



Perbandingan Teknik Penyeimbang Kelas Pada Multi-Layer Perceptron (MLP) Berbasis Backpropagation Untuk Klasifikasi Diabetes Mellitus

Robby Azhar, Siska Kurnia Gusti*, Iis Afrianty, Elvia Budianita

Fakultas Sains dan Teknologi, Prodi Teknik Informatika, Universitas Islam Negeri Sultan Syarif Kasim Riau, Pekanbaru, Indonesia

Email: ¹12150114654@students.uin-suska.ac.id, ^{2,*}siskakurniagusti@uin-suska.ac.id, ³iis.afrianty@uin-suska.ac.id,

⁴elvia.budianita@uin-suska.ac.id

Email Penulis Korespondensi: siskakurniagusti@uin-suska.ac.id

Abstrak—Diabetes Mellitus (DM) adalah penyakit jangka panjang yang berpotensi menimbulkan komplikasi serius apabila tidak terdeteksi sejak dini, sehingga diagnosis dini menjadi hal yang sangat penting. Salah satu metode yang dapat diterapkan untuk diagnosis dini adalah teknik klasifikasi pada data mining. Namun, proses klasifikasi sering terkendala oleh ketidakseimbangan kelas yang dapat menurunkan kinerja model. Penelitian ini bertujuan untuk menganalisis pengaruh teknik penyeimbangan kelas terhadap performa *Backpropagation Neural Network* (BPNN) dalam klasifikasi penyakit DM. BPNN merupakan bentuk *Multi-Layer Perceptron* (MLP) dengan struktur yang sederhana dan memiliki kemampuan menyelesaikan permasalahan kompleks dengan akurasi yang baik. Dataset yang digunakan adalah Pima Indians Diabetes Dataset dengan total 768 data, terdiri dari 500 data non-diabetes dan 268 data diabetes. Metode penelitian dilakukan dengan tiga skenario, yaitu tanpa penyeimbangan, *Synthetic Minority Over-sampling Technique* (SMOTE), dan *Random Under Sampling* (RUS). Model BPNN dirancang dengan dua variasi arsitektur (satu *hidden layer* dan dua *hidden layer*), tiga nilai *learning rate* (0,1; 0,01; 0,001), serta jumlah *neuron* yang bervariasi. Pembagian dataset dilakukan menggunakan teknik *10-Fold Cross Validation*. Hasil penelitian menunjukkan bahwa penerapan SMOTE menghasilkan kinerja terbaik, dengan rata-rata akurasi sebesar 90,89%, presisi 91,22%, *recall* 90,89%, dan *F1-score* 90,89% pada arsitektur BPNN dengan satu *hidden layer*. Selain itu, arsitektur satu *hidden layer* terbukti lebih stabil dibandingkan dua *hidden layer*, terutama ketika jumlah data berkurang akibat penerapan RUS. Dengan demikian, kombinasi SMOTE dan BPNN dengan satu *hidden layer* memberikan performa yang lebih baik dalam klasifikasi penyakit diabetes mellitus.

Kata Kunci: BPNN; Ketidakseimbangan Kelas; Klasifikasi; RUS; SMOTE

Abstract—Diabetes Mellitus (DM) is a chronic disease that can lead to serious complications if not detected early; therefore, early diagnosis is highly important. One of the methods that can be applied for early diagnosis is the classification technique in data mining. However, the classification process often faces challenges due to class imbalance, which can reduce model performance. This study aims to analyze the effect of class balancing techniques on the performance of the Backpropagation Neural Network (BPNN) in classifying DM cases. BPNN is a form of Multi-Layer Perceptron (MLP) with a simple structure and the ability to solve complex problems with good accuracy. The dataset used in this study is the Pima Indians Diabetes Dataset, consisting of 768 instances, including 500 non-diabetic and 268 diabetic cases. The research was conducted using three scenarios: without balancing, Synthetic Minority Over-sampling Technique (SMOTE), and Random Under Sampling (RUS). The BPNN model was designed with two architectural variations (one hidden layer and two hidden layers), three learning rate values (0.1, 0.01, and 0.001), and a varying number of neurons. The dataset was divided using the 10-Fold Cross Validation technique. The results show that applying SMOTE achieved the best performance, with an average accuracy of 90.89%, precision of 91.22%, recall of 90.89%, and F1-score of 90.89% on the BPNN architecture with one hidden layer. Furthermore, the single hidden layer architecture proved more stable than the two hidden layers, especially when the dataset size decreased due to RUS. Therefore, the combination of SMOTE and BPNN with one hidden layer provides better performance in classifying Diabetes Mellitus cases.

Keywords: BPNN; Class Imbalance; Classification; RUS; SMOTE

1. PENDAHULUAN

Diabetes Melitus (DM) adalah penyakit jangka panjang yang ditandai dengan peningkatan kadar glukosa darah, yang disebabkan oleh gangguan dalam produksi maupun respon tubuh terhadap insulin [1]. Berdasarkan data tahun 2019, di seluruh dunia terdapat sekitar 463 juta penderita diabetes, dan jumlah tersebut diprediksikan akan meningkat pada tahun 2045 hingga 700 juta kasus [2]. *World Health Organization* (WHO) juga memprediksi bahwa pada tahun 2030, DM diprediksi menjadi salah satu penyebab kematian utama secara global, seiring pada setiap tahunnya terdapat peningkatan jumlah penderita diabetes. Indonesia sendiri merupakan salah satu negara dengan jumlah penderita diabetes terbanyak yang menempati peringkat keempat di dunia [3].

Faktor penyebab utama dari penyakit ini meliputi pola hidup yang berubah, pola makan yang tidak seimbang, rendahnya kesadaran deteksi dini, serta rendahnya aktivitas fisik [4]. Penyakit ini ditandai dengan kondisi *hiperglikemia*, yaitu meningkatnya kadar gula darah yang disebabkan oleh gangguan produksi maupun penggunaan insulin secara efektif [5]. Apabila tidak ditangani dengan tepat, DM dapat menimbulkan komplikasi serius seperti gangguan ginjal, serangan jantung, amputasi, dan kerusakan saraf [3]. Oleh karena itu, diagnosis dini menjadi langkah yang penting sebagai upaya pencegahan untuk meminimalkan risiko komplikasi jangka panjang [6], [7], [8].

Diagnosis dini penyakit DM dapat dilakukan dengan memanfaatkan perkembangan teknologi data mining yang dapat menjadi alat yang potensial dalam membantu proses diagnosis [7]. Data mining merupakan bagian dari proses *Knowledge Discovery in Database* (KDD), yang bertujuan untuk menggali pola atau informasi bernilai dari basis data yang besar [9]. Salah satu teknik data mining yang banyak digunakan adalah klasifikasi, yaitu proses mengelompokkan data berdasarkan target tertentu [10].



Salah satu metode klasifikasi yang banyak diterapkan pada penelitian diagnosis penyakit adalah *Backpropagation Neural Network* (BPNN) yang merupakan bentuk *Multi-Layer Perceptron* (MLP). Penelitian oleh Nurhadi, Defit, dan Nurcahyo pada tahun 2025 yang menerapkan *multilayer perceptron* untuk mengidentifikasi demam *dengue* dan tifus memperoleh akurasi sebesar 98,68% pada rasio data 70:30 [11]. Selanjutnya, penelitian yang dilakukan oleh Ramadani dan kawan-kawan pada tahun 2024 dengan mengaplikasikan BPNN dalam klasifikasi penyakit jantung koroner dan mencatat rata-rata akurasi mencapai 98,42% dengan perbandingan data latih 90% dan data uji 10% [12]. Penelitian lainnya oleh Marwati dan Fauzi pada tahun 2024 menunjukkan bahwa penggunaan BPNN dalam klasifikasi penyakit diabetes melitus menghasilkan akurasi sebesar 80,75% dan presisi sebesar 81,74% [1]. Sementara itu, penelitian oleh Guswanti dan kawan-kawan pada tahun 2025 menggunakan *backpropagation* dengan *nguyen widrow* sebagai metode inialisasi bobot menghasilkan akurasi 92,11% pada learning rate 0,001 dan arsitektur BPNN [8; 9; 1] [8]. Berdasarkan hasil dari berbagai penelitian tersebut, dapat disimpulkan bahwa BPNN merupakan metode jaringan saraf tiruan (JST) yang terbukti efektif dan mampu menyelesaikan permasalahan klasifikasi yang kompleks dengan akurasi tinggi melalui proses pembelajaran berulang dan penyesuaian bobot secara bertahap [13], [14].

Pada proses klasifikasi memiliki tantangan utama yang perlu diperhatikan yaitu ketidakseimbangan kelas (*class imbalance*). Ketidakseimbangan kelas yaitu kondisi ketika distribusi jumlah data antar kelas tidak seimbang, di mana satu kelas memiliki data lebih besar dibanding kelas lain [15], [16]. Kondisi tidak seimbang ini membuat model klasifikasi lebih sering memprediksi kelas mayoritas, sehingga performa klasifikasi terhadap kelas minoritas menurun [17], [18]. Untuk mengatasi hal tersebut, digunakan teknik penyeimbang kelas seperti *oversampling* dan *undersampling*. *Synthetic Minority Oversampling Technique* (SMOTE) sebagai teknik *oversampling* yang menunjukkan kemampuan dalam meningkatkan akurasi dengan menambah sampel sintesis pada kelas minoritas tanpa menghapus data asli [19]. SMOTE juga efektif mengurangi risiko *overfitting* dan meningkatkan generalisasi model [20]. Di sisi lain, *Random Undersampling* (RUS) bekerja dengan menghapus sebagian data dari kelas mayoritas sehingga jumlah kelas seimbang [21].

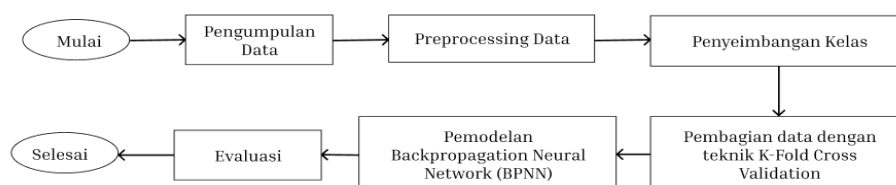
Penerapan SMOTE dan RUS telah diterapkan pada berbagai penelitian, diantaranya penelitian oleh Azhima dan kawan-kawan pada tahun 2024 membuktikan bahwa penggunaan SMOTE pada klasifikasi penyakit stroke dengan BPNN mampu meningkatkan akurasi hingga 96,14% [22]. Selanjutnya, penelitian oleh Muhidin, Danny, dan Surojudin pada tahun 2025 menunjukkan bahwa penerapan SMOTE pada prediksi kegagalan perangkat industri menggunakan *Random Forest* menghasilkan akurasi sebesar 97% serta meningkatkan nilai presisi, *recall*, dan *F1-score*, dengan peningkatan paling signifikan pada *recall* sebesar 21% [16]. Penelitian lainnya yang dilakukan oleh Ramadhan dan Salam pada tahun 2024 menunjukkan bahwa penerapan RUS pada klasifikasi kista ginjal menggunakan CNN mencapai akurasi sebesar 99% [15]. Selanjutnya penelitian oleh Untoro dan Yusuf pada tahun 2023 menunjukkan bahwa penggunaan RUS dengan *Decision Tree* pada dataset *resolve imbal anced* dapat mengatasi ketidakseimbangan data dengan nilai akurasi 76,21%, presisi 76,28%, *recall* 76,74%, dan *f-measure* 76,48% [23].

Penelitian ini merupakan lanjutan dari studi sebelumnya oleh Guswanti pada tahun 2025 yang menyarankan agar dilakukan pengujian lebih lanjut dengan mempertimbangkan teknik penyeimbang kelas, karena ketidakseimbangan kelas dapat mempengaruhi performa model [8]. Menindaklanjuti hal tersebut, penelitian ini mengadopsi dua teknik penyeimbang kelas yaitu SMOTE (*Synthetic Minority Over-sampling Technique*) sebagai teknik *oversampling* dan *Random Undersampling* (RUS) sebagai teknik *undersampling*, yang masing-masing akan diuji dalam model klasifikasi menggunakan BPNN. Selain itu, penelitian ini juga menerapkan teknik inialisasi bobot Xavier (*Glorot Initialization*) yang menentukan bobot dan bias secara acak dalam rentang tertentu berdasarkan jumlah *neuron* pada lapisan *input* dan *output*. Inialisasi Xavier terbukti efektif saat digunakan bersama fungsi aktivasi *sigmoid biner* [24], karena dapat mengurangi risiko *vanishing gradient* yang membuat proses pembelajaran menjadi lambat atau terhenti. Dengan demikian, teknik Inialisasi Xavier membantu menjaga distribusi aktivasi tetap stabil pada setiap lapisan jaringan.

Penelitian ini juga menilai performa model secara keseluruhan menggunakan beberapa metrik evaluasi yaitu presisi, *recall*, *F1-score*, dan akurasi. Dengan demikian, penelitian ini bertujuan untuk meningkatkan performa klasifikasi penyakit diabetes melitus melalui penerapan BPNN dengan teknik penyeimbang kelas SMOTE dan RUS, serta penggunaan inialisasi bobot Xavier.

2. METODOLOGI PENELITIAN

Penelitian ini bersifat kuantitatif eksperimental yang bertujuan untuk mengevaluasi pengaruh teknik penyeimbangan data terhadap performa algoritma *Backpropagation Neural Network* (BPNN) dalam mengklasifikasikan penyakit diabetes mellitus. Proses penelitian dilaksanakan melalui beberapa tahapan utama yang dilakukan secara sistematis hingga diperoleh hasil evaluasi kinerja model. Tahapan penelitian ini disajikan pada Gambar 1.



Gambar 1. Tahapan Penelitian



2.1 Pengumpulan Data

Pada penelitian ini, dataset yang digunakan adalah Pima Indians Diabetes Dataset, yang diperoleh dari platform *Kaggle* pada tautan <https://www.kaggle.com/datasets/jamaltariqcheema/pima-indians-diabetes-dataset>. Dataset ini terdiri dari 768 data, yang terbagi menjadi 268 data pasien diabetes dan 500 data pasien non-diabetes. Setiap data memiliki 8 atribut dan dua label klasifikasi, yaitu diabetes dan non-diabetes. Daftar atribut pada dataset ditampilkan pada Tabel 1.

Tabel 1. Daftar Atribut pada Dataset

No	Atribut	Deskripsi
1	<i>Pregnancies</i>	Jumlah kehamilan yang pernah dialami oleh pasien.
2	<i>Glucose</i>	Konsentrasi plasma glukosa 2 jam setelah dilakukan <i>Oral Glucose Tolerance Test</i> (mg/dl).
3	<i>Blood Pressure</i>	Tekanan darah diastolik pasien yang diukur dalam satuan mm/Hg.
4	<i>Skin Thickness</i>	Ketebalan lipatan kulit trisep yang diukur dalam milimeter (mm).
5	<i>Insulin</i>	Kadar insulin serum 2 jam setelah uji dalam satuan mikro unit per mililiter (mu U/ml).
6	<i>BMI</i>	Indeks Massa Tubuh pasien, dihitung berdasarkan berat badan (kg) dibagi tinggi badan kuadrat (m ²).
7	<i>Diabetes Pedigree Function</i>	Nilai yang merepresentasikan riwayat diabetes pada keluarga serta hubungan genetik dengan pasien. Semakin tinggi nilainya, semakin besar kemungkinan pasien menderita diabetes.
8	<i>Age</i>	Usia pasien dalam tahun.
9	<i>Outcome</i>	Variabel target, dengan nilai 0 menunjukkan pasien tidak menderita diabetes, dan 1 menunjukkan pasien menderita diabetes.

2.2 Preprocessing Data

Tahap *preprocessing* adalah langkah untuk memastikan kualitas data dan konsistensi data sebelum digunakan dalam pemodelan [25]. Pada penelitian ini, tahap *preprocessing* data meliputi:

2.2.1 Pembersihan Data

Langkah ini bertujuan untuk menangani data yang hilang (*missing value*), *noise* data, serta memastikan konsistensi dan relevansi data [14]. Dalam penelitian ini, dilakukan pengecekan terhadap duplikasi data dan data yang hilang (*missing values*). Jika ditemukan data duplikat, maka data tersebut akan dihapus. Sementara itu, *missing value* diatasi dengan pengisian menggunakan nilai rata-rata (*mean*) untuk meminimalkan potensi bias pada dataset yang berukuran kecil.

2.2.2 Transformasi Data

Transformasi data dilakukan untuk menyesuaikan format atau skala data supaya sesuai dengan kebutuhan pemodelan, sehingga dapat meningkatkan kinerja algoritma yang digunakan [26]. Pada penelitian ini, tidak dilakukan transformasi bentuk data karena seluruh atribut dalam dataset sudah berbentuk numerik dan sesuai dengan format yang dapat langsung diproses oleh model. Selanjutnya, normalisasi dilakukan pada tahap ini agar setiap fitur berada pada skala yang seragam, yaitu dalam rentang 0 sampai 1. Penelitian ini menerapkan teknik *Min-Max Normalization* [8], sebagaimana ditunjukkan pada Persamaan (1).

$$x' = \frac{x - \min(x)}{\max(x) - \min(x)} \quad (1)$$

Dimana x' merupakan hasil normalisasi, x merupakan data asli, $\max(x)$ merupakan nilai maksimum pada atribut, $\min(x)$ merupakan nilai minimum pada atribut.

2.3 Teknik Penyeimbang Kelas

Distribusi kelas pada dataset menunjukkan ketidakseimbangan antara kelas mayoritas (non-diabetes) dan kelas minoritas (diabetes). Untuk mengatasi hal tersebut, dua teknik penyeimbangan digunakan:

2.3.1 SMOTE (Synthetic Minority Oversampling Technique).

SMOTE adalah teknik *oversampling* yang menghasilkan data sintetis pada kelas minoritas dengan memanfaatkan kedekatan antar sampel [26]. Pemilihan SMOTE didasarkan pada kemampuannya menambah jumlah data minoritas tanpa mengurangi informasi kelas mayoritas, sehingga distribusi data menjadi lebih seimbang. Secara matematis, proses pembentukan data sintetis dalam SMOTE dapat dirumuskan pada Persamaan (2) [27].

$$x^b = x + u(x^k - x) \quad (2)$$

Dimana x^b adalah data sintetis, x^k adalah data tetangga terdekat, u adalah bilangan acak antara 0 dan 1, x adalah data kelas minoritas.



2.3.2 RUS (Random Undersampling)

Random undersampling (RUS) adalah teknik undersampling yang secara acak mengurangi sebagian sampel dari kelas mayoritas hingga diperoleh distribusi data yang lebih seimbang [28]. Metode ini dapat membuat model lebih fokus mengenali pola kelas minoritas tanpa mengabaikan kelas mayoritas, sekaligus mengurangi risiko overfitting dan mempercepat proses pelatihan [15].

2.4 K-Fold Cross Validation

Penelitian ini menggunakan 10-fold cross validation sebagai teknik untuk membagi data. Pada setiap iterasi, satu subset data digunakan untuk pengujian, sementara sembilan subset sisanya digunakan untuk pelatihan. Proses ini berlangsung sebanyak 10 kali hingga seluruh bagian pernah berperan sebagai data uji satu kali dan data latih sembilan kali. Salah satu keunggulan menggunakan k-fold cross validation adalah kemampuannya dalam meminimalkan risiko overfitting, karena model dilatih dan diuji menggunakan berbagai kombinasi data [29]. Alur kerja dari 10-Fold Cross Validation diilustrasikan pada Tabel 2.

Tabel 2. Alur Kerja 10-Fold Cross Validation [30]

Fold 1	1	2	3	4	5	6	7	8	9	10
Fold 2	1	2	3	4	5	6	7	8	9	10
Fold 3	1	2	3	4	5	6	7	8	9	10
Fold 4	1	2	3	4	5	6	7	8	9	10
Fold 5	1	2	3	4	5	6	7	8	9	10
Fold 6	1	2	3	4	5	6	7	8	9	10
Fold 7	1	2	3	4	5	6	7	8	9	10
Fold 8	1	2	3	4	5	6	7	8	9	10
Fold 9	1	2	3	4	5	6	7	8	9	10
Fold 10	1	2	3	4	5	6	7	8	9	10

Keterangan:

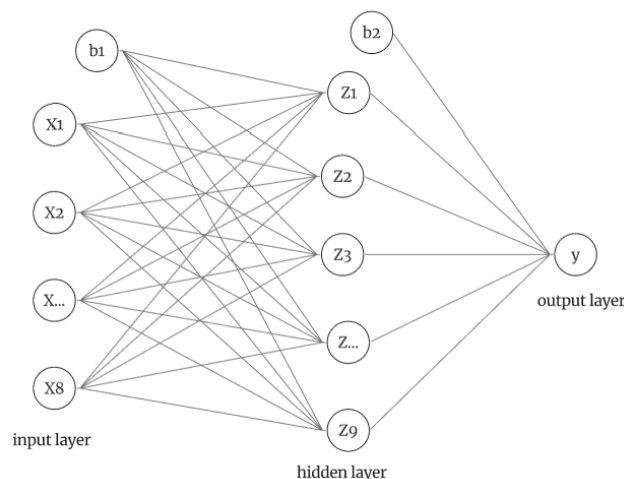
- : Data Training
- : Data Testing

2.5 Pemodelan BPNN

Backpropagation Neural Network (BPNN) merupakan metode pembelajaran terawasi yang terdiri atas tiga lapisan utama, yaitu lapisan input, lapisan hidden, dan lapisan output [14]. Setiap lapisan jaringan terdiri dari neuron-neuron yang terhubung dengan bobot (weight), dan bobot tersebut diperbarui secara berulang dalam tahap pelatihan [31]. Jumlah neuron pada hidden layer dapat ditentukan menggunakan Persamaan (3) [22].

$$i < m < 2i \tag{3}$$

Dimana m merupakan neuron hidden dan i merupakan neuron input. Salah satu arsitektur BPNN yang digunakan disajikan pada Gambar 2.



Gambar 2. Arsitektur BPNN

Gambar 2 menunjukkan salah satu arsitektur BPNN yang digunakan, terdiri atas 8 neuron pada lapisan input, 9 neuron hidden layer, dan 1 neuron pada lapisan output. Delapan neuron pada lapisan input merepresentasikan delapan fitur pada dataset, sedangkan sembilan neuron pada lapisan tersembunyi diperoleh berdasarkan Persamaan (3). Satu neuron pada lapisan output digunakan untuk menghasilkan keluaran berupa klasifikasi biner, yaitu diabetes atau non-



diabetes. Kemudian, antar lapisan dihubungkan dengan bobot dimana v antara *input-hidden* dan w antara *hidden-output* [32]. Sebelum proses pelatihan dimulai, bobot awal jaringan diinisialisasi menggunakan metode Xavier. Metode ini menginisialisasi bobot dan bias secara acak pada interval tertentu berdasarkan jumlah *neuron* pada lapisan *input* dan *output*. Rumus perhitungan inisialisasi Xavier ditunjukkan pada Persamaan (4).

$$W \sim U = \left[-\frac{\sqrt{6}}{\sqrt{n_i+n_{i+1}}}, \frac{\sqrt{6}}{\sqrt{n_i+n_{i+1}}} \right] \quad (4)$$

Dimana W adalah bobot yang akan diinisialisasikan, U adalah distribusi uniform, n_i adalah jumlah *neuron* pada *layer* sebelumnya, n_{i+1} adalah jumlah *neuron* pada *layer* berikutnya, dan $\sqrt{6}$ adalah nilai konstan dari *xavier initialization*. Proses pelatihan BPNN dilakukan melalui tiga tahap utama, yaitu propagasi maju (*feedforward*), propagasi balik (*backpropagation*), serta pembaruan bobot dan bias [8]. Parameter pelatihan yang digunakan dalam penelitian ini meliputi fungsi aktivasi *sigmoid*, jumlah *epoch* maksimum sebanyak 1.000, target *error* sebesar 0,01, serta variasi *learning rate* (0,1; 0,01; 0,001) serta optimasi menggunakan Adam. Adam melakukan pembaruan bobot secara adaptif dengan menyesuaikan *learning rate* selama proses pelatihan [33]. Optimizer Adam dipilih karena memiliki *adaptive learning rate* dengan menggabungkan keunggulan Momentum dan RMSProp, sehingga mampu mempercepat konvergensi, mengurangi tingkat kesalahan prediksi, dan menjaga stabilitas pelatihan [34].

2.6 Evaluasi

Evaluasi dilakukan untuk mengukur kemampuan model BPNN dalam mengklasifikasi data secara tepat. Pengukuran kinerja dilakukan menggunakan *Confusion Matrix*, yang berisi nilai *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN), yang merepresentasikan jumlah hasil prediksi yang tepat dan yang salah pada data uji [14]. Dari *confusion matrix*, terdapat beberapa metrik evaluasi yang digunakan sebagai berikut:

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN} \quad (5)$$

$$\text{Presisi} = \frac{TP}{TP+FP} \quad (6)$$

$$\text{Recall} = \frac{TP}{TP+FN} \quad (7)$$

$$\text{F1 score} = \frac{2 \times (\text{presisi} \times \text{recall})}{\text{presisi} + \text{recall}} \quad (8)$$

3. HASIL DAN PEMBAHASAN

3.1 Deskripsi Dataset

Penelitian ini menggunakan *Pima Indians Diabetes Dataset* yang terdiri dari 768 sampel dengan delapan atribut, dan label target dibagi menjadi dua kelas, yaitu kelas 0 yang menunjukkan pasien tidak menderita diabetes, dan kelas 1 yang menunjukkan pasien menderita diabetes. Kelas 0 berjumlah 500 sampel, sedangkan kelas 1 berjumlah 268 sampel. Distribusi ini menunjukkan adanya ketidakseimbangan kelas, di mana kelas mayoritas lebih dominan dibandingkan kelas minoritas. Tabel 3 menyajikan parameter pengukuran diabetes pada masing-masing atribut yang digunakan dalam penelitian ini. Nilai Outcome merupakan label target klasifikasi dengan nilai 0 untuk pasien tidak menderita diabetes dan 1 untuk pasien menderita diabetes.

Tabel 3. Parameter Pengukuran Dataset

No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age	Outcome
1	6	148	72	35	169,5	33,6	0,627	50	1
2	1	85	66	29	102,5	26,6	0,351	31	0
3	8	183	64	32	169,5	23,2	0,672	32	1
....
768	1	93	70	31	102,5	30,4	0,315	23	0

3.2 Hasil Preprocessing Data

Preprocessing data pada penelitian ini memuat tahapan pembersihan data dan transformasi data. Hasil dari tahap pembersihan data menunjukkan bahwa tidak ditemukan data duplikat maupun nilai kosong (*missing values*) dalam dataset. Dengan demikian, tidak ada data yang perlu dihapus dan seluruh data dapat digunakan secara utuh. Selanjutnya, karena semua atribut dalam dataset sudah berbentuk numerik, tidak dilakukan transformasi tipe data. Namun, untuk memastikan bahwa setiap fitur berada pada skala yang sebanding, dilakukan proses normalisasi menggunakan metode *Min-Max Normalization* pada Persamaan (1). Hasil normalisasi pada Tabel 4 menunjukkan bahwa seluruh nilai atribut telah berada dalam rentang 0 hingga 1.



Tabel 4. Hasil Normalisasi Data

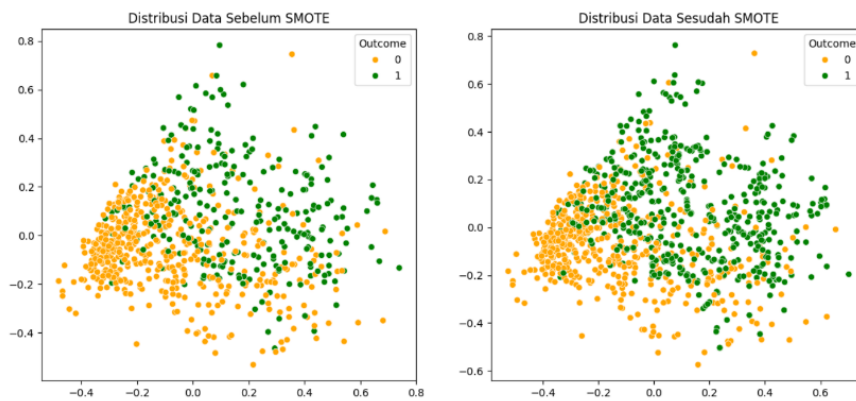
No	Pregnancies	Glucose	Blood Pressure	Skin Thickness	Insulin	BMI	Diabetes Pedigree Function	Age
1	0,35294	0,67096	0,48979	0,30434	0,18689	0,31492	0,23441	0,48333
2	0,05882	0,26451	0,42857	0,23913	0,10637	0,17177	0,11656	0,16666
3	0,47058	0,89677	0,40816	0,27173	0,18689	0,10429	0,25362	0,18333
...
768	0,05882	0,31612	0,46938	0,260862	0,10637	0,24948	0,10119	0,033333

3.3 Hasil Penyeimbang Kelas

Setelah tahap *preprocessing* selesai, dilakukan penyeimbangan distribusi kelas pada dataset. Hasil penyeimbangan sebagai berikut:

3.3.1 SMOTE (Synthetic Minority Over-sampling Technique)

Pada penelitian ini, penerapan SMOTE berhasil menambah jumlah sampel pada kelas 1 hingga seimbang dengan kelas 0, yaitu masing-masing sebanyak 500 sampel. Dengan demikian, total data setelah proses SMOTE menjadi 1.000 sampel. Visualisasi distribusi data sebelum dan sesudah penerapan SMOTE diperlihatkan pada Gambar 3 dengan menggunakan *scatter plot*.



Gambar 3. Distribusi Data Sebelum dan Sesudah SMOTE

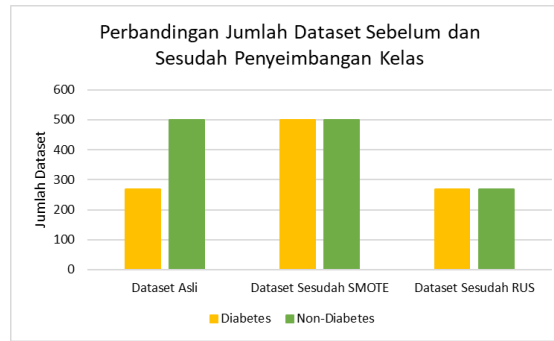
3.3.2 Random Undersampling

Pada penelitian ini, jumlah data pada kelas 0 dikurangi hingga seimbang dengan jumlah kelas 1, yaitu 268 sampel. Akibatnya, data kelas mayoritas yang tidak terpilih akan diabaikan dalam proses pelatihan. Total data setelah RUS menjadi 536 sampel. Visualisasi distribusi data sebelum dan sesudah penerapan Random Undersampling diperlihatkan pada Gambar 4 dengan menggunakan *scatter plot*.



Gambar 4. Distribusi Data Sebelum dan Sesudah Random Undersampling

Perbandingan jumlah dataset sebelum dan sesudah penyeimbangan kelas ditunjukkan pada Gambar 5. Hasil tersebut memperlihatkan bahwa baik SMOTE maupun RUS mampu mengatasi ketidakseimbangan kelas, namun dengan pendekatan yang berbeda. Teknik SMOTE menambah sampel sintetis pada kelas minoritas sehingga jumlah data menjadi seimbang, sedangkan RUS mengurangi jumlah data pada kelas mayoritas agar sesuai dengan jumlah data pada kelas minoritas.



Gambar 5. Perbandingan Jumlah Dataset Sebelum dan Sesudah Penyeimbangan Kelas

3.4 Hasil Pembagian Data Latih dan Data Uji

Pembagian dataset ke dalam data latih dan data uji dilakukan melalui metode *10-Fold Cross Validation*. Skema ini membagi data menjadi sepuluh subset berukuran sama, lalu secara berulang satu subset dipakai sebagai data uji, sementara sembilan lainnya dipakai untuk pelatihan. Proses ini diulang sebanyak sepuluh kali agar setiap subset pernah digunakan sebagai data uji satu kali. Penerapan *10-Fold Cross Validation* dilakukan setelah proses penyeimbangan kelas menggunakan SMOTE dan RUS. Dengan demikian, metode ini diterapkan pada tiga skenario, yaitu tanpa penyeimbangan, dengan SMOTE, dan dengan RUS. Seluruh subset data pada proses validasi berasal dari dataset yang telah disesuaikan pada masing-masing skenario, sehingga pembagian data latih dan data uji dilakukan secara konsisten berdasarkan hasil penyeimbangan awal.

3.5 Hasil Pemodelan BPNN

Model dikembangkan menggunakan *Backpropagation Neural Network* (BPNN) dengan dua konfigurasi arsitektur, yaitu satu *hidden layer* dan dua *hidden layer*. Jumlah *neuron* pada masing-masing layer bervariasi dimana *neuron* pada *input layer* berasal dari atribut. Jumlah *neuron* pada *hidden layer* dihitung menggunakan Persamaan (3) dengan tiga skenario, yaitu nilai terdekat, nilai tengah, dan nilai tertinggi dari perhitungan. Pada *output layer*, jumlah *neuron* ditentukan sesuai dengan jumlah kelas pada data target. Parameter jaringan yang digunakan adalah fungsi aktivasi *sigmoid*, jumlah maksimum iterasi (*epoch*) sebesar 1.000, target *error* 0,01, serta variasi *learning rate* (0,1; 0,01; dan 0,001) dengan optimizer Adam. Bobot dan bias diinisialisasi menggunakan metode Xavier (*Xavier Initialization*) dengan Persamaan (4).

3.6 Evaluasi Performa Model

Pengujian model dilakukan dengan mengevaluasi setiap konfigurasi BPNN pada berbagai kombinasi *learning rate* (0,1; 0,01; 0,001) dan jumlah *neuron* yang berbeda. Selain itu, pengujian dilakukan pada tiga skenario data yaitu tanpa penyeimbang, dengan SMOTE (*Synthetic Minority Over-sampling Technique*), dan dengan RUS (*Random Under Sampling*).

3.6.1 BPNN dengan Satu Hidden Layer

Pada arsitektur BPNN dengan satu *hidden layer*, performa terbaik tanpa penyeimbangan kelas diperoleh pada konfigurasi dengan *learning rate* 0,001 dan 12 *neuron* pada *hidden layer* dengan akurasi sebesar 93,42%, presisi 92%, *recall* 94%, dan *F1-score* 93%. Penerapan SMOTE mendapatkan konfigurasi terbaik pada *learning rate* 0,1 dengan 9 *neuron hidden*, menghasilkan akurasi 95%, presisi 95%, *recall* 95%, dan *F1-score* 95%. Sementara itu, penerapan RUS menunjukkan performa terbaik pada konfigurasi dengan *learning rate* 0,1 dan 9 *neuron hidden*, serta konfigurasi dengan *learning rate* 0,1 dan 12 *neuron hidden* yang memberikan nilai sama, yaitu akurasi 92,45%, presisi 93%, *recall* 92%, dan *F1-score* 92%. Hasil akurasi untuk setiap skenario pada arsitektur BPNN dengan satu *hidden layer* ditampilkan pada Tabel 5.

Tabel 5. Hasil Akurasi BPNN dengan Satu Hidden Layer

Learning rate	Jumlah neuron hidden	Tanpa penyeimbangan (%)	Dengan SMOTE (%)	Dengan RUS (%)
0,1	9	85,53	95	92,45
	12	86,84	88	92,45
	15	89,47	90	84,91
0,01	9	88,16	91	84,91
	12	89,47	87	90,57
	15	88,16	94	81,13
0,001	9	88,16	92	88,68
	12	93,42	90	92,45
	15	92,11	91	86,79
Rata-Rata		89,04	90,89	88,26



3.6.2 BPNN dengan Dua Hidden layer

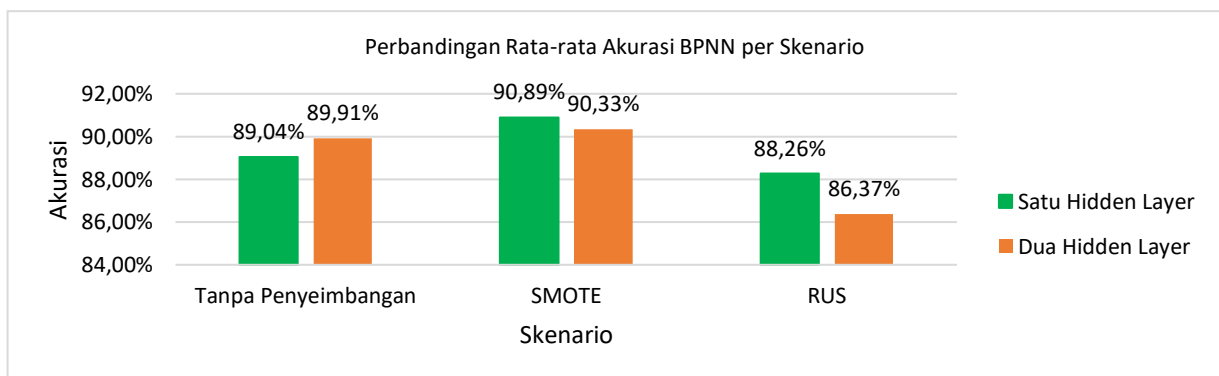
Pada arsitektur BPNN dengan dua *hidden layer*, performa tertinggi tanpa penyeimbang diperoleh pada konfigurasi *learning rate* 0,1 dan jumlah *neuron hidden* 12–15 dengan akurasi 93,42%, presisi 93%, *recall* 93%, dan *F1-score* 93%. SMOTE menghasilkan performa terbaik dengan akurasi 93%, presisi 93%, *recall* 93%, dan *F1-score* 93% pada konfigurasi *learning rate* 0,1 dan *neuron hidden* 15–9. Sementara itu, RUS menghasilkan performa tertinggi dengan akurasi 94,34%, presisi 94%, *recall* 94%, dan *F1-score* 94% pada konfigurasi *learning rate* 0,1 dan *neuron hidden* 12–15. Hasil akurasi untuk setiap skenario pada arsitektur BPNN dengan dua *hidden layer* ditampilkan pada Tabel 6.

Tabel 6. Hasil Akurasi BPNN dengan Dua Hidden Layer

Learning rate	Jumlah neuron hidden	Tanpa penyeimbangan (%)	Dengan SMOTE (%)	Dengan RUS (%)
0,1	9-12	90,79	88	92,45
	12-15	93,42	92	94,34
	15-9	88,16	93	88,68
0,01	9-12	90,79	90	81,13
	12-15	84,21	89	81,13
	15-9	88,16	88	81,13
0,001	9-12	90,79	92	86,79
	12-15	92,11	90	84,91
	15-9	90,79	91	86,79
Rata-Rata		89,91	90,33	86,37

Tabel 5 dan Tabel 6 memperlihatkan bahwa kinerja *Backpropagation Neural Network* (BPNN) bervariasi tergantung pada konfigurasi jumlah *hidden layer* dan teknik penyeimbangan kelas yang digunakan. Pada skenario tanpa penyeimbangan, arsitektur BPNN dengan dua *hidden layer* memperoleh rata-rata akurasi sebesar 89,91%, sedikit lebih tinggi dibandingkan arsitektur satu *hidden layer* yang mencapai 89,04%. Perbedaan yang relatif kecil ini menunjukkan bahwa penambahan *hidden layer* tidak memberikan peningkatan performa yang signifikan, karena kompleksitas data yang rendah sudah dapat ditangani dengan satu *hidden layer*.

Pada skenario dengan SMOTE, kinerja kedua arsitektur relatif setara, yakni 90,89% untuk satu *hidden layer* dan 90,33% untuk dua *hidden layer*. Hal ini mengindikasikan bahwa teknik *oversampling* melalui SMOTE mampu menyeimbangkan distribusi kelas sehingga proses pembelajaran model menjadi lebih optimal. Sebaliknya, penerapan RUS justru menurunkan akurasi, terutama pada arsitektur dua *hidden layer* yang hanya mencapai 86,37%, dibandingkan satu *hidden layer* dengan 88,26%. Penurunan ini disebabkan oleh berkurangnya jumlah data akibat *undersampling*, sehingga sebagian informasi penting hilang dan kemampuan generalisasi model melemah. Secara keseluruhan, perbandingan rata-rata akurasi pada setiap skenario ditunjukkan pada Gambar 6.



Gambar 6. Perbandingan Rata-Rata Akurasi BPNN per Skenario

Berdasarkan Gambar 6, teknik SMOTE memberikan akurasi tertinggi pada semua konfigurasi arsitektur yang diuji, menunjukkan bahwa metode *oversampling* lebih efektif dibandingkan tanpa penyeimbangan atau dengan *undersampling*. Keunggulan SMOTE terletak pada kemampuannya menambah data minoritas tanpa mengurangi data mayoritas, sehingga informasi tetap utuh. Sebaliknya, RUS justru mengurangi data mayoritas sehingga sebagian informasi hilang. Selain itu, arsitektur dengan satu *hidden layer* terbukti lebih stabil pada seluruh skenario, khususnya ketika jumlah data berkurang akibat RUS. Stabilitas ini dipengaruhi oleh struktur model yang lebih sederhana dan jumlah bobot yang lebih sedikit, sehingga meskipun data berkurang, model tetap dapat belajar dengan baik dan menghasilkan kinerja yang konsisten. Sementara itu, arsitektur dengan dua *hidden layer* cenderung lebih sensitif terhadap perubahan jumlah data dan berpotensi mengalami *overfitting* karena memiliki lebih banyak parameter, terutama jika data sintetis yang dihasilkan terlalu mirip dengan sampel asli.

Tahap berikutnya dalam evaluasi kinerja model diterapkan *confusion matrix* dengan menghitung nilai presisi, *recall*, dan *F1-score* pada Persamaan (6) hingga Persamaan (8). Metrik ini dipilih karena lebih representatif dalam menilai

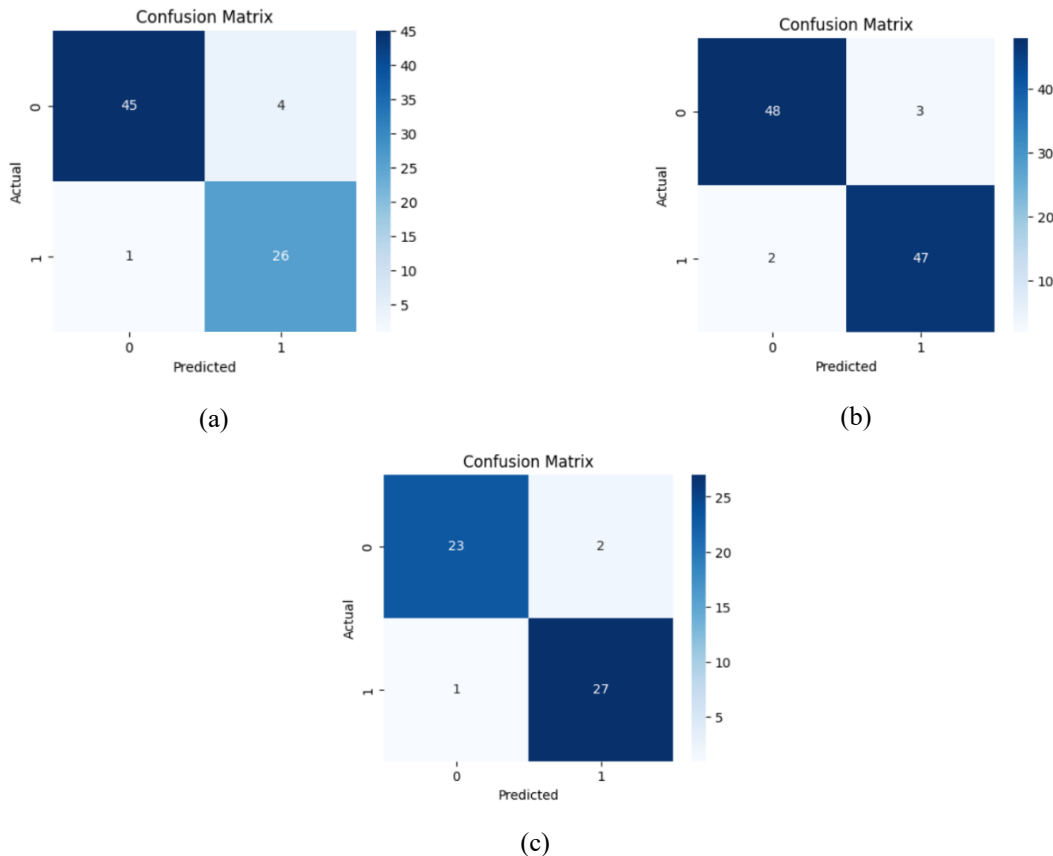


performa klasifikasi pada data medis yang tidak seimbang dibandingkan hanya menggunakan akurasi. Rata-rata hasil evaluasi ditampilkan pada Tabel 7.

Tabel 7. Rata-Rata Hasil Evaluasi Kinerja Model BPNN

Jumlah Hidden Layer	Skenario	Presisi (%)	Recall (%)	F1-Score (%)
1	Tanpa Penyeimbang	88,22	88,22	88
	SMOTE	91,22	90,89	90,89
	RUS	88,44	88,11	88,22
2	Tanpa Penyeimbang	88,78	89,78	89,22
	SMOTE	90,67	90,33	90,33
	RUS	86,67	86,33	86,33

Tabel 7 menunjukkan bahwa penerapan SMOTE konsisten menghasilkan nilai presisi, recall, dan F1-score yang lebih tinggi dibandingkan tanpa penyeimbang maupun RUS. Pada konfigurasi satu hidden layer, SMOTE mencapai presisi 91,22%, recall 90,89%, dan F1-score 90,89%. Temuan ini menunjukkan bahwa penggunaan SMOTE dapat meningkatkan kemampuan model dalam mengenali kelas minoritas dengan baik dan tetap mempertahankan nilai presisi. Sebaliknya, nilai terendah diperoleh pada skenario RUS dengan dua hidden layer, yaitu presisi 86,67%, recall 86,33%, dan F1-score 86,33%. Kondisi ini terjadi karena RUS menghapus sebagian data kelas mayoritas, sehingga informasi penting hilang dan kinerja model menurun. Adapun hasil confusion matrix untuk masing-masing skenario ditampilkan pada Gambar 7.



Gambar 7. Hasil Confusion Matrix dengan (a) Tanpa Penyeimbang, (b) SMOTE, dan (c) RUS

Gambar 7 menunjukkan hasil Confusion Matrix untuk setiap skenario, yang diambil dari konfigurasi model dengan akurasi tertinggi secara keseluruhan. Konfigurasi terbaik tersebut meliputi model tanpa penyeimbang dengan learning rate 0,001 dan 12 neuron hidden layer, model dengan SMOTE pada learning rate 0,1 dan 9 neuron hidden, serta model dengan RUS pada learning rate 0,1 dan 12-15 neuron hidden. Pada skenario tanpa penyeimbang, terlihat bahwa model cenderung bias terhadap kelas 0 (non-diabetes), dengan nilai True Negative (TN) yang tinggi dan True Positive (TP) yang rendah. Setelah penerapan SMOTE, distribusi prediksi menjadi lebih seimbang dengan peningkatan jumlah TP, yang menandakan kemampuan model yang lebih baik dalam mengenali kelas positif (diabetes). Sementara itu, pada skenario RUS, meskipun keseimbangan kelas berhasil dicapai, jumlah prediksi benar pada kelas mayoritas mengalami penurunan akibat berkurangnya data. Dengan demikian, hasil ini membuktikan bahwa penerapan teknik penyeimbangan data dapat membantu menyeimbangkan distribusi kelas serta meningkatkan kemampuan model dalam mendeteksi kasus positif secara lebih akurat.



Temuan ini memperkuat hasil penelitian sebelumnya oleh Muhidin, Danny, dan Surojudin pada tahun 2025 yang menyatakan bahwa penerapan SMOTE mampu mengatasi ketidakseimbangan kelas serta meningkatkan nilai akurasi, presisi, *recall*, dan *F1-score* [16]. Selain itu, jika dibandingkan dengan penelitian sebelumnya oleh Guswanti dan kawan-kawan pada tahun 2025 yang menggunakan inisialisasi bobot *Nguyen-Widrow* pada BPNN dengan akurasi 92,11%, penelitian ini memperlihatkan peningkatan kinerja [8]. Dengan penerapan inisialisasi Xavier dan penyeimbangan data menggunakan SMOTE, arsitektur BPNN dengan satu *hidden layer* mampu mencapai akurasi 95%, atau meningkat 2,89% dibandingkan penelitian terdahulu. Temuan ini menegaskan bahwa teknik penyeimbangan kelas memiliki peran penting dalam meningkatkan performa model.

Secara keseluruhan, penelitian ini menunjukkan bahwa kombinasi SMOTE dan BPNN dengan satu *hidden layer* memberikan performa yang baik dalam klasifikasi penyakit diabetes mellitus. Selain itu, pendekatan ini memperlihatkan kemampuan model dalam menangani data medis yang tidak seimbang secara lebih efektif. Hasil tersebut tidak hanya berkontribusi pada pengembangan metode klasifikasi berbasis jaringan saraf tiruan, tetapi juga memiliki potensi penerapan secara praktis. Model yang dihasilkan dapat dimanfaatkan sebagai alat bantu dalam proses deteksi awal berbasis komputer, sehingga dapat membantu mempercepat pengambilan keputusan, mengurangi beban kerja manual, serta meningkatkan akurasi.

4. KESIMPULAN

Temuan penelitian ini membuktikan bahwa teknik penyeimbangan kelas memiliki pengaruh signifikan terhadap performa model *Backpropagation Neural Network* (BPNN) dalam klasifikasi penyakit diabetes mellitus. Model BPNN diuji menggunakan dua konfigurasi arsitektur, yaitu satu *hidden layer* dan dua *hidden layer*, serta tiga nilai *learning rate* (0,1; 0,01; dan 0,001) dengan jumlah *neuron* yang bervariasi. Pengujian dilakukan pada tiga skenario data, yaitu tanpa penyeimbangan, dengan SMOTE (*Synthetic Minority Over-sampling Technique*), dan dengan RUS (*Random Under Sampling*). Dari ketiga skenario tersebut, SMOTE menghasilkan performa terbaik pada semua konfigurasi, dengan rata-rata akurasi terbaik sebesar 90,89% pada BPNN dengan satu *hidden layer*. Penerapan SMOTE juga menghasilkan nilai presisi, *recall*, dan *F1-score* yang lebih tinggi dibandingkan dua skenario lainnya. Hasil terbaik didapatkan pada konfigurasi satu *hidden layer*, dimana SMOTE mencapai presisi 91,22%, *recall* 90,89%, dan *F1-score* 90,89%. Nilai *recall* sebesar 90,89% menunjukkan kemampuan model dalam mengenali kasus positif diabetes dengan baik. Hal ini penting dalam konteks medis karena *recall* mencerminkan kemampuan sistem untuk mendeteksi pasien yang benar-benar menderita diabetes, sehingga dapat meminimalkan kesalahan diagnosis. Kinerja unggul SMOTE disebabkan oleh kemampuannya menambah data minoritas tanpa mengurangi data mayoritas, sehingga informasi tetap utuh. Sebaliknya, RUS pada dua *hidden layer* justru memperoleh nilai terendah dengan presisi 86,67%, *recall* 86,33%, dan *F1-score* 86,33%. Hal ini dikarenakan RUS mengurangi data mayoritas sehingga sebagian informasi hilang dan kinerja model menurun. Selain itu, arsitektur satu *hidden layer* terbukti lebih stabil dibandingkan dua *hidden layer*, khususnya pada jumlah data yang berkurang akibat RUS. Hal ini dikarenakan arsitektur satu *hidden layer* memiliki struktur yang lebih sederhana dengan jumlah bobot lebih sedikit, sehingga model tetap mampu belajar secara optimal meskipun data berkurang. Secara keseluruhan, kombinasi SMOTE dan BPNN satu *hidden layer* merupakan konfigurasi terbaik dan berpotensi digunakan sebagai alat bantu dalam proses awal untuk membantu mendeteksi kemungkinan penyakit diabetes secara lebih cepat dan akurat. Namun, pada penelitian ini hanya menerapkan metode SMOTE dan RUS sebagai teknik penyeimbang kelas. Sehingga, untuk penelitian selanjutnya dapat mengeksplorasi teknik *hybrid sampling* seperti SMOTE–*Tomek Links* atau SMOTE-ENN. Metode ini menggabungkan kelebihan *oversampling* dan *undersampling*, sehingga data menjadi lebih seimbang dan model dapat mengenali pola pada kelas minoritas dengan lebih baik.

REFERENCE

- [1] F. Marwati and R. Fauzi, "Prediksi Penyakit Diabetes Melitus Menggunakan Jaringan Syaraf Tiruan Dengan Metode Backpropagation," *Jitu J. Inform. Utama Hal*, vol. 2, no. 1, pp. 26–34, 2024.
- [2] Z. Mutaqin, C. Rozikin, and Y. A. Tomo, "Klasifikasi Penyakit Diabetes Menggunakan Algoritma Logistic Regression," *J. Sist. dan Teknol. Inf.*, vol. 06, no. 3, pp. 320–329, 2024, [Online]. Available: <https://journalpedia.com/1/index.php/jsti>
- [3] S. Delfina, I. Carolita, S. Habsah, and S. Ayatillah, "Analisis Determinan Faktor Risiko Kejadian Diabetes Mellitus Tipe 2 Pada Usia Produktif," *J. Kesehat. Tambusai*, vol. 2, no. 4, pp. 141–151, 2021, doi: 10.31004/jkt.v2i4.2823.
- [4] M. K. Murtiningsih, K. Pandelaki, and B. P. Sedli, "Gaya Hidup sebagai Faktor Risiko Diabetes Mellitus Tipe 2," *e-CliniC*, vol. 9, no. 2, p. 328, 2021, doi: 10.35790/ecl.v9i2.32852.
- [5] D. N. S. Purqoti, Z. Arifin, D. Istiana, Ilham, B. R. Fatmawati, and H. P. Rusiana, "Sosialisasi konsep penyakit Diabetes Mellitus untuk meningkatkan pengetahuan Lansia tentang Diabetes Mellitus," *ABSYARA J. Pengabd. Pada Masy.*, vol. 3, no. 1, pp. 71–78, 2022, doi: 10.29408/ab.v3i1.5771.
- [6] R. P. Fadhillah, R. Rahma, A. Sepharni, R. Mufidah, B. N. Sari, and A. Pangestu, "Klasifikasi Penyakit Diabetes Mellitus Berdasarkan Faktor-Faktor Penyebab Diabetes menggunakan Algoritma C4.5," *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 7, no. 4, pp. 1265–1270, 2022, doi: 10.29100/jupi.v7i4.3248.
- [7] A. Prastyo, S. Sutikno, and K. Khadijah, "Improving support vector machine and backpropagation performance for diabetes mellitus classification," *Comput. Sci. Inf. Technol.*, vol. 5, no. 2, pp. 140–149, 2024, doi: 10.11591/csit.v5i2.pp140-149.
- [8] W. Guswanti, I. Afrianty, E. Budianita, and F. Syafria, "Perbandingan Inisialisasi Bobot Random dan Nguyen-Widrow Pada Backpropagation Dalam Klasifikasi Penyakit Diabetes," *J. Inform. J. Pengemb. IT*, vol. 10, no. 2, pp. 323–332, 2025, doi: 10.30591/jpit.v9ix.xxx.



- [9] P. M. S. Tarigan, J. T. Hardinata, H. Qurniawan, M. Saffi, and R. Winanjaya, "Implementasi Data Mining Menggunakan Algoritma Apriori Dalam Menentukan Persediaan Barang (Studi Kasus : Toko Sinar Harahap)," *Just IT J. Sist. Informasi, Teknol. Inf. dan Komput.*, vol. 12, no. 2, pp. 51–61, 2022, doi: 10.35316/justify.v3i1.5335.
- [10] P. Rosyani, S. Saprudin, and R. Amalia, "Klasifikasi Citra Menggunakan Metode Random Forest dan Sequential Minimal Optimization (SMO)," *J. Sist. dan Teknol. Inf.*, vol. 9, no. 2, p. 132, 2021, doi: 10.26418/justin.v9i2.44120.
- [11] N. Nurhadi, S. Defit, and G. W. Nurcahyo, "Model Deep Learning Berbasis Multilayer Perceptron untuk Identifikasi Demam Berdarah Dengue dan Tifus," *Bull. Comput. Sci. Res.*, vol. 5, no. 5, pp. 1095–1102, 2025, doi: 10.47065/bulletincsr.v5i5.754.
- [12] R. Ramadani, E. Budianita, F. Yanto, and S. K. Gusti, "Klasifikasi Penyakit Jantung Koroner Menggunakan Metode Backpropagation Neural Network," *Semin. Nas. Teknol. Informasi, Komun. dan Ind.*, pp. 192–200, 2024.
- [13] F. K. Wardhana et al., "Penerapan backpropagation jaringan saraf tiruan untuk prediksi diabetes menggunakan dataset pima indians," *Semin. Nas. AMIKOM Surakarta*, no. November, pp. 331–344, 2024.
- [14] L. A. Ma'rifah, I. Afrianty, E. Budianita, and F. Syafria, "Klasifikasi Tulang Tengkorak Berdasarkan Jenis Kelamin Menggunakan Correlation-Based Feature Selection (CFS) dengan Backpropagation Neural Network (BPNN)," *J. Inform. J. Pengemb. IT*, vol. 10, no. 2, pp. 333–347, 2025, doi: 10.30591/jpit.v9ix.xxx.
- [15] I. S. Ramadhan and A. Salam, "Teknik Random Undersampling untuk Mengatasi Ketidakseimbangan Kelas pada CT Scan Kista Ginjal," *Techno.Com*, vol. 23, no. 1, pp. 20–28, 2024, doi: 10.62411/tc.v23i1.9738.
- [16] A. Muhidin, M. Danny, and N. Surojudin, "Prediksi Kegagalan Perangkat Industri Menggunakan Random Forest dan SMOTE untuk Pemeliharaan Preventif," *Bull. Comput. Sci. Res.*, vol. 5, no. 5, pp. 1089–1094, 2025, doi: 10.47065/bulletincsr.v5i5.745.
- [17] M. P. Pulungan, A. Purnomo, and A. Kurniasih, "Penerapan SMOTE untuk Mengatasi Imbalance Class dalam Klasifikasi Kepribadian MBTI Menggunakan Naive Bayes Classifier," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 10, no. 7, pp. 1493–1502, 2023, doi: 10.25126/jtiik.1077989.
- [18] L. Pasiolo, I. Afrianty, E. Budianita, and R. Abdillah, "Penerapan Teknik Smote Pada Klasifikasi Penyakit Stroke Dengan Algoritma Support Vector Machine," *J. Sist. Inf.*, vol. 7, no. 1, pp. 61–73, 2025, [Online]. Available: <https://www.kaggle.com/fedesoriano/stroke-prediction-dataset>.
- [19] L. Qadrini, H. Hikmah, and M. Megasari, "Oversampling, Undersampling, Smote SVM dan Random Forest pada Klasifikasi Penerima Bidikmisi Sejava Timur Tahun 2017," *J. Comput. Syst. Informatics*, vol. 3, no. 4, pp. 386–391, 2022, doi: 10.47065/josyc.v3i4.2154.
- [20] M. Sulistiyono, Y. Pristyanto, S. Adi, and G. Gumelar, "Implementasi Algoritma Synthetic Minority Over-Sampling Technique untuk Menangani Ketidakseimbangan Kelas pada Dataset Klasifikasi," *Sistemasi*, vol. 10, no. 2, p. 445, 2021, doi: 10.32520/stmsi.v10i2.1303.
- [21] E. Saputro and D. Rosiyadi, "Penerapan Metode Random Over-Under Sampling Pada Algoritma Klasifikasi Penentuan Penyakit Diabetes," *Bianglala Inform.*, vol. 10, no. 1, pp. 42–47, 2022, doi: 10.31294/bi.v10i1.11739.
- [22] M. Azhima, I. Afrianty, E. Budianita, and S. Kurnia Gusti, "Penerapan Metode Backpropagation Neural Network untuk Klasifikasi Penyakit Stroke," *KLIK Kaji. Ilm. Inform. dan Komput.*, vol. 4, no. 6, pp. 3013–3021, 2024, doi: 10.30865/klik.v4i6.1956.
- [23] M. C. Untoro and M. A. N. M. Yusuf, "Evaluate of Random Undersampling Method and Majority Weighted Minority Oversampling Technique in Resolve Imbalanced Dataset," *IT J. Res. Dev.*, vol. 8, no. 1, pp. 1–13, 2023, doi: 10.25299/itjrd.2023.12412.
- [24] M. Karmila and I. Nirmala, "Prediksi Jumlah Produksi Kebutuhan Air Pada Perumda Air Minum Tirta Khatulistiwa Pontianak Menggunakan Metode Extreme Learning Machine (Elm)," *Coding J. Komput. dan Apl.*, vol. 11, no. 1, p. 137, 2023, doi: 10.26418/coding.v11i1.58052.
- [25] F. Alghifari and D. Juardi, "Penerapan Data Mining Pada Penjualan Makanan Dan Minuman Menggunakan Metode Algoritma Naive Bayes," 2021.
- [26] A. J. P. Sibarani, "Implementasi Data Mining Menggunakan Algoritma Apriori Untuk Meningkatkan Pola Penjualan Obat," *JATISI (Jurnal Tek. Inform. dan Sist. Informasi)*, vol. 7, no. 2, pp. 262–276, 2020, doi: 10.35957/jatisi.v7i2.195.
- [27] R. S. Pradana and R. Nooraeni, "Penerapan SMOTE pada Data Tidak Seimbang dalam Pemodelan Status NEET Penduduk Usia Muda di Provinsi Banten Tahun 2022," *J. Kebijak. Pembang.*, vol. 18, no. 1, pp. 91–104, 2023.
- [28] I. Kurniawan, D. C. P. Buani, A. Abdussomad, W. Apriliah, and E. Fitriani, "Penerapan Teknik Random Undersampling untuk Mengatasi Imbalance Class dalam Prediksi Kebakaran Hutan Menggunakan Algoritma Decision Tree," *Acad. J. Comput. Sci. Res.*, vol. 5, no. 1, p. 1, 2023, doi: 10.38101/ajcsr.v5i1.617.
- [29] W. Wijiyanto, A. I. Pradana, S. Sopingi, and V. Atina, "Teknik K-Fold Cross Validation untuk Mengevaluasi Kinerja Mahasiswa," *J. Algoritma*, vol. 21, no. 1, pp. 239–248, 2024, doi: 10.33364/algoritma/v.21-1.1618.
- [30] H. Hafid, "Penerapan K-Fold Cross Validation untuk Menganalisis Kinerja Algoritma K-Nearest Neighbor pada Data Kasus Covid-19 di Indonesia," *J. Math.*, vol. 6, no. 2, pp. 161–168, 2023, [Online]. Available: <http://www.ojs.unm.ac.id/jmathcos>
- [31] I. Saluza, L. Widya Astuti, and E. Yulianti, "Ensemble Backpropagation Neural Network Dalam Memprediksi Inflasi," *J. JUPITER*, vol. 15, no. 1, pp. 732–741, 2023.
- [32] B. F. K. Lestari and L. A. Kusnaraharja, "Peran Ilmu Forensik Dalam Memecahkan Kasus Kriminalitas: Studi Di Rumah Sakit Bhayangkara Mataram," *UnizarLawReview*, vol. 4, no. 1, pp. 117–6, 2021.
- [33] F. Ramadhan and J. Hernadi, "Evaluasi Optimizer Adam dan RMSProp pada Arsitektur VGG-19 Klasifikasi Ekpresi Wajah Manusia," *JUPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform.)*, vol. 10, no. 2, pp. 1414–1426, 2025, [Online]. Available: <https://doi.org/10.29100/jupi.v10i2.6197>
- [34] P. Prihandoko and P. Alkhairi, "Optimasi JST Backpropagation dengan Adaptive Learning Rate Dalam Memprediksi Hasil Panen Padi," *Jurasik (Jurnal Ris. Sist. Inf. dan Tek. Inform.)*, vol. 10, no. 1, p. 441, 2025, doi: 10.30645/jurasik.v10i1.887.