Implementasi Metode Gravitational Search Algorithm- Adaboost Untuk Pada Prediksi Diabetes Pada Anak Berdasarkan Data Ekspresi Gen

Mochammad Rafi Farid
Fakultas Informatika
Universitas Telkom, Bandung
mochammadrafifarid@students.telkomuniversity.ac.id

Isman Kurniawan
Divisi Digital Service
PT Telekomunikasi Indonesia
ismankrn@telkomuniversity.ac.id

Abstrak Diabetes merupakan penyakit parah yang terjadi pada saat insulin tidak dapat dihasilkan dengan baik pleh pankreas atau pada saat tubuh tidak bisa menggunakan insulin yang diproduksi oleh pankreas secara efektif, Insulin merupakan suatu hormon dengan fungsi mengontrol glukosa dalam darah. Diabetes melitus **tipe 1 (DMT1) sering terjadi pada anak dan remaja hingga 90%. Data yang berisi profil ekspresi gen pada anak-anak dengan T1D dan T2D, pengukuran dilakukan saat diagnosis pada awal dan diulang 4 bulan setelahnya , dan juga setelah mendapatkan pengobatan, maka Matriks ekspresi gen kemudian ditransposisikan dan tiga fitur demografis yang dianggap penting yaitu usia, jenis kelamin, dan ras. Setelah melakukan proses GSA, dataset akan dilakukan proses klasifikasi,** dengan **menggunakan metode utama yaitu Adaptive Boosting (AdaBoost), selanjutnya ditambahkan 2 metode ensemble sebagai pembanding yaitu KNeighbors (KNN), Multi- Layer Perceptron (MLP). Kemudian dilakukan hyperparameter tuning bertujuan untuk mencari niai yang paling optimal dengan meningkatkan kinerja pada model. Parameter scanning pada proses tuning dilakukan dengan menggunakan search cross validation (grid search CV). tersebut akan menjadi tolok ukuruntuk mengevaluaisi ketiga model yang digunakan sehingga diperoleh hasil paling optimal yakni AdaBoost dengan accuracy 0,666 dan F1-Score 0,769.**

Kata kunci: ***Diabetes melitus, Gravitational Search Algorithm , Multi-Layer Perceptron, Adaptive Boosting, KNeighbors***

Abstract
Diabetes is a serious disease that occurs when the pancreas cannot produce insulin properly or when the body cannot use insulin produced by the pancreas effectively. Insulin is a hormone with the function of controlling glucose in the blood. Type 1 diabetes mellitus (T1DM) 90% occurs in children and adolescents. Data containing gene expression profiles in children with T1D and T2D, measurements were taken at the time of initial diagnosis and repeated 4 months later, and also after receiving treatment, then the gene expression matrix was transposed and three demographic features were considered important, namely age, gender, and race. After carrying out the GSA process, the dataset will be classified, using the main method, namely Adaptive Boosting (AdaBoost), then 2 ensemble methods were added as a comparison, namely KNeighbors (KNN), Multi-Layer Perceptron (MLP). Then hyperparameter tuning was carried out to find the most optimal value by improving performance on the model. Parameter scanning in the tuning process is carried out using search cross validation (grid search CV). This will be the benchmark for evaluating the three models used to obtain the most optimal results, namely AdaBoost with an accuracy of 0.666 and an F1-Score of 0.769.

Keywords: Diabetes mellitus, Gravitational Search Algorithm, Multi-Layer Perceptron, Adaptive Boosting, KNeighbors

1. PENDAHULUAN

Diabetes merupakan penyakit parah yang terjadi pada saat insulin tidak dapat dihasilkan dengan baik pleh pankreas atau pada saat tubuh tidak bisa menggunakan insulin yang diproduksi oleh pankreas secara efektif, Insulin merupakan suatu hormon dengan fungsi mengontrol glukosa dalam darah. Hiperglikemia, sering disebut sebagai peningkatan glukosa dalam darah, selain itu efek diabetes yang tidak teratur dan berlajut menimbulkan rusaknya sistem dalam tubuh, terutama pembuluh darah dan jaringan saraf.[1]. Gejada dari penyakit ini adalah hilangnya berat badan, poliuria, polidipsi, dan polipagia. Pada diabetes, kekurangan insulin berdampak terjadinya peningkatan glukosa dalam darah dan menyebabkan perubahan metabolisme protein dan lemak[2].

DM tipe1 adalah suatu penyakit parah yang biasanya banyak dialami oleh anak. penanganan diabetes pada anak sangat rumit sehingga dibutuhkan bantuan dari orang tua untuk melewatinya[3]

Diabetes melitus tipe 1 (DMT1) sering terjadi pada anak dan remaja hingga 90%. Hal ini memiliki perbedaan di belahan dunia lainnya, contohnya hasil tertinggi tedapat di Finlandia sebesar 40/100.000 populasi dan populasi terendah berada di cina dengan 0,1/100.000 populasi. Menurut Data Unit Kerja Koordinasi (UKK) Endokrinologi Anak Pengurus Pusat Ikatan Dokter Anak Indonesia (PPIDAI), jumlah pasien DM tipe 1 pada tahun 2012 berjumlah 731 pasien [4].

Gejala diabetes pada anak, khususnya diabetes tipe 1, dapat muncul dengan cepat dan lebih parah dibandingkan dengan diabetes tipe 2. Berikut adalah gejala umum yang sering ditemukan pada anak dengan diabetes tipe 1 Sering Buang Air Kecil (Polyuria), Haus Berlebihan (Polidipsia), Nafsu Makan Meningkat (Polyphagia), Penurunan Berat Badan, Keletihan, Kaburnya englihatan, Luka Sulit Sembuh, Infeksi Berulang[5].

Salah satu cara untuk mendia DM tipe-1 perlu dilakukan tes hemoglobin A1c (HbA1c) untuk ngukur rata-rata gula dalam darah degan melakukan pengecekan dalan 2-3 bulan terakhir[6]. Dengan bayaknya cara dalam

pendeteksian diabetes, salah satu cara adalah dengan klasifikasi ekspresi gen dengan menggunakan machile learning. Ekspresi gen adalah sebuah rangaian data yang digunakan untuk megidentifikasi informasi gen dalam bentuk microarray (DNA dan RNA)[7]. Machine learning dapat mengkasifikasi data microarray pada ekspresi gen. dengan kata lain Machine learning mempunyai kemampuannya dalam menangani data yang memiliki dimensi yang tinggi[8].

Dalam Upaya memudahkan klasifikasi dan mengatasi permasalah pada Diabetes pada anak, penelitian ini menerapkan metode AdaBoost, dengan ditambahkan metode KNN, dan MLP sebagai pembanding, pada proses klasifikasinya dan dioptimalkan menggunakan Gravitational Search Algorithm. Beberapa penelitian terkait implementasi machine learning pada data ekspresi telah dilakukan untuk mengidentifikasi diabetes.

Pada tahun 2021, Fallucchi dkk, melakukan pengujian untuk Prediksi Resiko Diabetes Menggunakan MLP, hasul yang didapatkan untuk model keseluruhan lebih besar dari 0,939, dengan hasil terbaik lebih dari 0,96 untuk 1500 elemen berlabel sebagai kumpulan data pelatihan[9].

Pada tahun 2020, Iqbal Fathur, dkk melakukan analisis diabetes melitus tipe 2 dengan membandingkan SVM, Multilayer Perceptron (MLP) dan Xtreme Gradient Boosting (XBOOST) menggunakan data ekspresi gen. pada penelitian tersebut dapat disimpulkan bahwa SVM memperoleh hasil yang paling paling baik dalam klasifikasi dengan nilai akurasi 91,30%, Sedangkan MLP memperoleh nilai akurasi sebesar 78,26%, dan XBOOST medapatkan akurasi sebesar 73,91%. [8]

Pada tahun 2022, Madhubala dkk, melakukan penelitian untuk mempredisi diabetes. Penelitian ini menggunakan dataset PID untuk membangun pengklasifikasi untuk IBk, OneR, J48, Multilayer Perceptron, RandowForest, dan Naive Bayes. Model Nave Bayes menghasilkan kinerja yang baik dibandingkan dengan model lain , pada ANN yang dimodifikasi menggunakan multilayer perceptron, menghasilkan akurasi pelatihan 79% dan akurasi pengujian 77%[10].

Pada tahun 2020, Goudjerkan dkk, melakukan Klasifikasi Otomatis Penderita Diabetes dengan MLP. Hasil yang didapat pada akurasi hingga 95% untuk Klasifikasi.[11].

Penelitian terkait GSA telah pernah dilakukan pada tahun 2021 oleh Priyadarshini, R. untuk memprediksi akurasi diabetes meitus. aklasifikasi untuk dataset pima dihitung menggunakan metode GSA-K-means. Hasil yang diperoleh kemudian dibandingkan dengan hasil algoritma pengelompokan k-means secara terpisah dan dijalankan selama 10 iterasi. Hasil yang ditunjukkan di atas adalah rata-rata dari semua hasil iterasi Hasil pengamatan menunjukkan bahwa teknik GSA-k-means yang diusulkan mampu mendeteksi diabetes melitus dengan tingkat akurasi yang tinggi sebesar 86% sedangkan untuk klasifikasi akurasi full train and full test peengujian ini mendaparkan hasil 92%[12].

Sehubungan dengan penelitian diatas, penelitian ini menggunakan dataset Diabetes pada anak berdasarkan ekspresi gen menggunakan AdaBoost dengan ditambahkan metode KNN, dan MLP sebagai pembandin dalam mengklasifikasi dan dioptimalkan dengan Gravitational Search Algorithm. Pengujian model dilakukan dengan menerapkan Confusion Matrix dalam menguji seberapa baik sebuah model dengan data yang memiliki. Tujuan dari penelitian ini adalah klasifikasi penyakit Diabetes Melitus pada anak.

Topik dan Batasannya

Topik dan Batasan yang dibahas pada tugas akhir ini adalah megidentifikasi diabetes pada anak berdasarkan ekspresi gen berdasarkan dataset GSE9006. yang berisi profil ekspresi gen dengan penelitian yang difokuskan untuk meneliti T1D yang terdiri dari 162 sampel. Adapun batasan masalah pada penelitian ini adalah Gravitational Search Algorithm sebagai seleksi fitur. Dan pengklasifikasian menggunakan metode metode utama yaitu Adaptive Boosting (AdaBoost), selanjutnya ditambahkan 2 metode ensemble sebagai pembanding yaitu KNeighbors (KNN), Multi- Layer Perceptron (MLP).

Rumusan Masalah

Rumusan masalah yang diangkat pada penelitian ini adalah:

1. Bagaimana performa metode Gravitational Search Algorithm dalam melakukan klasifikasi pada identifikasi Diabetes pada anak?
2. Bagaimana pengaruh hyperparameter dalam melakukan klasifikasi pada identifikasi Diabetes pada anak?
3. Bagaimana performa AdaBoost dan 2 model pembanding yaitu KNN dan MLP dalam melakukan klasifikasi pada identifikasi Diabetes pada anak

Tujuan

Tujuan yang ingin dicapai pada penelitian ini adalah:

1. Mengetahui performa metode Gravitational Search Algorithm dalam melakukan klasifikasi pada identifikasi diabetes pada anak.
2. Mengetahui pengaruh hyperparameter dalam melakukan identifikasi dan tunning Diabetes Melitus pada anak
3. Mengetahui performa metode AdaBoost dan 2 model pembanding yaitu KNN dan MLP dalam melakukan klasifikasi pada identifikasi diabetes pada anak.

2. KAJIAN TEORI

2.1 Penelitian Terkait

Pada tahun 2021, Fallucchi dkk, melakukan pengujian untuk Prediksi Resiko Diabetes Menggunakan MLP, hasul yang didapatkan untuk model keseluruhan lebih besar dari 0,939, dengan hasil terbaik lebih dari 0,96 untuk 1500 elemen berlabel sebagai kumpulan data pelatihan[9].

Pada tahun 2020, Iqbal Fathur, dkk melakukan analisis diabetes melitus tipe 2 dengan membandingkan SVM, Multilayer Perceptron (MLP) dan Xtreme Gradient Boosting (XBOOST) menggunakan data ekspresi gen. pada penelitian tersebut dapat disimpulkan bahwa SVM memperoleh hasil paling paling baik dengan nilai akurasi 91,30%, Sedangkan MLP dapat memperoleh akurasi sebesar 78,26%, dan XBOOST mamperoleh akurasi sebesar 73,91%. [8]

Pada tahun 2022, Madhubala dkk, melakukan penelitian untuk mempredisi diabetes. Penelitian ini menggunakan dataset PID untuk membangun pengklasifikasi untuk IBk, OneR, J48, Multilayer

Perceptron, RandowForest, dan Naive Bayes. Model Nave Bayes menghasilkan kinerja yang baik dibandingkan dengan model lain , pada ANN yang dimodifikasi menggunakan multilayer perceptron, menghasilkan akurasi pelatihan 79% dan akurasi pengujian 77%[10].

Pada tahun 2020, Goudjerkan dkk, melakukan Klasifikasi Otomatis Penderita Diabetes dengan MLP. Hasil yang didapat pada akurasi hingga 95% untuk Klasifikasi[11].

Pada tahun 2019, Mohapatra dkk, melakukan pengujian Deteksi Diabetes Menggunakan MLP Performanya ditemukan lebih baik dibandingkan dengan metode sebelumnya dengan akurasi klasifikasi sebesar 77,5%[13].

Pada tahun 2022, Putry, N. M, dkk, melakukan Penelitian mengklasifikasikan diagnosis penyakit diabetes melitus dengan membandingan KNN dan Naïve Bayes. penelitian ini menggunakan lima pembagian data, nilai akurasi dari Naïve Bayes yaitu sebesar 80% lebih tinggi dibandingkan KNN dengan nilai akurasi tertinggi sebesar 75%. Lalu nilai recall tertinggi diperoleh KNN sebesar 0.92. Dan untuk nilai presisi diperoleh Naïve Bayes yaitu 0.86[14].

tahun 2021, Priyadarshini, R dkk . Penelitian terkait GSA untuk memprediksi akurasi diabetes meitus. Hasil yang didapatkan menunjukkan bahwa teknik GSA-k-means dapat mengkasilkan akurasi yang tinggi sebesar 86% sedangkan akurasi full train and full test mendaparkan hasil 92%[12].

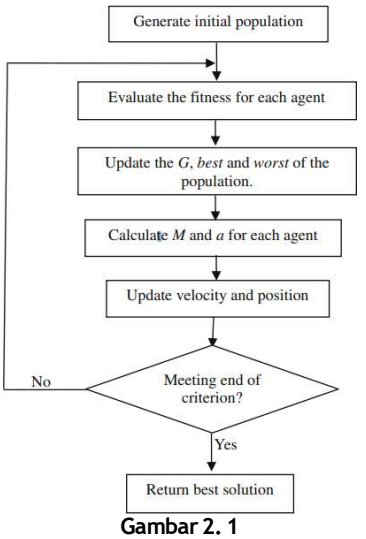
Pada tahun 2021, V Vaidya, dkk melakukan prediksi Diabetes Berbasis Data Mining menggunakan neural network yang dioptimalkan firefly algoritm,. Setelah menganalisis hasil tabel, dapat mengatakan bahwa pendekatan pengklasifikasi neural network yang dioptimalkan firefly algoritm mendapat akurasi maksimum 95,07%[15].

2.2 Gravitational Search Algorithm (GSA)

Gravitational Search Algorithm (GSA) adalah teknik optimasi heuristik berbasis populasi dan telah diusulkan untuk memecahkan masalah optimasi berkelanjutan[16].

Dalam GSA, kumpulan objek saling berinteraksi di bawah gravitasi Newton dan hukum gerak. Performa objek diukur berdasarkan massa. Semua objek ini saling tarik menarik dengan gaya gravitasi, sementara gaya menimbulkan gerakan pada seluruh objek ke arah objek yang memilik massa lebih berat[17].

Dalam GSA, massa (agen) terdapat empat spesifikasi sebagai berikut: posisi, massa inersia, massa gravitasi aktif, dan massa gravitasi pasif. Popada massa sesuai dengan solusi masalah, dan massa gravitasi dan fitness menentukan inersia. Dengan begitu, massa dapat menghasilkan solusi, lalu massa gravitasi inersia dengan disesuaikan degan algoritme pada navigasi. Dan massa ini akan menampilkan hasil optimum di ruang pencarian[18].



Gambar 2. 1
Flowchart Gravitational Search Algorithm (GSA)

Berikut merupakan penjelasan prinsip dasar GSA merujuk pada gambar 2.1:

Langkah 1: Agent initialization:

$$X_i = (X_i(1), \dots, X_i(d), \dots, X_i(n)) \; for \; i = 1,2, \dots, N. \quad (1)$$

$X_i(d)$ mempresentasikan posisi dari agen ke-i pada dimensi ke-d, mempresentasikandimensi ruang pencarian dan N mempresentasikan jumlah partikel.

Langkah 2: Fitness evolution and best fitness computation:

Selanjutnya, $best(t)$ dan $worst(t)$ dihitung. Untuk masalah minimisasi, definisi $best(t)$ dan $worst(t)$ diberikan sebagai berikut:

$$best(t) = \min_{j\in\{1,\dots,N\}} fit_j(t) \quad (2)$$

$$worst(t) = \max_{j\in\{1,\dots,N\}} fit_j(t) \quad (3)$$

Untuk masalah maximization, (2) dan (3) diubah menjadi (4) dan (5), masing-masing.

$$best(t) = \min_{j\in\{1,\dots,N\}} fit_j(t) \quad (4)$$

$$worst(t) = \max_{j\in\{1,\dots,N\}} fit_j(t) \quad (5)$$

Langkah 3: Gravitational constant (G) computation:

$$G(t) = G_0 \, e^{-\beta \frac{t}{T}} \quad (6)$$

di mana T adalah jumlah iterasi maksimum, G_0 dan β adalah nilai konstan. Konstanta gravitasi adalah fungsi waktu yang menurun di mana ia dinilai sebagai G_0 di awal dan secara eksponensial menurun menuju nol saat iterasi meningkat

untuk mengendalikan akurasi pencarian.

$$F(x) = \sum_{t=1}^{T} \propto_t h_t(x)$$

Langkah 4: Masses of the agents' calculation:
Massa gravitasi dan inersia kemudian diperbarui menggunakan (6) dan (7):

$$m_i(t) = \frac{fit_i(t) - worst(t)}{best(t) - worst(t)} \quad (7)$$

$$M_i(t) = \frac{m_i(t)}{\sum_{j=1}^{N} m_j(t)} \quad (8)$$

di mana $M_i(t)$ adalah massa inersia agen ke-i. Percepatan, α, dari massa i pada t dalam dimensi ke-d dihitung sebagai berikut:

Langkah 5: Accelerations of agents' calculation:

$$\propto_i (t,d) = \frac{F_i(t,d)}{M_i(t)} \quad (9)$$

$F_i^d(t)$Dihitung dengan persamaan berikut

$$F_i(t,d) = \sum_{j=1, j\neq 1}^{N} (rand_j F_{ij}(t,d) \quad (10)$$

$$F_{ij}(t,d) = G(t) \frac{M_j(t) \times M_i(t)}{R_{ij}(t) + \varepsilon} (x_j(t) - x_i(t,d)) \quad (11)$$

ε adalah konstanta kecil.
$R_{ij}(t)$ adalah jarak Euclidian antara dua agen i dan j
$rand_j$ adalah bilangan acak yang terdistribusi secara seragam antara 0 dan 1..

Langkah 6: Velocity and positions of agents:
Kecepatan dan posisi agen kemudian diperbarui menggunakan (12) dan (13), masing-masing seperti yang diberikan oleh:

$$v_i(t+1,d) = rand_i x v_i(t,d) + x_i(t,d) \quad (12)$$

$$x_i(t+1,d) = x_i(t,d) + v_i(t+1,d) \quad (13)$$

di mana i rand adalah angka acak yang terdistribusi secara seragam antara 0 dan 1.

Langkah 7: Ulangi Langkah 2 sampai Langkah 6
Algoritma tersebut akan berulang hingga kondisi berhenti terpenuhi, baik jumlah iterasi maksimum tercapai atau jumlah kesalahan tertentu diperoleh.

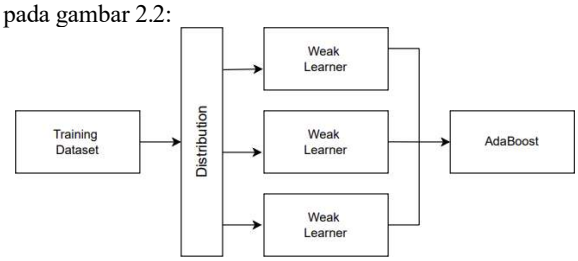
2.3 Adaptive Boosting (AdaBoost)

(14) AdaBoost merupakan salah satu algoritma yang terkenal dalam pembuatan klasifikasi ensemble dengan memilih anggota terlemah didalam klasifikasi. AdaBoost menghasilkan klasifikasi kuat yang dikombinasikan dengan beberapa klasifikasi lemah[19].

Karakteristik AdaBoost adalah menggunakan data pelatihan awal untuk menghasilkan pembelajar lemah, kemudian menyesuaikan distribusi data pelatihan sesuai dengan kinerja prediksi untuk pelatihan pembelajar lemah putaran berikutnya. Perhatikan bahwa sampel pelatihan dengan akurasi prediksi rendah pada langkah sebelumnya akan mendapat lebih banyak perhatian pada langkah berikutnya[20].

Setiap masukan pelatihan telah diberi bobot, bobot yang lebih besar diberikan pada masukan pelatihan yang belum dilatih oleh pembelajar sebelumnya. Penyesuaian bobot ini mengarahkan model yang diusulkan untuk lebih memperhatikan kesalahan di babak berikutnya, hal ini secara signifikan mengurangi masalah kesalahan identifikasi atau identifikasi yang hilang[21].

Berikut merupakan ilustrasi dari Adaboost yang ditunjukkan pada gambar 2.2:



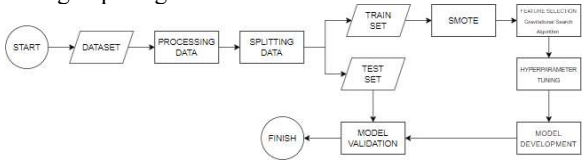
GAMBAR 2.2
Ilustrasi AdaBoost

AdaBoostmemiliki persamaan sebagai berikut:

Keterangan:
$h_t(x)$: Pengklasifikasi dasar atau lemah
$\propto_t$: Tingkat Pembelajaran (learning rate)
$F(x)$: Hasil berupa pengklasifikasi kuat atau akhir

3. METODE

Gravitational Search Algorithm (GSA) akan dikombinasikan dengan model untuk diklasifikasi menjadi seleksi fitur dan akan dikombinasikan dengan model sebagai klasifikasi. Berikut merupakan skema yang akan dibangun pada gambar 3.1



GAMBAR 2.1
skema yang akan dibangun

3.1 Dataset

Dalam penelitian ini data yang digunakan diambil dari

GSE9006. berisi profil ekspresi gen pada anak- anak dengan T1D dan T2D, pengukuran dilakukan saat diagnosis pada awal dan diulang 4 bulan setelahnya, dan juga setelah mendapatkan pengobatan. Total sampel yang didapatkan sebesar 234 sampel, yang terdiri dari 162 sampel anak dengan T1D, 24 sampel anak dengan T2D dan 48 adalah anak yang sehat.

Dikarenakan profil ekspresi gen dapat terpengaruh oleh pengobatan, Matriks ekspresi gen kemudian ditransposisikan dan tiga fitur demografis yang dianggap penting yaitu usia, jenis kelamin, dan ras. Dengan rentang umur anak sehat dan penderita Diabetes Melitus yaitu diantara umur 2-18 tahun.

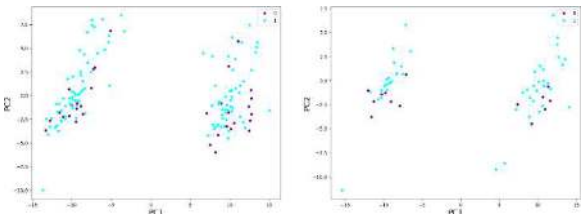
Dataset dibagi menjadi 2 Data Train train dan Data Test, jumlah dari kedu data tersebut dilihat pada tabel 1:

TABEL 1
Jumlah data train dan data test

	Normal	***Diabetes Milletus Type-1***
Data Train	34	113
Data Test	14	49
Total	48	162

Pada Tabel 1, ditunjukkan bahwa jumlah kelas pada normal dan Diabetes Melitus Type-1 memiliki jumlah data yang berbeda dan dengan begitu data tersebut tidak seimbang. Selanjutnya akan dilakukan SMOTE agar data seimbang antara normal dan Diabetes Milletus Type-1

Berikut persebaran label pada setiap data yang dapat dilihat pada gambar 3.2:



GAMBAR 3. 1
Persebaran Setiap Data (a) Data Train dan (b) Data Test

3.2 Praproses Data

Synthetic Minority Oversampling Technique (SMOTE)

Dilakukan nya SMOTE bertujuan untuk menangani kumpulan data yang tidak seimbang, pengklasifikasi sering kali mengutamakan kelas mayoritas, mengabaikan sampel kelas minoritas. untuk menyeimbangkan kumpulan data. SMOTE menghasilkan sampel sintetis dengan melakukan interpolasi antara sampel kelas minoritas di ruang fitur. Secara khusus, SMOTE memilih sampel acak dari kelas minoritas.

Dengan menggabungkan SMOTE, secara efektif dapat menyeimbangkan antara jumlah dataset train yang berbeda dengan dataset normal dan dataset penderita diabetes type-1[22].

3.3 Seleksi fitur

Pada penelitian ini seleksi fitur yang dilakukan menggunakan Gravitational Search Algorithm adalah pendekatan yang memanfaatkan prinsip fisika gravitasi untuk memilih fitur terbaik dalam dataset. Parameter yang digunakan pada Gravitational Search Algortihm yaitu Massa (Mass), Gravitational Constant (G), Kecepatan (Velocity), Posisi (Position), Iterasi (Iteration Count), Parameter Kontrol (Control Parameters), Daya Tarik (Attraction), Penghentian Kriteria (Stopping Criteria)

3.4 Ensemble Method

Selanjutnya dataset akan dilakukan proses klasifikasi. Dalam proses klasifikasi ini akan menggunakan metode utama yaitu Adaptive Boosting (AdaBoost), selanjutnya ditambahkan 2 metode ensemble sebagai pembanding yaitu KNeighbors (KNN), Multi-Layer Perceptron (MLP). Kemudian,dilakukan hyperparameter tuning bertujuan untuk meningkatkan kinerja model. Parameter scanning pada proses tuning dilakukan dengan menggunakan search cross validation (grid search CV). AdaBoost sebagai metode utama memilki tipe parameter optimasi yaitu n_estimators dan learning_rate, lalu KNN memiliki tipe parameter optimasi n_neighbors, weights, metric, dan MLP memilki tipe parameter optimasi yaitu hidden_layer_sizes, activation, solver, max_depth, dan learning_rate.

TABEL 2
Parameter dan Rentang untuk Nilai **MLP, AdaBoost, *KNeighbors***

Metode	Parameter	Rentang Nilai
AdaBoost	*n_estimators*	[50, 100, 150, 200, 250]
	learning_rate	[0.0001, 0.001, 0.01, 0.1, 1.0]
KNeighbors	*n_neighbors*	[5,7,9,11,13,15]
	weights	['uniform','distance']
	metric	['minkowski','euclidean','manhattan']
Multi-Layer Perceptron	*hidden_layer_sizes*	[(10,30,10),(20,)]
	activation	['tanh', 'relu']
	solver	['sgd', 'adam']
	max_depth	[0.0001, 0.05]
	learning_rate	['constant','adaptive']

3.5 Validasi Model

Setelah melakukan proses prediksi, maka dilakukan validasi model untuk mengevaluasi kualitas dari model klasifikasi yang sudah dibangun. Kinerja algoritma machine learning biasanya dievaluasi oleh Confusion Matrix seperti yang diilustrasikan pada tabel 1. Kolom adalah Predicted class dan baris adalah Actual class .

Dalam confusion matrix Terdapat empat istilah yang merupakan respresentasi hasil proses klasifikasi pada confusion matrix yaitu TP(True Positive) adalah jumlah contoh positif yang benar diklasifikasikan, TN(True Negative) adalah umlah contoh positif yang diklasifikasikan secara benar, FN(False Negative) adalah jumlah contoh positif yang diklasifikasikan secara salah

sebagai negatif, dan FP(False Positive) adalah jumlah contoh negatif yang diklasifikasikan secara salah sebagai positif.
Berikut merupakan confusion matrix untuk kelas prediksi dan kelas aktual yang terdapat pada tabel 4:[23]

TABEL 3
Confusion Matrix

	Predicted Negatif	*Predicted* Positif
Actual Negatif	TN	FN
Actual Positif	FP	TP

Kemudian, parameter pada confusion matrix akan digunakan ddalam pernghitung accuracy (Q), precision (PR), recall/Sensitivity(SE) dan F1-Score (F1). Dengan persamaan berikut :

$$Q = \frac{TP + TN}{TP + TN + FP + FN} \quad (15)$$

$$PR = \frac{TP}{(TP + FP)} \quad (16)$$

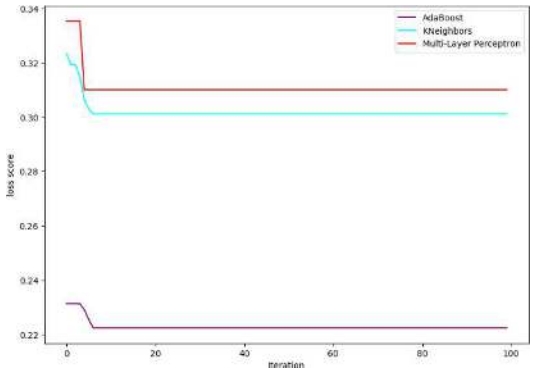
$$SE = \frac{TP}{(TP + FN)} \quad (17)$$

$$F1 = 2 \times \frac{PR \times RC}{(PR + RC)} \quad (18)$$

4. HASIL DAN PEMBAHASAN

4.1 Seleksi Fitur

Pada proses ini, dilakukan seleksi fitur dengan menggunakan Gravitational Search Algorithm (GSA) dengan menggabungkan model 3 model klasifikasi Ensemble yaitu AdaBoost, Kneighbors dan Multi-Layer Perceptron. Pada proses ini akan melakukan pengujian performa dengan mencari nilai loss score, loss score sendiri bertujuan untuk memilih fitur yang paling relevan dengan hasil terbaik.
Berikut merupakan pengujian performa nilai loss score dari iterasi yang diuji di ketiga metode Ensemble:



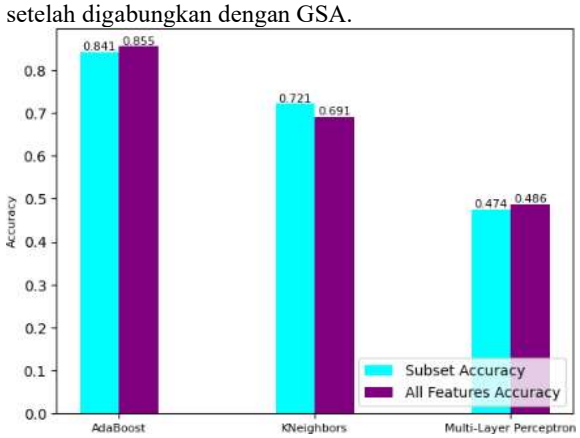
GAMBAR 4. 1
Convergence Data *Multi-Layer Perceptron*, *AdaBoost* dan *XGBoost*

Perubahan objective score selama proses iterasi ditunjukkan melalui convergence plot pada gambar 4.1. Dari ketiga metode Ensemble diperoleh hasil penurunan skor pada awal proses iterasi dan setelah itu hasil iterasi mendatar yang menandakan bahwa jumlah iterasi yang digunakan sudah mencapai nilai optimum.Dari ketiga metode Ensemble, diperoleh hasil terbaik yaitu Adaboost dengan hasil skor loss 0.222 dengan fitur terbaik sebanyak 76 fitur. Sedangkan *Kneighbors* dengan jumlah fitur terbaik sebanyak 75 fitur dengan hasil *loss* dan 0.301, dan *Multi-Layer Perceptron* dengan jumlah fitur terbaik sebanyak 80 fitur dengan hasil *loss* sebesar 0.310. berikut merupakan hasil convergence akan ditunjukan pada tabel Pada pada tabel 5 dibawah ini.

TABEL 4
Hasil Seleksi Fitur

Model	Jumlah Fitur Terbaik	Hasil Loss Score
AdaBoost	76	0.222
KNeighbors	75	0.301
Multi-Layer Perceptron	80	0.310

Merujuk pada gambar 4.3 diperoleh score default parameter dan score best parameter dari ketiga model. Diantara AdaBoost, KNN, dan *Multi-Layer Perceptron*. Score default parameter dan score best parameter pada AdaBoost diperoleh hasil selisih score yaitu 0,014 dengan hasil masing-masing 0,846 dan 0,885. Score default parameter dan score best parameter pada MLP diperoleh selisih sebesar 0,054 dengan hasil masing- masing 0,536 dan 0,482.berbeda dengan KNN pada metode ini hasil yang didapatkan berbanding terbalik dengan kedua metode sebelumnya, KNN tidak mengalami perubahan setelah digabungkan dengan GSA.



GAMBAR 4. 2
Perbandingan Akurasi Tanpa dan dengan GSA

4.2 Hyperparameter tuning

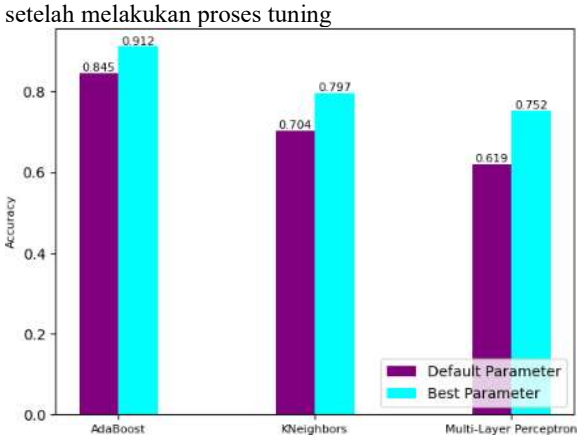
hyperparameter tuning mempunyai tujuan untuk mencari niai yang paling optimal dengan meningkatkan kinerja

pada model, seperti akurasi, generalisasi, dan interperbabilitas. Dalam proses ini seluruh model yang dipakai memiliki parameter yang sama untuk nilai default dan nilai terbaik setelah proses tuning. hyperparameter tuning, menghasilkan kombinasi yang unik dari setiap parameter yang selanjutkan akan menghasilkan score accuracy yang berbeda, lalu akan diambil hasil terbaik dengan kombinasi parameter terbaik nya. Dan hasil tersebut akan menjadi tolok ukuruntuk mengevaluaisi ketiga model tersebut.

TABEL 5
Nilai Parameter Terbaik

Metode	Parameter	Nilai Parameter Default	Nilai Parameter Terbaik Setelah Tuning
AdaBoost	*n_estimators*	50	200
	learning_rate	1,0	1,0
KNeighbors	*n_neighbors*	5	5
	weights	*uniform*	*Distance*
	metric	*minkowski*	*manhatta*
Multi-Layer Perceptron	*hidden_layer_sizes*	100	20
	activation	*relu*	tanh
	solver	*adam*	*adam*
	max_depth	*0.0001*	0,05
	learning_rate	*constant*	*constant*

Pada tabel 5, ditunjukkan parameter untuk setiap model dan nilai parameter default dan nilai parameter terbaik setelah melakukan proses tuning



GAMBAR 4. 3
Score Default dan Best Parameter

Merujuk pada gambar 4.3 diperoleh score default parameter dan score best parameter dari ketiga model. Diantara, AdaBoost KNN dan *Multi-Layer Perceptron*. Score default parameter dan score best parameter pada. Score default parameter dan score best parameter pada AdaBoost diperoleh dengan hasil masing-masing 0,845 dan 0,912, Score default parameter dan score best parameter KNN diperoleh dengan hasil masing-masing 0,704 dan 0,797. dan MLP diperoleh selisih sebesar dengan hasil masing-masing 0,482 dan 0,721. Sehingga dapat disimpulkan bahwa AdaBoost diperoleh hasil terbesar dengan selisih score yaitu 0,061 dengan hasil Score default parameter 0,845 dan score best parameter 0,912.

4.3 Validasi Data

Setelah dilakukannya Hyperparameter tuning, pada tabel 7 menunjukan validasi untuk semua model yang dibangundengan kombinasi parameter yang terbaik. Pada Data Train untuk AdaBoost, dan KNeighbors memperoleh hasil yang sama, untuk Confusion Matrix, hal tersebut menyebabkan nilai Accuracy, Precision, Recall dan F1-Score memilki nilai yang sama pula. Sedangkan untuk Multi-Layer Perceptron memperoleh hasil yang berbeda,

TABEL 6
Hasil Validasi Model

Model	TP	FP	TN	FN	Q	PR	SE	F1-Score
Train								
AdaBoost	113	0	113	0	1,0	1,0	1,0	1,0
KNeighbors	113	0	113	0	1,0	1,0	1,0	1,0
Multi-Layer Perceptron	113	0	4	109	0,517	1,0	0,035	0,068
Test								
AdaBoost	7	7	35	14	0,666	0,833	0,714	0,769
KNeighbors	11	3	27	22	0,603	0,9	0,551	0,683
Multi-Layer Perceptron	12	2	24	25	0,571	1.0	0.035	0.068

Berdasarkan data yang tercantum pada tabel 7, klasifikasi menggunakan data train untuk **AdaBoost dengan 76 fitur mendapatkan nilai *accuracy* 0,666 dan *f1-score* 0,769.** Hasil ini menunjukan jika adaboost lebih optimal dibandingkan KNeighbors dengan 75 fitur menghasilkan nilai *accuracy* 0,603 dan *f1-score* 0,683 dan dengan Multi-Layer Perceptron dengan 80 fitur mendapatkan nilai *accuracy* 0,571 dan *f1-score* 0.068.

5. KESIMPULAN

Pada penelitian ini, didapat hasilkan dengan menggunakan Gravitional Search Algorithm dan dikombinasikan dengan model dapat menghasilkn data microarray dalam mengidentifikasi Diabetes Melitus pada anak. Pada proses seleksi fitur dilakukan dengan menggabungkan dan mengkombinsaikan Gravitation Search Algorithm dengan Ensemble method yaitu, Adaptive Boosting, Kneighbors, dan Multi-Layer Perceptron dan juga objective function mampu mengurangi jumlah fiur dengan mempertimbangkan nilai loss score dari itersi yang telah diuji.

method yaitu Adaptive Boosting, Kneighbors, dan Multi-Layer Perceptron meningkatkan kinerja dengan prediktif model tunggal dengan melakukan trainng model dan melakukan penggabugan prediksi yang telah dilakukan. Untuk meningkatkan performa kinerja model dilakukan proses hyperparameter tuning, menghasilkan

kombinasi yang unik dari setiap parameter yang selanjutkan akan menghasilkan score accuracy yang berbeda, selanjutnya akan diambil hasil terbaik dengan kombinasi parameter terbaik nya. Dan hasil tersebut akan menjadi tolok ukur untuk mengevaluasi ketiga model tersebut. sehingga diperoleh hasil paling optimal yakni AdaBoost dengan accuracy 0,666 dan F1-Score 0,76.

REFERENSI

[1] Direktorat P2PTM, "Tanda dan Gejala Diabetes ."

[2] E. T. Faisal, "Demografi Diabetes Melitus Tipe-I pada Anggota Ikatan Keluarga Penyandang Diabetes Anak dan Remaja (IKADAR)," *Majalah Kedokteran Bandung*, vol. 42, no. 2, pp. 82–85, 2010.

[3] A. Ispriantari and D. P. Priasmoro, "PERBEDAAN TANGGUNG JAWAB ANAK DAN ORANG TUA DALAM PENGELOLAAN DIABETES ANAK DENGAN DM TIPE 1 DI KOTA MALANG," *Jurnal Kesehatan Hesti Wira Sakti*, vol. 7, no. 1, pp. 33–39, 2019.

[4] W. Wibisono and H. A. Tjahjono, "Hubungan Kadar 25-Hidroksi-Vitamin D dengan HbA1c Melalui Interleukin-17 pada Anak Diabetes Melitus Tipe 1," *Sari Pediatri*, vol. 17, no. 6, pp. 469–477, 2016.

[5] A. B. Pulungan, G. Fadiana, and D. Annisa, "Type 1 diabetes mellitus in children: experience in Indonesia," *Clinical Pediatric Endocrinology*, vol. 30, no. 1, pp. 11–18, 2021.

[6] S. Afzali and O. Yildiz, "An effective sample preparation method for diabetes prediction.," *Int. Arab J. Inf. Technol.*, vol. 15, no. 6, pp. 968–973, 2018.

[7] P. D. Pakan, "KLASIFIKASI KANKER LEUKIMIA MENGGUNAKAN MICROARRAY EKSPRESI GEN," *Jurnal Ilmiah Flash*, vol. 4, no. 2, pp. 78–83, 2018.

[8] I. F. Rahman, "Implementasi Metode SVM, MLP dan XGBoost pada Data Ekspresi Gen (Studi Kasus: Klasifikasi Data Ekspresi Gen Skeletal Muscle NGT, IGT dan Diabetes Melitus Tipe-2 GSE18732)," 2020.

[9] F. Fallucchi and A. Cabroni, "Predicting Risk of Diabetes using a Model based on Multilayer Perceptron and Features Extraction," *Journal of Computer Science*, vol. 17, no. 9, pp. 748–761, 2021.

[10] T. Madhubala, R. Umagandhi, and P. Sathiamurthi, "Diabetes Prediction using Improved Artificial Neural Network using Multilayer Perceptron," *SSRG International Journal of Electrical and Electronics Engineering*, vol. 9, no. 12, pp. 167–179, 2022.

[11] Vinnarasi F.Sangeetha Francelin, Rose J.T.Anita, and Jesline, "An Automatic Classification of Diabetics with Multilayer Perceptron using Machine Learning," *International Journal of Recent Technology and Engineering*, 2020, [Online]. Available: https://api.semanticscholar.org/CorpusID:240926256

[12] R. Priyadarshini, R. K. Barik, N. Dash, B. K. Mishra, and R. Misra, "A hybrid GSA-K-mean classifier algorithm to predict diabetes mellitus," in *Cognitive Analytics: Concepts, Methodologies, Tools, and Applications*, IGI Global, 2020, pp. 589–603.

[13] S. K. Mohapatra, J. K. Swain, and M. N. Mohanty, "Detection of diabetes using multilayer perceptron," in *International Conference on Intelligent Computing and Applications: Proceedings of ICICA 2018*, Springer, 2019, pp. 109–116.

[14] N. M. Putry, "Komparasi algoritma knn dan naïve bayes untuk klasifikasi diagnosis penyakit diabetes mellitus," *Evolusi: Jurnal Sains Dan Manajemen*, vol. 10, no. 1, 2022.

[15] C. Kalpana and B. Booba, "Framework for Prediction of Diabetes using FireFly Swarm Intelligence Algorithm, Fuzzy C Mean and SVM Algorithm," in *2022 International Conference on Inventive Computation Technologies (ICICT)*, IEEE, 2022, pp. 1263–1269.

[16] O. Findik, "Investigation Effects of Selection Mechanisms for Gravitational Search Algorithm," *Journal of Computer and Communications*, vol. 2, no. 04, p. 117, 2014.

[17] N. Siddique and H. Adeli, "Gravitational search algorithm and its variants," *Intern J Pattern Recognit Artif Intell*, vol. 30, no. 08, p. 1639001, 2016.

[18] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Inf Sci (N Y)*, vol. 179, no. 13, pp. 2232–2248, 2009.

[19] T.-K. An and M.-H. Kim, "A new diverse AdaBoost classifier," in *2010 International conference on artificial intelligence and computational intelligence*, IEEE, 2010, pp. 359–363.

[20] D.-C. Feng *et al.*, "Machine learning-based compressive strength prediction for concrete: An adaptive boosting approach," *Constr Build Mater*, vol. 230, p. 117000, 2020.

[21] A. Khan *et al.*, "Lung cancer nodules detection via an adaptive boosting algorithm based on self-normalized multiview convolutional neural network," *J Oncol*, vol. 2022, no. 1, p. 5682451, 2022.

[22] Q. Zheng, C. Yu, J. Cao, Y. Xu, Q. Xing, and Y. Jin, "Advanced payment security system: xgboost, lightgbm and smote integrated," in *2024 IEEE International Conference on Metaverse Computing, Networking, and Applications (MetaCom)*, IEEE, 2024, pp. 336–342.

[23] N. V Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp. 321–357, 2002.