Laporan Tugas Besar Swarm Intelligence

Penerapan Efisiensi Particle Swarm Optimization Dengan Naive Bayes Dalam Klasifikasi Penyakit Diabetes

Disusun Oleh :

Kelompok 11 (RA)

1.	**Ericson Chandra Sihombing**	**121450026**
2.	**Rahmat Putra Aji**	**121450053**
3.	**Frisca Sihotang**	**121450092**
4.	**Ghozi Alvin Karim**	**121450123**

PROGRAM STUDI SAINS DATA

FAKULTAS SAINS

INSTITUT TEKNOLOGI SUMATERA

2024

ABSTRAK

Diabetes adalah penyakit serius yang menyebabkan peningkatan kadar gula darah dan memicu berbagai komplikasi kesehatan. Deteksi dini diabetes sangat penting untuk mencegah komplikasi serius. Metode klasifikasi data mining, seperti algoritma Naive Bayes, digunakan untuk memprediksi diabetes, tetapi memiliki keterbatasan dalam menangani data berdimensi tinggi. Penelitian ini menggabungkan algoritma Naive Bayes dengan Particle Swarm Optimization (PSO) untuk meningkatkan akurasi dan efisiensi klasifikasi diabetes. Dataset yang digunakan adalah dataset Pima Indian Diabetes dari Kaggle. Metode penelitian meliputi normalisasi data, pembagian data (train-test split), dan evaluasi kinerja menggunakan teknik cross-validation. Hasil penelitian menunjukkan bahwa kombinasi Naive Bayes dan PSO mampu meningkatkan akurasi klasifikasi, dengan nilai fitness optimal sebesar -0.751337 yang menunjukkan skor validasi silang sekitar 75.13%. Kesimpulan dari penelitian ini adalah bahwa PSO efektif dalam mengoptimalkan hyperparameter Naive Bayes, memberikan kontribusi signifikan dalam deteksi dini diabetes, dan meningkatkan kualitas hidup penderita diabetes.

Kata Kunci : Diabetes, Naïve Bayes, Particle Swarm Optimization (PSO), Klasifikasi.

I. PENDAHULUAN

1.1 Latar Belakang

Diabetes merupakan masalah serius yang menyebabkan peningkatan kadar gula darah secara berkelanjutan dalam tubuh manusia [1]. Selain itu, penyakit ini juga menjadi pemicu terjadinya berbagai komplikasi serius, termasuk stroke, penyakit ginjal, jantung koroner, gangguan pada mata dan saraf, bahkan dapat berujung pada amputasi dan kematian [2]. Organisasi Kesehatan Dunia (WHO) mencatat bahwa diabetes menyebabkan 1,5 juta kematian pada tahun 2012, dan diprediksi akan menjadi salah satu penyebab utama kematian di dunia pada tahun 2030 [3]. Untuk mengurangi dampak buruk yang ditimbulkan oleh diabetes, deteksi dini terhadap penyakit ini menjadi sangat penting guna mengantisipasi kemungkinan terjadinya komplikasi yang serius. Berbagai metode klasifikasi data mining telah diusulkan untuk memprediksi dan mengidentifikasi diabetes, salah satunya adalah algoritma Naive Bayes. Algoritma ini memiliki prinsip kerja berbasis probabilitas dan tergolong sederhana dalam implementasinya [4]. Namun, Naive Bayes memiliki keterbatasan dalam menangani data berdimensi tinggi, sehingga performanya dapat menurun. Untuk mengatasi kelemahan ini, Particle Swarm Optimization (PSO) hadir sebagai solusi. PSO merupakan teknik optimasi metaheuristik yang terinspirasi dari perilaku sosial pada kawanan burung atau ikan [5]. Algoritma ini mampu mencari solusi optimal secara efisien dan efektif, bahkan pada permasalahan kompleks dengan dimensi tinggi. Oleh karena itu, tugas besar kami kali ini adalah menggabungkan Naive Bayes dan PSO untuk membangun model klasifikasi diabetes yang lebih efisien dan akurat. Penerapan PSO diharapkan dapat meningkatkan kemampuan Naive Bayes dalam menangani data berdimensi tinggi dan menghasilkan kinerja klasifikasi yang lebih optimal. Dengan demikian, kombinasi antara Naive Bayes dan PSO diharapkan dapat memberikan kontribusi signifikan dalam deteksi dini diabetes, yang pada gilirannya dapat membantu mencegah terjadinya komplikasi serius serta meningkatkan kualitas hidup penderita diabetes.

1.2 Rumusan Masalah

Bagaimana penerapan algoritma Particle Swarm Optimization (PSO) yang efisien dapat meningkatkan kinerja algoritma Naive Bayes dalam klasifikasi penyakit diabetes?

1.3 Tujuan

1. Mengembangkan model klasifikasi penyakit diabetes yang mengintegrasikan Particle Swarm Optimization (PSO) dengan algoritma Naive Bayes.
2. Menguji efektivitas PSO dalam mengoptimalkan parameter algoritma Naive Bayes untuk meningkatkan akurasi klasifikasi penyakit diabetes.
3. Menganalisis dan membandingkan hasil kinerja model yang menggunakan PSO dengan model Naive Bayes standar tanpa optimasi.
4. Memberikan rekomendasi penggunaan metode PSO dan Naive Bayes dalam aplikasi medis untuk diagnosis penyakit diabetes berdasarkan hasil penelitian.

II. TEORI DASAR

2.1 Diabetes

Diabetes adalah penyakit kronis yang ditandai oleh peningkatan kadar glukosa darah, disertai dengan gangguan metabolisme lemak dan protein. Kadar glukosa darah meningkat karena tidak dapat dimetabolisme di dalam sel-sel, akibat kurangnya produksi insulin oleh pankreas atau ketidakmampuan sel-sel untuk menggunakan insulin dengan efektif yang diproduksi. Ada tiga jenis utama diabetes:

1. Tipe 1 di mana pankreas tidak memproduksi insulin
2. Tipe 2 di mana sel-sel tubuh kebal terhadap tindakan insulin yang diproduksi dan seiring waktu produksi insulin secara progresif menurun
3. Tipe 3 diabetes gestasional yang terjadi selama kehamilan dan dapat menyebabkan beberapa komplikasi selama kehamilan, pada saat melahirkan, serta meningkatkan risiko diabetes tipe 2 pada ibu dan obesitas pada bayi yang dilahirkan.

Selain itu, ada dua kategori intoleransi glukosa lainnya - intoleransi glukosa puasa (IFG) dan intoleransi glukosa puasa (IGT) yang merupakan kondisi perantara antara kadar glukosa darah normal dan diabetes, meskipun transisi tidak dapat dihindari. Orang dengan IFG dan IGT memiliki risiko peningkatan penyakit kardiovaskular dibandingkan dengan orang dengan nilai glukosa darah normal [6].

2.2 Algoritma Naive Bayes

Naive Bayes Classifier merupakan sebuah metoda klasifikasi yang berakar pada teorema Bayes. Metode pengklasifikasian dengan menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes. Ciri utama dari Naïve Bayes Classifierini adalah asumsi yang sangat kuat (naif) akan independensi dari masing-masing kondisi / kejadian. Menurut Olson Delen(2008) menjelaskan NaiveBayesuntuk setiap kelas keputusan, menghitung probabilitas dengan syarat bahwa kelas keputusan adalah benar, mengingat vektor informasi obyek. Algoritma ini mengasumsikan bahwa atribut obyek adalah independen. Probabilitas yang terlibat dalam memproduksi perkiraan akhir dihitung sebagai jumlah frekuensi dari ” master

” tabel keputusan.Naive Bayes Classifier bekerja sangat baik dibanding dengan model classifier lainnya. Hal ini dibuktikan oleh Xhemali, Hinde Stone dalam jurnalnya “Naïve Bayes vs. Decision Trees vs. Neural Networks in the Classification of Training Web Pages” mengatakan bahwa “Naïve Bayes Classifiermemiliki tingkat akurasi yang lebih baik dibanding model classifier lainnya”.Keuntungan penggunaan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (training data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Karena yang diasumsikan sebagai variable independent, maka hanya varians dari suatu variabledalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians [7].

a) Kegunaan NaiveBayes
    1. Mengklasifikasikan dokumen teks seperti teks berita ataupun teks akademis
    2. Sebagai metode machine learning yang menggunakan probabilitas
    3. Untuk membuat diagnosis medis secara otomatis
    4. Mendeteksi atau menyaring spam

b) Kelebihan NaiveBayes
    1. Bisa dipakai untuk data kuantitatif maupun kualitatif
    2. Tidak memerlukan jumlah data yang banyak
    3. Tidak perlu melakukan data training yang banyak
    4. Jika ada nilai yang hilang, maka bisa diabaikan dalam perhitungan
    5. Perhitungannya cepat dan efisien
    6. Mudah dipahami
    7. Mudah dibuat
    8. Pengklasifikasian dokumen bisa dipersonalisasi, disesuaikan dengan kebutuhan setiap orang
    9. Jika digunakan dalaam bahasa pemrograman, code-nya sederhana
    10. Bisa digunakan untuk klasifikasi masalah biner ataupun multiclass

c) Kekurangan Naive Bayes
    1. Apabila probabilitas kondisionalnya bernilai nol, maka probabilitas prediksi juga akan bernilai nol

2. Asumsi bahwa masing-masing variabel independen membuat berkurangnya akurasi, karena biasanya ada korelasi antara variabel yang satu dengan variabel yang lain
3. Keakuratannya tidak bisa diukur menggunakan satu probabilitas saja. Butuh bukti-bukti lain untuk membuktikannya
4. Untuk membuat keputusan, diperlukan pengetahuan awal atau pengetahuan mengenai masa sebelumnya. Keberhasilannya sangat bergantung pada pengetahuan awal tersebut Banyak celah yang bisa mengurangi efektivitasnya dirancang untuk mendeteksi katakata saja, tidak bisa berupa gambar.

d) Rumus Umum Algoritma Naïve Bayes

$$P(X) = \frac{P(C) * P(C)}{P(X)}$$

$P(X)$: probabilitas kelas C berdasrkan fitur X

$P(C)$: probabilitas fitur X berdasarkan kelas C

$P(C)$: probabilitas prior kelas C

$P(X)$: probabilitas prior fitur X

Kemudian rumus probabilitas likelihood $P(C)$ yang digunakan dengan megasumsikan atribut terdistribusi normal menggunakan distribusi Gaussian [8] dimana rumus probabilitas distribusi normal yaitu:

$$P(C) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{(x-\mu)^x}{2\pi\sigma^2}}$$

μ : rata-rata atau mean kelas C

σ : simpangan baku atau standar deviasi kelas C

2.3 Confusion Matrix

Confusion matrix digunakan dalam mengevaluasi kinerja dari model klasifikasi dengan algoritma naïve bayes dengan menampilkan informasi sebrapa baik model dalam memprediksi kelas-kelas dari data uji [9]. Dengan melakukan perhitungan

akurasi, recall,presisi,spesifikasi maupun F! score untuk meningkatkan hasil prediksi dengan pengujian pada data latih.

- Sensitivitas/Recall

$$\frac{TN}{(TP + TN)}$$

- Spesifikasi

$$\frac{TN}{(TN + FP)}$$

- Akurasi

$$\frac{(TP + TN)}{(TP + TN + FP + FN)}$$

- Presisi

$$\frac{TN}{(TN + TP)}$$

- F1 Score

$$\frac{2(Presisi * Recall)}{(Presisi + Recall)}$$

TN $(True\ Negative)$: jumlah negatif diprediksi benar

TP $(True\ Positive)$: jumlah positif diprediksi benar

FN $(False\ Negatif)$: jumlah negatif diprediksi salah

FP $(False\ Positif)$: jumlah positif diprediksi salah

2.4 Particle Swarm Optimization (PSO)

Particle Swarm Optimization (PSO) adalah algoritma optimisasi yang terinspirasi oleh perilaku kawanan serangga seperti semut, rayap, lebah, atau burung. PSO meniru perilaku sosial organisme ini, di mana setiap individu atau "partikel" dalam kawanan berinteraksi satu sama lain dan juga dengan lingkungan mereka. Dalam PSO, setiap partikel menggunakan kecerdasannya sendiri dan dipengaruhi oleh perilaku kolektif kelompok. Dalam konteks optimasi, PSO digunakan untuk mencari solusi terbaik dalam ruang multidimensi. Setiap partikel memiliki posisi dan kecepatan yang

diperbarui secara iteratif berdasarkan informasi tentang posisi terbaik yang pernah dicapai oleh partikel itu sendiri (pBest) dan posisi terbaik dari seluruh kawanan (gBest). PSO memiliki tiga komponen utama: partikel, komponen kognitif dan sosial, dan kecepatan partikel. Setiap partikel mewakili solusi potensial dalam ruang pencarian. Pembelajaran partikel terdiri dari dua faktor: pembelajaran kognitif (pBest) dan pembelajaran sosial (gBest). pBest merepresentasikan posisi terbaik yang pernah dicapai oleh partikel, sedangkan gBest merepresentasikan posisi terbaik dari seluruh kawanan. Informasi dari pBest dan gBest digunakan untuk menghitung kecepatan partikel, yang pada gilirannya digunakan untuk memperbarui posisi partikel dalam pencarian solusi [10].

a) Pseudocode

Pseudocode PSO menjadi struktur dalam algoritma Particle Swarm Optimization untuk mempermudah dalam membaca dan dipahami yang dimana tadinya merupakan bahasa pemrograman menjadi bahasa sehari-hari, pseudocode PSO sebagai berikut:

Begin
for setiap partikel
inisialisasi partikel
end
repeat
for partikel menghitung fitness
if new fitness lebih baik perbaharui nilai fitness (Pbest) pada partikel
end
end
pilih partikel yang merupakan nilai best fitness dari tetangga partikel
disimpan sebagai Gbest
for partikel menghitung kecepatan dan perbaharui
end
until stopping kriteria terpenuhi
end begin

b) Rumus PSO (Particle Swarm Optimization)

- Rumus update kecepatan/velocity

$$v_{i,j}^{t+1} = w.v_{i,j}^{t} + c_1.r_1\left(Pbest_{i,j}^{t} - x_{i,j}^{t}\right) + c_2.r_2\left(Gbest_{g,j}^{t} - x_{i,j}^{t}\right)$$

- Rumus update posisi

$$x_{i,j}^{t+1} = x_{i,j}^{t} + v_{i,j}^{t}$$

v : kecepatan

x : posisi

i : partikel

j : dimensi

$C1$: learning rates for cognition

$C2$: learning rates for social

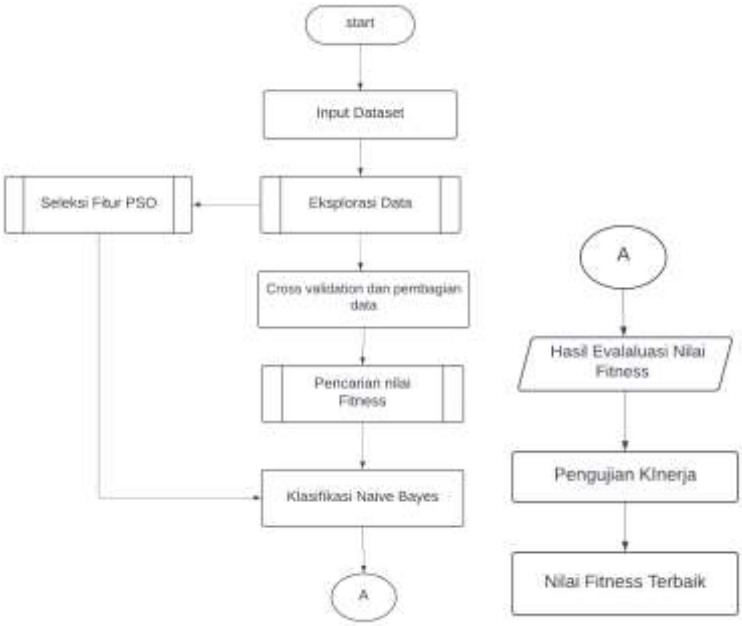
$Pbest$: vector dari best fitness

$Gbest$: index terbaik dari partikel

r : nilai random[0,1]

III. METODE

Dalam Metode yang digunakan dalam tugas ini terdiri dari beberapa tahapan yang dilakukan, berikut adalah tahapan-tahapan metode yang dilakukan :



Gambar 1. Tahapan Metode

3.1 Dataset

Data yang digunakan adalah dataset pima Indian diabetes yang di peroleh dari situs kagle.com dimana data ini terdiri dari 768 data dan 8 atribut atau fitur. Atribut dataset pima ditunjukkan oleh Tabel 1 berikut :

Tabel 1. Atribut dataset pima

Atribut	Keterangan	Label
Pregnancies	Angka kehamilan	X1
Glucose	Kadar glukosa 2 jam setelah makan. Menurut WHO salah satu yang menjadi kriteria penyakit diabetes, yaitu kadar glukosa minimal 200 mg/dl.	X2
Blood Pressure	Tekanan Darah	X3
Skin Thickness	Ketebalan Kulit	X4
Insulin	Insulin	X5

BMI	Berat Badan	X6
Diabetes Pedigree Function	Riwayat diabetes dalam keluarga	X7
Age	Umur	X8
Outcome	Status deabetes (1 = mengidap penyakit diabetes, 0 = tidak mengidap penyakit diabetes) .	Y

3.2 Pre-processing data

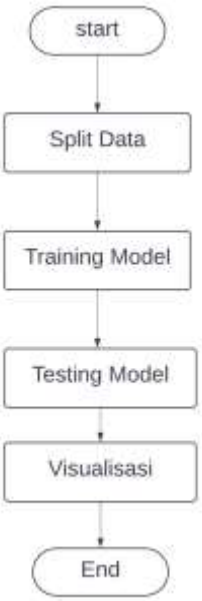
Tahapan pre-processing data terdiri dari dua tahapan. Tahapan pertama adalah normalisasi dataset Pima yang digunakan untuk memastikan bahwa semua fitur memiliki skala yang seragam. Tahapan kedua melibatkan seleksi fitur menggunakan algoritma Particle Swarm Optimization (PSO). Namun, tahapan seleksi fitur akan dilakukan setelah mendapatkan nilai akurasi dari algoritma Naive Bayes sebelum dilakukan seleksi fitur. Hal ini bertujuan untuk membandingkan akurasi algoritma Naive Bayes sebelum dan setelah seleksi fitur menggunakan PSO. Dengan demikian, akan diamati perbedaan akurasi antara algoritma Naive Bayes sebelum dilakukan seleksi fitur menggunakan PSO dengan algoritma Naive Bayes setelah dilakukan seleksi fitur menggunakan PSO.

3.3 Pembagian Data (Train-Test Split)

Dataset dibagi menjadi dua bagian, yaitu data pelatihan dan data pengujian. Model dilatih menggunakan data pelatihan dan kemudian dievaluasi menggunakan data pengujian yang tidak terlihat sebelumnya. Pendekatan ini memberikan perkiraan kinerja model pada data baru.

3.4 Klasifikasi Naïve Bayes

Pada tahap ini akan dilakukan proses klasifikasi menggunakan algoritma Naïve Bayes. Berikut adalah proses klasifikasi menggunakan algoritma Naïve Bayes :



Gambar 2. Tahapana Klasifikasi Naïve Bayes

3.5 Mengevaluasi Kinerja Model Naive Bayes Dalam Fungsi Fitness

Proses evaluasi dilakukan dengan menggunakan teknik cross-validation sebanyak lima kali (cv=5). Dalam setiap iterasi cross-validation, data pelatihan dibagi menjadi lima subset, di mana empat subset digunakan untuk melatih model dan satu subset digunakan untuk menguji model, dan proses ini diulang sehingga setiap subset digunakan sekali sebagai data uji. Skor performa model dari setiap iterasi cross-validation dihitung dan dirata-ratakan untuk mendapatkan nilai akhir. Rata-rata skor tersebut kemudian dikalikan dengan -1 untuk menghasilkan nilai fitness. Penggunaan nilai negatif ini penting karena algoritma Particle Swarm Optimization (PSO) yang digunakan dalam pencarian hyperparameter optimal bekerja dengan prinsip minimisasi nilai fitness, sehingga nilai fitness yang lebih kecil menunjukkan kinerja model yang lebih baik. Dengan demikian, metode ini menggabungkan cross-validation dan optimasi berbasis PSO untuk menemukan parameter var_smoothing yang menghasilkan kinerja terbaik bagi model Gaussian Naive Bayes pada data yang digunakan.

3.6 Metode Particle Swarm Optimization (PSO)

Fungsi pso menjalankan algoritma PSO untuk mengoptimalkan fungsi fitness dengan beberapa parameter utama, yaitu w (inertia weight), c1, dan c2 (koefisien untuk local best dan global best). Pertama, swarm diinisialisasi dengan sejumlah partikel yang

memiliki posisi dan kecepatan acak dalam batasan yang telah ditentukan. Selanjutnya, kecepatan dan posisi setiap partikel diperbarui berdasarkan formula PSO, yang mempertimbangkan inersia dari kecepatan sebelumnya, serta pengaruh dari posisi terbaik yang pernah dicapai oleh partikel tersebut (local best) dan posisi terbaik yang pernah dicapai oleh seluruh swarm (global best). Setelah pembaruan, nilai fitness dari setiap partikel dihitung dan dibandingkan dengan nilai fitness terbaik yang pernah dicapai oleh partikel tersebut dan oleh swarm secara keseluruhan. Proses ini berulang sampai jumlah iterasi maksimum tercapai, dengan tujuan untuk menemukan posisi terbaik yang meminimalkan fungsi fitness.

3.7 Pengujian Kinerja

Pada tahap ini, akan dilakukan pengujian kinerja untuk mengevaluasi performa algoritma Naïve Bayes dalam mengklasifikasi penyakit diabetes berdasarkan dataset yang tersedia. Selain itu, pengujian ini juga bertujuan untuk mengetahui peningkatan kinerja yang diperoleh setelah menerapkan algoritma PSO sebagai metode seleksi fitur. Evaluasi kinerja dilakukan dengan membandingkan nilai fitness yang diperoleh, di mana semakin tinggi nilai fitness menunjukkan kualitas model yang lebih baik.

IV. HASIL dan PEMBAHASAN

Pada Gambar 3 menunjjukkan bahwa data yang digunakan terdiri dari beberapa kolom yang berisi informasi tentang pasien diabetes. Setiap kolom memiliki atribut yang berbeda yang dapat digunakan untuk mengidentifikasi risiko diabetes pada pasien. Misalnya, kolom "Pregnancies" menunjukkan jumlah kehamilan pasien, yang dapat menjadi faktor risiko untuk diabetes. Demikian pula, kolom "Glucose" dan "Blood Pressure" menunjukkan kadar glukosa darah dan tekanan darah pasien, yang juga merupakan indikator penting untuk risiko diabetes. Kolom lain seperti "BMI" (Indeks Massa Tubuh) dan "Age" (Usia) juga memberikan informasi penting tentang risiko diabetes. Selain itu, kolom "Outcome" memberikan hasil tes toleransi glukosa oral pasien, di mana nilai 1 menunjukkan hasil tes abnormal yang dapat menandakan adanya diabetes. Dengan menganalisis data dalam tabel ini, peneliti dapat mengidentifikasi pola dan tren yang berkaitan dengan risiko diabetes, yang dapat membantu dalam diagnosis, pengelolaan, dan pencegahan kondisi tersebut.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

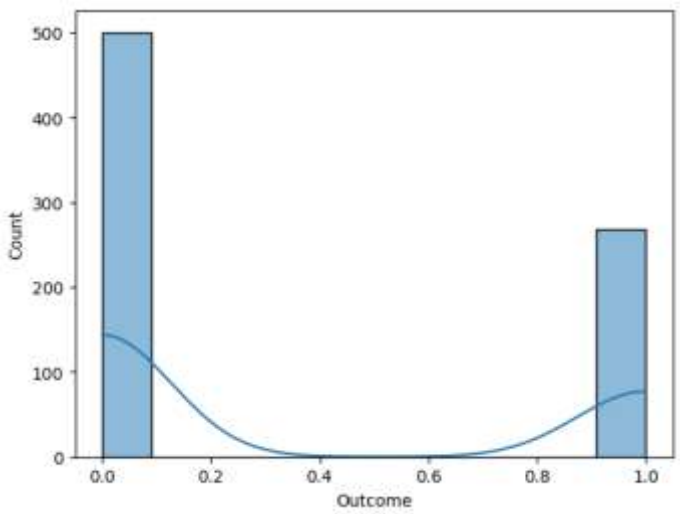
Gambar 3. Dataset pima sebelum eksplorasi

Ekplorasi data merupakan langkah penting dalam analisis data yang bertujuan untuk memahami karakteristik dan struktur dataset yang digunakan. Salah satu langkah dalam proses ini adalah mengidentifikasi nilai null dalam dataset dengan menjumlahkan jumlah null dalam setiap kolom. Hal ini krusial karena nilai null dapat memengaruhi hasil analisis dan memerlukan penanganan khusus. Selanjutnya, kita akan memperoleh ringkasan statistik seperti rata-rata, deviasi standar, dan kuartil untuk setiap kolom numerik. Informasi ini memberikan gambaran awal tentang sebaran dan pusat data serta membantu mengidentifikasi potensi outlier atau masalah lain yang perlu ditangani sebelum melanjutkan analisis lebih lanjut.

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
count	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000	768.000000
mean	3.845052	120.894531	69.105469	20.536458	79.799479	31.992578	0.471876	33.240885	0.348958
std	3.369578	31.972618	19.355807	15.952218	115.244002	7.884160	0.331329	11.760232	0.476951
min	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.078000	21.000000	0.000000
25%	1.000000	99.000000	62.000000	0.000000	0.000000	27.300000	0.243750	24.000000	0.000000
50%	3.000000	117.000000	72.000000	23.000000	30.500000	32.000000	0.372500	29.000000	0.000000
75%	6.000000	140.250000	80.000000	32.000000	127.250000	36.600000	0.626250	41.000000	1.000000
max	17.000000	199.000000	122.000000	99.000000	846.000000	67.100000	2.420000	81.000000	1.000000

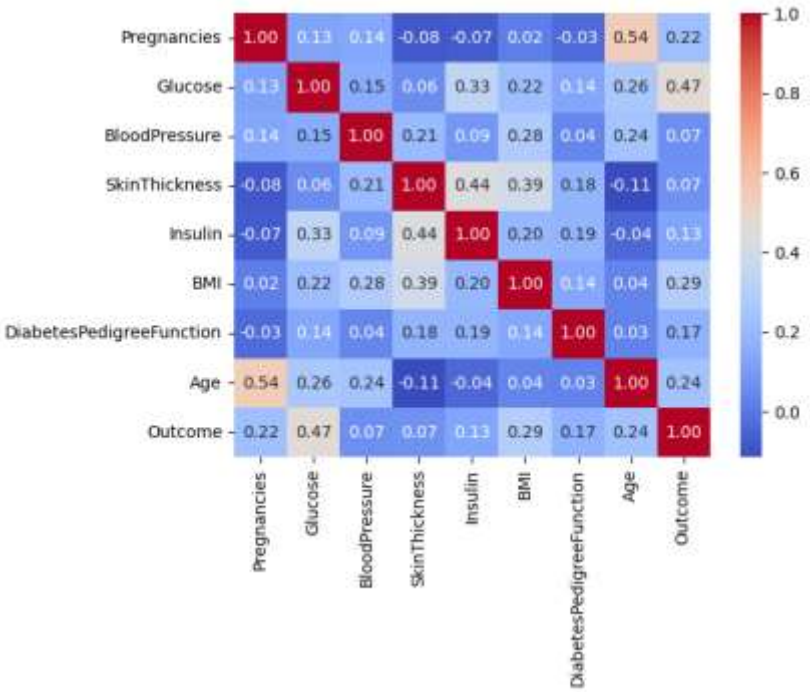
Gambar 4. Dataset pima setelah eksplorasi

Pada Gambar 5 menunjukkan sebuah histogram dan kurva kepadatan kernel (KDE) untuk variabel Outcome dalam dataset diabetes. Dataset ini berisi informasi tentang pasien diabetes, termasuk diagnosis diabetes (0: tidak diabetes, 1: diabetes) dan hasil tes toleransi glukosa oral (0: normal, 1: abnormal). Histogram di bagian bawah gambar menunjukkan distribusi frekuensi variabel Outcome**.** Histogram membagi nilai variabel 'Outcome' menjadi interval (bin) dan menunjukkan jumlah data yang terdapat dalam setiap interval. Mayoritas data (sekitar 50%) memiliki nilai 'Outcome' 0, menunjukkan prevalensi diabetes yang relatif rendah dalam dataset ini. Kurva KDE di bagian atas gambar merupakan distribusi probabilitas variabel 'Outcome'. Kurva ini dihasilkan dengan menghaluskan histogram menggunakan metode kernel density estimation. Distribusi variabel 'Outcome' tidak simetris, dengan ekor yang lebih panjang di sebelah kanan, menunjukkan adanya beberapa pasien dengan nilai 'Outcome' yang lebih tinggi, yang lebih mungkin mengalami diabetes..



Gambar 5. Histogram dan kurva kepadatan kernel (KDE)

Dengan menggunakan skala warna yang berkisar dari biru hingga merah, heatmap memperlihatkan korelasi negatif (biru) dan positif (merah) antara setiap pasangan variabel. Angka-angka dalam heatmap merepresentasikan koefisien korelasi antara variabel, di mana nilai 1 menunjukkan korelasi positif sempurna, nilai -1 menunjukkan korelasi negatif sempurna, dan nilai 0 menunjukkan tidak adanya korelasi. Hasil analisis menunjukkan beberapa korelasi yang signifikan, seperti korelasi positif yang kuat antara kadar glukosa darah (Glukosa) dan hasil tes toleransi glukosa oral (Outcome). Selain itu, terdapat korelasi positif yang kuat antara indeks massa tubuh (BMI) dan Outcome. Di sisi lain, terdapat korelasi negatif yang kuat antara usia (Usia) dan jumlah kehamilan (Kehamilan). Temuan ini memberikan wawasan penting tentang hubungan antara variabel-variabel dan dapat digunakan untuk mengidentifikasi faktor risiko potensial untuk diabetes, serta menyediakan landasan untuk analisis lebih lanjut dalam pemahaman terhadap kondisi pasien.



Gambar 6. Korelasi antara variabel-variabel dalam dataset

Pada algoritma Naive Bayes, dilakukan split data untuk proses membagi dataset menjadi dua subset: data pelatihan (training data) dan data pengujian (testing data). Tujuannya adalah untuk melatih model menggunakan data pelatihan dan menguji

kinerjanya dengan data pengujian. Data pelatihan digunakan untuk mengestimasi probabilitas kelas dan atribut yang diperlukan oleh model. Model yang telah dilatih kemudian diuji pada data pengujian yang belum pernah dilihat sebelumnya untuk mengukur kinerjanya secara objektif. Pembagian ini membantu mencegah overfitting dan memungkinkan evaluasi kinerja model pada data yang tidak dikenal.

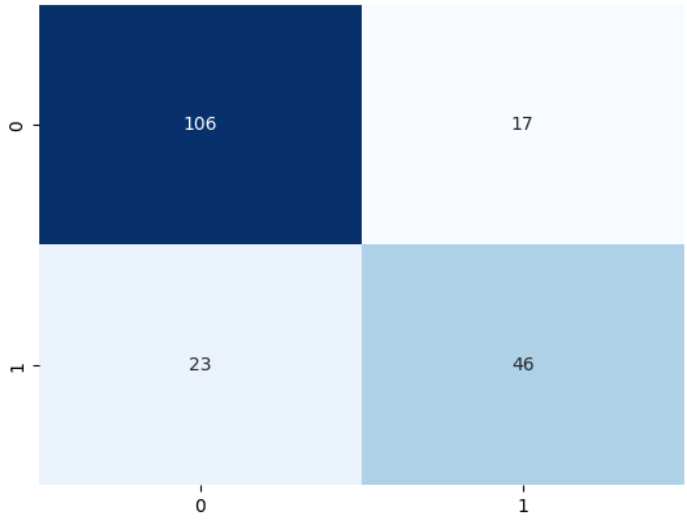
Dari hasil evaluasi pada gambar 7, diketahui bahwa model memiliki akurasi sekitar 79.17%, yang mengindikasikan seberapa baik model dalam memprediksi kelas yang benar secara keseluruhan. Laporan klasifikasi memberikan informasi rinci tentang precision, recall, dan F1-score untuk setiap kelas, serta rata-rata dari keseluruhan kelas, yang memungkinkan untuk mengevaluasi kinerja model dalam mengklasifikasikan data. Selain itu, nilai F1-score sekitar 0.697 memberikan gambaran tentang keseimbangan antara precision dan recall, yang penting untuk menilai performa model dalam mengklasifikasikan data secara menyeluruh.

```
> Akurasi Score :  0.7916666666666666

> Classification Report:
              precision    recall  f1-score   support

           0       0.82      0.86      0.84       123
           1       0.73      0.67      0.70        69

    accuracy                           0.79       192
   macro avg       0.78      0.76      0.77       192
weighted avg       0.79      0.79      0.79       192


> F1 Score: 0.696969696969697
```

Gambar 7. Evaluasi kinerja model klasifikasi

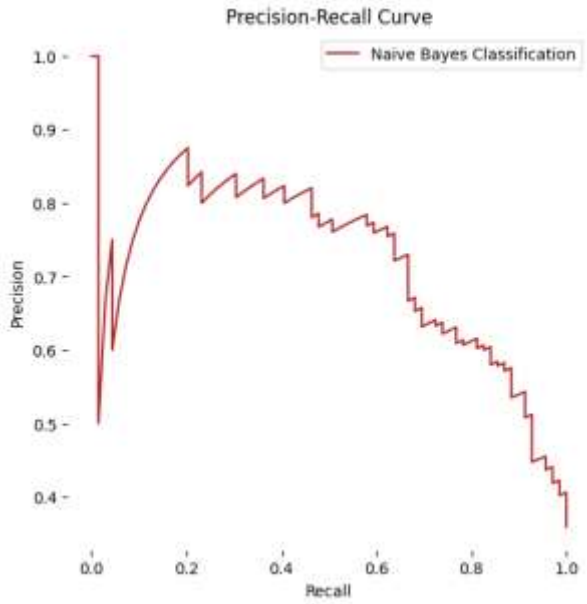
Confusion matrix pada gambar 8 memberikan gambaran detail tentang performa model klasifikasi dalam memprediksi kasus diabetes pada dataset tersebut. Nilai 106, yang merupakan True Negative (TN), menunjukkan bahwa model berhasil mengidentifikasi pasien yang sebenarnya tidak memiliki diabetes dengan tepat. Namun, terdapat 17 kasus False Positive (FP), yang menandakan bahwa model secara keliru memprediksi bahwa pasien memiliki diabetes padahal sebenarnya tidak. Di sisi lain, ada 23 kasus False Negative (FN), yang menunjukkan bahwa model gagal mengidentifikasi pasien yang sebenarnya memiliki diabetes sebagai negatif. Akhirnya, nilai 46, yang merupakan True Positive (TP), menunjukkan bahwa model berhasil

memprediksi dengan benar bahwa pasien menderita diabetes. Melalui matriks kebingungan ini, kita dapat mengevaluasi kekuatan dan kelemahan model dalam mengklasifikasikan kasus diabetes, serta memahami jenis-jenis prediksi yang dilakukan oleh model dengan lebih baik.



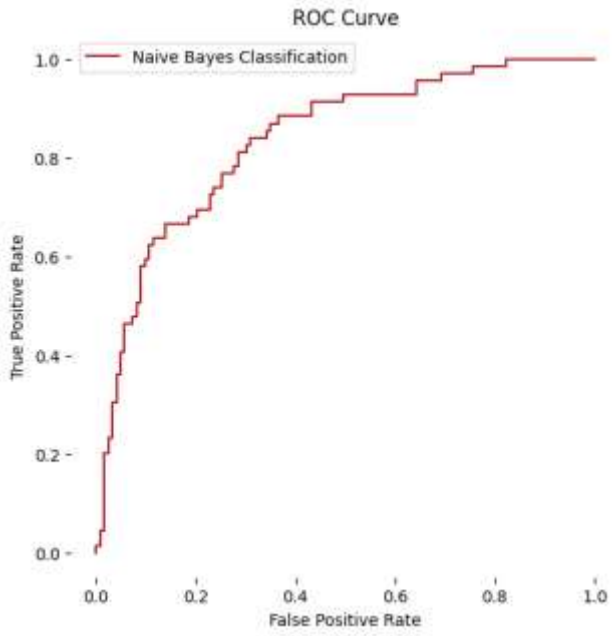
Gambar 8. Confusion matrix

Berdasarkan gambar 9 menjelasakan kurva Precision-Recall (PR) memberikan pemahaman mendalam tentang kinerja model klasifikasi biner Naive Bayes. Dimulai dari sudut kiri atas, yang menandakan presisi sempurna pada recall yang rendah, kurva menyoroti strategi awal model yang sangat konservatif. Namun, dengan penurunan ambang untuk mengklasifikasikan instansi sebagai positif, model menjadi lebih sensitif, meningkatkan recall tetapi mengorbankan presisi secara bertahap. Ini menggambarkan bahwa model cenderung mengidentifikasi lebih banyak kasus positif, namun dengan tingkat kesalahan positif palsu yang meningkat. Meskipun sulit untuk menentukan secara tepat AUC-PRC dari gambar, secara visual kurva menunjukkan kinerja yang cukup baik dengan luasan yang signifikan di bawahnya. AUC-PRC yang lebih besar menandakan kinerja yang lebih baik secara keseluruhan dalam menyeimbangkan presisi dan recall. Dengan demikian, analisis ini menunjukkan bahwa model Naive Bayes telah mencapai keseimbangan yang cukup baik antara presisi dan recall, meskipun masih mungkin untuk meningkatkan kinerja lebih lanjut tergantung pada kebutuhan aplikasi spesifiknya.



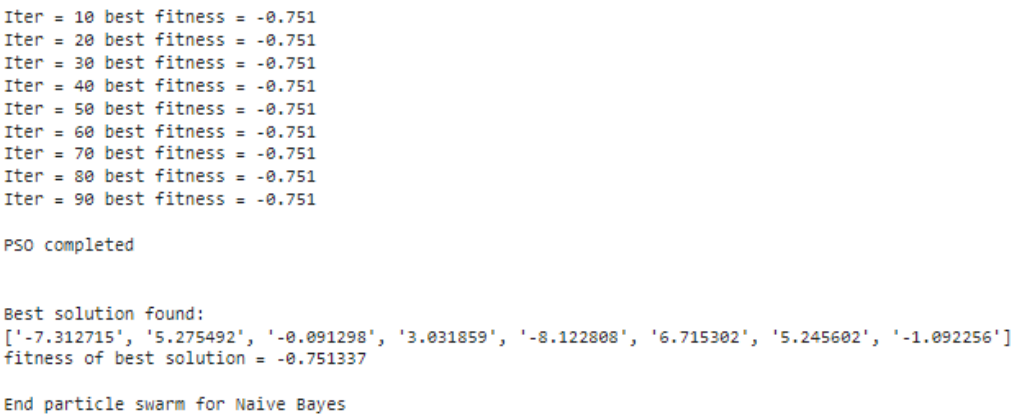
Gambar 9. kurva presisi-recall

Kurva ROC pada gambar 10 memberikan gambaran tentang bagaimana kinerja model berubah seiring dengan penyesuaian ambang klasifikasi. Titik-titik pada kurva mencerminkan perubahan dalam True Positive Rate (TPR) dan False Positive Rate (FPR) ketika ambang tersebut diubah. Titik yang lebih dekat ke sudut kiri bawah menunjukkan ambang yang lebih tinggi, di mana model lebih berhati-hati dalam mengklasifikasikan instance sebagai positif. Hal ini menghasilkan FPR yang lebih rendah (jumlah positif palsu yang lebih sedikit), tetapi juga TPR yang lebih rendah (jumlah positif benar yang lebih sedikit). Sebaliknya, titik yang lebih dekat ke sudut kanan atas menunjukkan ambang yang lebih rendah, di mana model lebih cenderung mengklasifikasikan instance sebagai positif. Ini menghasilkan TPR yang lebih tinggi (jumlah positif benar yang lebih banyak), tetapi juga FPR yang lebih tinggi (jumlah positif palsu yang lebih banyak).



Gambar 10. Kurva ROC

Pada gambar 11 menggunakan Particle Swarm Optimization (PSO) untuk mengoptimalkan hyperparameter model Gaussian Naive Bayes dalam klasifikasi data diabetes. Fungsi fitness mengevaluasi performa model dengan validasi silang 5-fold, menggunakan negatif dari rata-rata skor validasi silang sebagai nilai fitness. Kelas Particle merepresentasikan partikel dalam PSO, yang posisinya mencerminkan set hyperparameter dan kecepatannya menentukan perubahan posisi di iterasi berikutnya. Algoritma PSO mengelola partikel-partikel ini, memperbarui posisi dan kecepatan mereka, serta melacak posisi terbaik yang ditemukan. Hasil menunjukkan bahwa PSO berhasil menemukan hyperparameter terbaik dengan nilai fitness sekitar -0.751, menunjukkan performa model yang baik. Ini mengindikasikan bahwa PSO efektif dalam mengoptimalkan hyperparameter Naive Bayes untuk klasifikasi data diabetes.



Gambar 11. Klasifikasi Diabetes Menggunakan Particle Swarm Optimization

Berdasarkan gambar 11 menunjukan nilai fitness terbaik yang ditemukan oleh algoritma Particle Swarm Optimization (PSO) adalah -0.751337. Nilai fitness ini merupakan nilai negatif dari rata-rata skor validasi silang (cross-validation score) yang diperoleh dari model Naive Bayes Gaussian dengan menggunakan nilai hyperparameter var_smoothing yang optimal. Skor validasi silang adalah metrik evaluasi yang digunakan untuk menilai performa model pada data yang belum pernah dilihat sebelumnya. Semakin tinggi skor validasi silang, semakin baik generalisasi model pada data baru.

Nilai fitness -0.751337 yang ditemukan oleh algoritma PSO mengindikasikan bahwa model Naive Bayes Gaussian dengan nilai hyperparameter var_smoothing yang optimal memiliki rata-rata skor validasi silang sebesar 0.751337 atau sekitar 75.13% pada dataset diabetes yang digunakan. Skor validasi silang 75.13% dapat dianggap cukup baik untuk sebuah model klasifikasi pada dataset diabetes ini, tetapi masih ada ruang untuk perbaikan lebih lanjut.

Jadi, nilai fitness -0.751337 menunjukkan bahwa model Naive Bayes Gaussian dengan hyperparameter var_smoothing yang optimal memiliki kemampuan generalisasi yang cukup baik pada dataset diabetes yang digunakan dalam eksperimen ini.

V. KESIMPULAN

Secara keseluruhan, berdasarkan hasil dan pembahasan menunjukkan bahwa model Gaussian Naive Bayes dengan hyperparameter yang dioptimalkan menggunakan PSO mampu memberikan performa yang cukup baik dalam klasifikasi data diabetes. Meskipun akurasi dan skor validasi silang menunjukkan performa yang memadai, masih ada ruang untuk perbaikan lebih lanjut, baik melalui eksplorasi lebih dalam terhadap data maupun dengan penggunaan model dan teknik optimasi yang lebih canggih. PSO terbukti efektif dalam mengoptimalkan hyperparameter dan dapat digunakan sebagai pendekatan yang berguna untuk meningkatkan performa model dalam tugas klasifikasi serupa. Keseluruhan analisis menunjukkan pentingnya pemahaman dan eksplorasi data awal, evaluasi kinerja model, serta optimasi hyperparameter untuk mencapai model yang lebih akurat dan andal dalam klasifikasi kondisi medis seperti diabetes.

Lampiran :

Pengerjaan Menggunakan Colab : Tubes swarm intelligence_Kelompok 11_RA

DAFTAR PUSTAKA

[1] D. K. Choubey, P. Kumar, S. Tripathi, and S. Kumar, "Performance evaluation of classification methods with PCA and PSO for diabetes," *Network Modeling Analysis in Health Informatics and Bioinformatics*, vol. 9, no. 1, Dec. 2020, doi: 10.1007/s13721-019-0210-8.

[2] A. M. Argina, "Indonesian Journal of Data and Science Penerapan Metode Klasifikasi K-Nearest Neigbor pada Dataset Penderita Penyakit Diabetes," vol. 1, no. 2, pp. 29–33, 2020.

[3] D. Susilowati, S. Sutrisno, and M. Yunus, "Penerapan Particle Swarm Optimization Untuk Meningkatkan Kinerja Algoritma K-Nearest Neighbor Dalam Klasifikasi Penyakit Diabetes," *J-REMI : Jurnal Rekam Medik dan Informasi Kesehatan*, vol. 4, no. 3, pp. 176–184, Jun. 2023, doi: 10.25047/j-remi.v4i3.3980.

[4] G. Satya Nugraha, M. Nurkholis Abdillah, and M. Innuddin, "KOMPARASI AKURASI METODE CORRELATED NAIVE BAYES CLASSIFIER DAN NAIVE BAYES CLASSIFIER UNTUK DIAGNOSIS PENYAKIT DIABETES."

[5] "BAB II TINJAUAN PUSTAKA."

[6] "who_global_report_on_diabetes__a_summary.2".

[7] A. Felicia Watratan, A. B. Puspita, D. Moeis, S. Informasi, and S. Profesional Makassar, "Implementasi Algoritma Naive Bayes Untuk Memprediksi Tingkat Penyebaran Covid-19 Di Indonesia," 2020. [Online]. Available: http://journal.isas.or.id/index.php/JACOST

[8] A. Ashari Muin, "Metode Naive Bayes Untuk Prediksi Kelulusan (Studi Kasus: Data Mahasiswa Baru Perguruan Tinggi)," *Jurnal Ilmiah Ilmu Komputer*, vol. 2, no. 1, 2016, [Online]. Available: http://ejournal.fikom-unasman.ac.id

[9] M. Y. H. Setyawan, R. M. Awangga, and S. R. Efendi, "Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot," in *Proceedings of the 2018 International Conference on Applied Engineering, ICAE 2018*, Institute of Electrical and Electronics Engineers Inc., Dec. 2018. doi: 10.1109/INCAE.2018.8579372.

[10] I. Cholissodin, "Buku Ajar Swarm Intelligence," 2016. [Online]. Available: https://www.researchgate.net/publication/317706705