54%	86.54%	86.54%	86.54%	86.54%	92.31%
0.01	5	50	78.85%	84.62%	80.77%	86.54%	88.46%	82.69%	86.54%	88.46%	82.69%
0.05	5	50	82.69%	86.54%	84.62%	88.46%	88.46%	84.62%	88.46%	88.46%	88.46%
0.01	7	50	75.00%	78.85%	78.85%	82.69%	84.62%	78.85%	88.46%	84.62%	80.77%
0.05	7	50	78.85%	84.62%	84.62%	86.54%	86.54%	82.69%	86.54%	86.54%	84.62%
0.01	3	75	84.62%	86.54%	86.54%	84.62%	90.38%	86.54%	88.46%	88.46%	92.31%
0.05	3	75	84.62%	84.62%	86.54%	84.62%	86.54%	84.62%	84.62%	86.54%	90.38%
0.01	5	75	80.77%	84.62%	84.62%	86.54%	88.46%	86.54%	86.54%	86.54%	86.54%
0.05	5	75	80.77%	82.69%	84.62%	88.46%	88.46%	82.69%	86.54%	86.54%	88.46%
0.01	7	75	76.92%	80.77%	80.77%	82.69%	84.62%	80.77%	84.62%	86.54%	80.77%
0.05	7	75	80.77%	82.69%	80.77%	88.46%	86.54%	82.69%	84.62%	84.62%	82.69%
0.01	3	100	84.62%	86.54%	86.54%	86.54%	88.46%	84.62%	88.46%	88.46%	92.31%
0.05	3	100	84.62%	84.62%	86.54%	86.54%	88.46%	84.62%	80.77%	86.54%	90.38%
0.01	5	100	84.62%	84.62%	84.62%	88.46%	90.38%	86.54%	88.46%	88.46%	86.54%
0.05	5	100	80.77%	80.77%	86.54%	88.46%	88.46%	84.62%	84.62%	84.62%	86.54%
0.01	7	100	78.85%	86.54%	82.69%	84.62%	86.54%	82.69%	82.69%	84.62%	82.69%
0.05	7	100	82.69%	80.77%	80.77%	88.46%	86.54%	82.69%	84.62%	82.69%	84.62%
0.01	3	125	86.54%	86.54%	88.46%	86.54%	88.46%	84.62%	86.54%	88.46%	92.31%
0.05	3	125	82.69%	82.69%	84.62%	86.54%	88.46%	82.69%	80.77%	86.54%	88.46%
0.01	5	125	84.62%	86.54%	84.62%	88.46%	88.46%	86.54%	88.46%	88.46%	88.46%
0.05	5	125	82.69%	82.69%	86.54%	86.54%	86.54%	84.62%	84.62%	84.62%	86.54%
0.01	7	125	80.77%	84.62%	84.62%	86.54%	86.54%	82.69%	86.54%	82.69%	82.69%
0.05	7	125	82.69%	78.85%	84.62%	86.54%	82.69%	80.77%	84.62%	78.85%	84.62%

Tabel 5 adalah kombinasi dari parameter yang digunakan dalam paper ini berserta performa dari model yang dibangun berdasarkan kombinasi parameter tersebut. Dalam Tabel 5 terlihat bahwa model XGBoost dengan parameter *eta* sebesar 0.01, *max_depth* sebesar 3 dan *nround* sebesar 50 yang dibangun berdasarkan data hasil pembangkitan dengan metode ADASYN 100% (sepenuhnya seimbang) memiliki performa terbaik dibandingkan model XGBoost lainnya. Hal ini ditunjukkan pada nilai spesifikasi yang sangat tinggi yaitu 100%.

3.6.4. *Perbandingan Performa Model*

Setelah model terbaik dari masing-masing metode diperoleh, model-model tersebut akan dibandingkan satu sama lain untuk mengetahui metode terbaik dalam menyelesaikan kasus pengklasifisian diabetes Indian Pima.



Gambar 5. Performa Model Machine Learning.

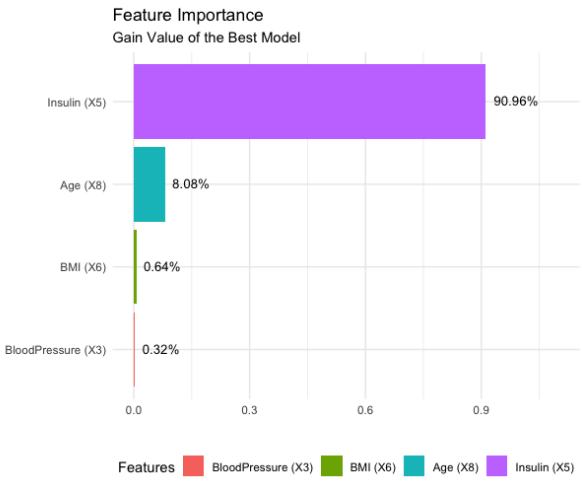
Pada Gambar 5 terlihat bahwa model XGBoost setelah penanganan ketidakseimbangan menggunakan metode ADASYN dengan persentase pembangkitan sebesar 100% (sepenuhnya seimbang) memiliki performa terbaik dibandingkan model lainnya. Hal ini ditunjukkan pada nilai spesifikasi yang sangat tinggi yaitu 100% (semua pasien kelas diabetes diklasifikasikan dengan benar) dengan nilai akurasi tertinggi yaitu 92.45%.

Apabila performa model diukur berdasarkan nilai akurasi, XGBoost tetap menjadi model dengan performa terbaik, walaupun nilai sensitivitas yang dari model XGBoost cukup rendah dibandingkan dengan model lain. Namun, seperti yang dijelaskan sebelumnya bahwa resiko kesalahan diagnosa pada pasien yang benar-benar terkena diabetes menjadi prioritas maka model XGBoost dengan penanganan ketidakseimbangan menggunakan metode ADASYN menjadi model dengan performa terbaik dibandingkan dengan model lainnya.

3.6.5. *Tingkat Kepentingan Fitur (Feature Importance)*

Kepentingan fitur (variabel) menunjukkan seberapa besar kontribusi masing-masing fitur terhadap prediksi model. Pada dasarnya, tingkat kepentingan fitur menentukan tingkat kegunaan dari variabel tertentu untuk model dan prediksi yang dilakukan. Nilai tingkat kepentingan fitur digambarkan menggunakan nilai Gain yang diper-

oleh pada saat pembentukan pohon dan kemudian distandarisasi kedalam bentuk persentase. Berdasarkan model XGBoost setelah penanganan ketidakseimbangan menggunakan metode ADASYN (model terbaik) diperoleh nilai Gain seperti pada Gambar 6.



Gambar 6. Performa Model Machine Learning.

Pada Gambar 6 terlihat bahwa fitur (variabel) Insulin (X5) memiliki tingkat kepentingan sebesar 90.96% dalam membentuk model klasifikasi. Sehingga dapat disimpulkan bahwa variabel Insulin (X5) memiliki kontribusi terbesar dalam memprediksi status diabetes seseorang dalam data Diabetes Indian Pima. Adapun fitur (variabel) Age (X8) sebagai fitur yang memiliki kontribusi terbesar ke-2 dalam memprediksi status diabetes, namun memiliki nilai gain yang jauh lebih kecil dibandingkan fitur (variabel) Insulin (X5) yaitu 8.08%. Hal ini menunjukkan bahwa Insulin (X5) sudah cukup memberikan kontribusi dalam menentukan status diabetes Indian Pima namun dalam beberapa kasus tertentu faktor Age (X8) juga dapat memberikan informasi yang penting sebelum status diabetes Indian Pima ditentukan.

4. Kesimpulan

Berdasarkan hasil penelitian diketahui bahwa ADASYN bekerja lebih baik dibandingkan SMOTE pada data. Selain itu, perbedaan performa antara *Bagging*, *Random Forest* dan XGBoost tidak begitu besar, namun di antara ketiga model, XGBoost memiliki performa terbaik dibandingkan *Bagging* dan *Random Forest*. Selain itu, dalam penelitian diketahui bahwa Insulin (X5) memberikan peran yang sangat besar dalam menentukan Status Diabetes dalam data Diabetes Indian Pima.

5. Ucapan Terima kasih

Penulis mengucapkan terima kasih kepada National Institute of Diabetes and Digestive and Kidney Diseases (NIDDK) yang telah menyediakan akses pada data.

Daftar Pustaka

[1] American Diabetes Association, 2015, Standards of medical care in diabetes-2015 abridged for primary care providers, *Clinical Diabetes* Vol. **33**(2) : 97 – 111

[2] Baros, R. C., Basgalupp, M. P., Carvalho, A. C. P. L. F., & Freitas, A. A., 2011, Towards the automatic design of decision tree induction algorithms, Dalam: *Proceedings of the 13th annual conference companion on Genetic and evolutionary computation*, 182 – 196

[3] Yang, P., Yang, Y. H., Zhou, B. B., & Zomaya, A. Y., 2010, A Review of Ensemble Methods in Bioinformatics: Including Stability of Feature Selection and Ensemble Feature Selection Methods, *Current Bioinformatics* Vol. **5**(4) : 296 – 308

[4] Aqsha, M., Thamrin, S. A., & Lawi, A., 2021, Combination of ADASYN-N and Random Forest in Predicting of Obesity Status in Indonesia: A Case Study of Indonesian Basic Health Research 2013, *Journal of Physics: Conference Series*, **2123** : 012039

[5] Rombe, Y., Thamrin, S. A., & Lawi, A., 2022, Application of Adaptive Synthetic Nominal and Extreme Gradient Boosting Methods in Determining Factors Affecting Obesity: A Case Study of Indonesian Basic Health Research Survey 2013, *Indonesian Journal of Statistics and Its Applications* Vol. **6**(2): 309 – 317

[6] Mustaqim, M., Warsito, B., & Surarso, B., 2019, Kombinasi *Synthetic Minority Oversampling Technique* (SMOTE) dan *Neural Network Back propagation* untuk menangani data tidak seimbang pada prediksi pemakaian alat kontrasepsi implan, *Jurnal Ilmiah Teknologi Sistem Informasi* Vol. **5**(2): 116 – 127

[7] He, H., Bai, Y., Gracia, E. A., & Li, S., 2008, ADASYN: Adaptive Synthetic Sampling Approach for Imbalanced Learning, Dalam: *2008 IEEE International Joint Conference on Neural Networks (IEEE World Congress on Computational Intelligence)*, 1322 – 1328

[8] Yap, B. W., Rani, K. A., Rahman, H. A. A., Fong, S., Khairudin, Z., & Abdullah, N. N., 2014, An Application of Oversampling, Undersampling, Bagging and Boosting in Handling Imbalanced Datasets, Dalam: *Herawan, T., Deris, M., Abawajy, J. (eds) Proceedings of the First International Conference on Advanced Data and Information Engineering (DaEng-2013)*, **285** : 13 – 22

[9] Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P., 2002, SMOTE: Synthetic Minority Over-sampling Technique, *Journal of Artificial Intelligence Research* Vol. **16**(1): 321 – 357

[10] Zhu, T., Lin, Y., & Liu, Y., 2017, Synthetic Minority Oversampling Technique for Multiclass Imbalance Problems, *Pattern Recognition* Vol. **72(C)** : 327 – 340

[11] Alghamdi, M., Al-Mallah, M., Keteyian, S., Brawner, C., Ehrman, J., & Sakr, S., 2017, Predicting diabetes mellitus using SMOTE and ensemble machine learning approach: The Henry Ford ExercIse Testing (FIT) project, *PLOS ONE* Vol. **12(7)** : e0179805

[12] Breiman, L., 1996, *Bagging Predictors In Machine Learning*, Kluwer Academic,

Boston

[13] Han, J., Kamber, M., & Pei, J., 2012, *Data Mining: Concept and Techniques*, Edisi ke-3, Elsevier Inc, USA

[14] Breiman, L., 2001, *Random Forests In Machine Learning*, Kluwer Academic, Boston

[15] Chen, T., & Guestrin, C., 2016, XGBoost: A Scalable Tree Boosting System, Dalam: *International Conference on Knowledge Discovery and Data Mining*, 785 – 794

[16] Patil, I., 2021, Visualizations with statistical details: The 'ggstatsplot' approach, *Journal of Open Source Software* Vol. **6**(61): 3167