

**KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN
METODE *RANDOM FOREST* DAN *ADABOOST***

SKRIPSI

Oleh :
NAFIAH NUR MUTTAQIN
NIM. 210605110127



**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

**KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN
METODE *RANDOM FOREST* DAN *ADABOOST***

SKRIPSI

Diajukan kepada:
Universitas Islam Negeri Maulana Malik Ibrahim Malang
Untuk memenuhi Salah Satu Persyaratan dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)

Oleh:
NAFIAH NUR MUTTAQIN
NIM. 210605110127

**PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2024**

HALAMAN PERSETUJUAN

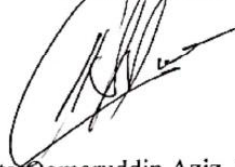
KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN
METODE *RANDOM FOREST* DAN *ADABOOST*

SKRIPSI

Oleh :
NAFIAH NUR MUTTAQIN
NIM. 210605110127

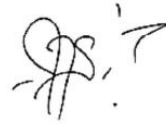
Telah Diperiksa dan Disetujui untuk Diuji:
Tanggal: 02 Desember 2024

Pembimbing I,



Okta Qomaruddin Aziz, M.Kom
NIP. 19911019 201903 1 013

Pembimbing II,




Khadijah Fahmi Hayati Holle, M.Kom
NIP. 19900626 202203 2 002

Mengetahui,

Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi

Universitas Islam Gama
Gegeri Maulana Malik Ibrahim Malang




Achrul Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN





KLASIFIKASI PENYAKIT DIABETES MENGGUNAKAN
METODE *RANDOM FOREST* DAN *ADABOOST*

SKRIPSI

Oleh :
NAFIAH NUR MUTTAQIN
NIM. 210605110127


Telah Dipertahankan di Depan Dewan Penguji Skripsi
dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 10 Desember 2024

Susunan Dewan Penguji

Ketua Penguji	: <u>Dr. Cahyo Crvsdian, M.CS</u> NIP. 19740424 200901 1 008	()
Anggota Penguji I	: <u>Roro Inda Melani, MT., M.Sc</u> NIP. 19780925 200501 2 008	()
Anggota Penguji II	: <u>Okta Qomaruddin Aziz, M.Kom</u> NIP. 19911019 201903 1 013	()
Anggota Penguji III	: <u>Khadijah Fahmi Hayati Holle, M.Kom</u> NIP. 19900626 202203 2 002	()

Mengetahui dan Mengesahkan,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang




Dr. Agus Kurniawan, M.MT., IPU
NIP. 19771020 200912 1 001

PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini:

Nama : Nafiah Nur Muttaqin
NIM : 210605110127
Fakultas / Program Studi : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : Klasifikasi Penyakit Diabetes Menggunakan Metode *Random Forest* dan *Adaboost*.

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar merupakan hasil karya saya sendiri, bukan merupakan pengambil alihan data, tulisan, atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan skripsi ini merupakan hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 10 Desember 2024
Yang membuat pernyataan,



Nafiah Nur Muttaqin
NIM.200605110111

MOTTO

“Lakukan selagi tidak merugikan orang lain”

“Apa yang melewatkanmu tidak akan pernah menjadi takdirmu, dan apa yang ditakdirkan untukmu tidak akan pernah melewatkanmu.”

(Umar bin Khattab)

HALAMAN PERSEMBAHAN

Puji Syukur atas kehadiran Allah Subhanahu wa ta'ala, karena berkat rahmat dan petunjuk-Nya, sehingga penulis dapat menyelesaikan skripsi ini dengan baik dan lancar. Shalawat serta salam kepada Rasulullah Shallallahu 'alaihi wasallam, yang telah membawa kita dari zaman jahiliyah menuju addinul Islam. Skripsi ini tidak akan selesai tanpa adanya kontribusi dan dukungan dari berbagai pihak. Oleh karena itu, penulis mempersembahkan skripsi ini kepada seluruh pihak yang telah berjasa dalam pengerjaan penelitian ini. Karya ini penulis persembahkan kepada:

1. Keluarga tercinta Bapak Slamet dan Ibu Yatimah selaku orang tua penulis, serta Nurul Hidayah Muttaqin S.pd dan Achmad Choirul Muttaqin S.pd selaku kakak penulis yang sangat penulis cintai dan tidak pernah berhenti dalam memberikan semangat, dukungan maupun doa kepada penulis hingga penulis dapat menyelesaikan skripsi ini dengan baik dan lancar.
2. Bapak Okta Qomaruddin Aziz, M.Kom selaku dosen pembimbing I serta Ibu Khadijah Fahmi Hayati Holle, M.Kom selaku dosen pembimbing II atas segala pelajaran yang diberikan kepada penulis yang banyak kekurangannya ini dan senantiasa membimbing, memberikan semangat, memberi arahan dan masukan, serta membantu penulis dalam mengerjakan dan menyelesaikan skripsi ini.

KATA PENGANTAR

Assalamu'alaikum Warahmatullahi Wabarakatuh

Alhamdulillah rabbilalamin, segala puji dan syukur senantiasa penulis panjatkan ke hadirat Allah subhanahu wa ta'ala atas berkat Rahmat, serta hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Klasifikasi Penyakit Diabetes Menggunakan Metode *Random Forest* dan *Adaboost*” dengan baik dan lancar. Skripsi ini diajukan untuk memenuhi salah satu syarat lulus sebagai sarjana komputer (S.Kom) pada Program Studi Teknik Informatika, Fakultas Sains dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang.

Penulisan skripsi ini tentunya tidak akan berjalan dengan baik tanpa adanya dukungan dan bantuan dari berbagai pihak, baik secara langsung maupun tidak langsung. Oleh karena itu, dengan penuh rasa hormat, penulis menyampaikan rasa terima kasih yang sebesar-besarnya kepada:

1. Prof. Dr. M. Zainuddin, M.A., selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim Malang.
2. Prof. Dr. Hj. Sri Harini, M.Si., selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Ir. Fachrul Kurniawan, M.MT, IPU., selaku Ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Okta Qomaruddin Aziz, M.Kom., selaku pembimbing utama yang dengan penuh kesabaran dan ketulusan hati memberikan bimbingan, arahan, serta dorongan dalam setiap tahap penyusunan skripsi ini.

5. Khadijah Fahmi Hayati Holle, M.Kom., selaku dosen pembimbing kedua penulis yang selalu memberikan bimbingan hingga terselesaikannya skripsi ini.
6. Dr. Cahyo Crysdian, M.CS, dan Roro Inda Melani, MT., M.Sc selaku penguji utama dan Roro Inda Melani, M, T., M. Sc., selaku penguji kedua yang telah berkenan menguji serta memberikan masukan yang membangun sehingga skripsi ini dapat terselesaikan dengan baik.
7. Nia Faricha S, Si., selaku admin Program Studi Teknik Informatika yang selalu sabar memberikan informasi, membantu, dan memberikan arahan selama perkuliahan dan proses penulisan skripsi ini.
8. Segenap dosen, laboran, dan jajaran staff Program Studi Teknik Informatika yang telah memberikan ilmu, pengetahuan, dan dukungan selama penulis menjalani studi hingga selesainya skripsi ini.
9. Kedua orang tua tercinta, Bapak Slamet dan Ibu Yatimah yang selalu menjadi sumber kekuatan bagi penulis. Terima kasih karena selalu mengusahakan yang terbaik. Semoga Allah senantiasa memberikan kesehatan dan perlindungan, sehingga dapat selalu berada disetiap perjalanan dan pencapaian penulis.
10. Kepada kakak-kakak tercinta Nurul Hidayah Muttaqin S.Pd dan Achmad Choirul Muttaqin S.Pd yang telah menjadi sumber inspirasi, motivasi, dan dukungan selama perjalanan ini. Terima kasih atas semangat, nasihat, serta perhatian yang tak henti diberikan kepada penulis. Dan kepada seluruh

keluarga besar yang tiada henti memberikan doa dan dukungan sehingga penulis mampu menyelesaikan skripsi ini.

11. Teman dekat penulis, Firda Arinda Eka Putri yang sudah setia menemani, memberikan semangat serta segala bantuan dari awal perkuliahan hingga saat ini. Dukunganmu, perhatianmu selalu menjadi penyemangat bagi penulis, terutama saat-saat sulit. Terima kasih atas kehadiranmu yang tidak hanya menjadi teman berbagi cerita, tetapi juga selalu ada untuk mendengarkan dan memberikan semangat saat dibutuhkan.
12. Teman dekat penulis “Mencoba Kuat Sampai Tamat”, Firda, Wanda dan Arneiza, kehadiran kalian memberikan warna dalam perjalanan ini, dengan segala bantuan, perhatian, dan dukungan yang tak pernah kalian ragu untuk berikan. Bersama kalian, penulis merasa tidak pernah sendirian, bahkan dalam situasi yang sulit. Terima kasih telah menjadi teman yang selalu ada dan penyemangat yang luar biasa.
13. Himatif Encoder, Divisi Seni dan Olahraga '22, yang telah memberikan semangat, dukungan, dan menjadi tempat penulis untuk menjadikan motivasi dan panutan selama masa perkuliahan.
14. Teman-teman “Kelas E”, yang telah dikenal sejak awal semester dan menjadi bagian penting dalam perjalanan penulis selama ini. Dengan rutinitas bermain badminton yang kami lakukan bersama, penulis dapat meluapkan berbagai perasaan dan menikmati momen kebersamaan. Terima kasih atas semangat, dukungan, dan kebersamaan yang telah kalian berikan, serta menjadi tempat penulis mengembangkan hobi badminton.

Kenangan bersama Kelas E akan selalu menjadi bagian yang berharga dalam perjalanan hidup ini.

15. Seluruh warga Teknik Informatika khususnya angkatan 2021 “Aster” yang telah memberikan kehangatan, motivasi, semangat, dan dukungan kepada penulis.
16. Kepada orang istimewa, Ahmad Mubarak Fathillah yang telah menjadi pendukung terbaik dalam perjalanan panjang ini. Terima kasih atas doa, kesabaran, mendengarkan keluh kesah, memberikan semangat, dan senantiasa menguatkan penulis, bahkan di saat-saat terberat sekalipun menjadi salah satu sumber kekuatan bagi penulis.
17. Seluruh pihak yang telah terlibat, baik secara langsung maupun tidak langsung dari awal perkuliahan hingga akhir penulisan skripsi ini.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kata sempurna. Maka dari itu penulis menerima saran, kritik dan masukan yang bersifat membangun sehingga skripsi ini dapat lebih dikembangkan. Penulis berharap semoga skripsi ini dapat memberikan manfaat untuk kedepannya.

Malang, 7 Desember 2024

Penulis

DAFTAR ISI

HALAMAN JUDUL	ii
HALAMAN PERSETUJUAN	iii
HALAMAN PENGESAHAN	iv
PERNYATAAN KEASLIAN TULISAN	iv
MOTTO.	vi
HALAMAN PERSEMBAHAN	vii
KATA PENGANTAR	viii
DAFTAR ISI	xii
DAFTAR GAMBAR	xv
DAFTAR TABEL	xvi
ABSTRAK	xviii
ABSTRACT	xix
مستخلص البحث	xx
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Pernyataan Masalah.....	4
1.3 Batasan Masalah	4
1.4 Tujuan Penelitian	5
1.5 Manfaat Penelitian.....	5
BAB II STUDI PUSTAKA	6
2.1 Klasifikasi Penyakit.....	6
2.2 Diabetes	9
2.3 <i>Machine Learning</i>	10
2.4 Balancing Data	12
2.5 Random Forest.....	13
2.6 <i>Adaboost</i>	15
2.6 <i>Feature Importance</i>	15
2.7 Faktor Usia terhadap Klasifikasi Diabetes.....	16
BAB III DESAIN DAN IMPLEMENTASI	18
3.1 Tahapan Penelitian.....	18
3.2 Pengumpulan Data.....	18
3.3 Desain Sistem	20
3.4 Preprocessing.....	22
3.4.1 Interpolasi	22
3.4.2 Tranformasi Data	22
3.4.3 Normalisasi Data.....	23
3.4.4 Synthetic Minority Oversampling Technique (<i>SMOTE</i>)	24
3.4 <i>Split Data</i>	25

3.5 Implementasi Model	25
3.5.1 <i>Random Forest</i>	25
3.5.2 <i>Adaboost</i>	28
3.6 Evaluasi Model	30
3.7 Skenario Pengujian	32
BAB IV UJI COBA DAN PEMBAHASAN.....	35
4.1 Hasil Training	35
4.2 Hasil Uji Coba Model <i>Random Forest</i>	38
4.2.1 Model 1	38
4.2.2 Model 2	38
4.2.3 Model 3	39
4.2.4 Model 4	39
4.2.5 Model 5	40
4.2.6 Model 6	41
4.2.7 Model 7	41
4.2.8 Model 8	42
4.2.9 Model 9	42
4.2.10 Model 10	43
4.2.11 Model 11	43
4.2.12 Model 12	44
4.2.13 Model 13	44
4.2.14 Model 14	45
4.2.15 Model 15	46
4.2.16 Model 16	46
4.2.17 Model 17	47
4.2.18 Model 18	47
4.2.19 Model 19	48
4.2.20 Model 20	48
4.2.21 Model 21	49
4.2.22 Model 22	50
4.2.23 Model 23	50
4.2.24 Model 24	51
4.3 Hasil Uji Coba Algoritma <i>Adaboost</i>	51
4.3.1 Model 25	52
4.3.2 Model 26	52
4.3.3 Model 27	53
4.3.4 Model 28	53

4.3.5 Model 29	54
4.3.6 Model 30	54
4.3.7 Model 31	55
4.3.8 Model 32	55
4.3.9 Model 33	56
4.3.10 Model 34	57
4.3.11 Model 35	57
4.3.12 Model 36	58
4.4 Pembahasan	58
4.4.1 Analisa Hasil Uji Coba Algoritma <i>Random Forest</i>	58
4.4.2 Analisa Hasil Uji Coba Algoritma <i>Adaboost</i>	62
4.4.3 Perbandingan <i>Random Forest</i> dan <i>Adaboost</i>	66
4.4.3 Analisa <i>Feature Importance</i>	68
BAB V KESIMPULAN DAN SARAN	73
5.1 Kesimpulan	73
5.2 Saran	74
DAFTAR PUSTAKA	

DAFTAR GAMBAR

Gambar 2. 1 Desain model <i>Random Forest</i>	13
Gambar 3. 1 Tahapan Penelitian	18
Gambar 3. 2 Desain sistem	21
Gambar 3. 3 Desain Model <i>Random Forest</i>	26
Gambar 3. 4 Desain Model <i>Adaboost</i>	28
Gambar 3. 5 Skenario Pengujian.....	32
Gambar 4. 1 Grafik Hasil Uji Coba <i>Random Forest</i>	59
Gambar 4. 2 Grafik Hasil Uji Coba <i>Adaboost</i>	63
Gambar 4. 3 3 model terbaik <i>Random Forest</i>	66
Gambar 4. 4 3 model terbaik <i>Adaboost</i>	67
Gambar 4. 5 Hasil <i>Feature Importance</i>	68

DAFTAR TABEL

Tabel 2. 1 Penelitian Terdahulu	8
Tabel 3. 1 Penjelasan Fitur.....	19
Tabel 3. 2 Contoh Dataset Penyakit Diabetes	20
Tabel 3. 3 Confusion matrix.....	31
Tabel 3. 4 Skenario Pengujian	32
Tabel 4. 3 Confusion matrix Uji Coba Ke-3	39
Tabel 4. 4 Confusion matrix Uji Coba Ke-4.....	40
Tabel 4. 5. Confusion matrix Uji Coba Ke-5	40
Tabel 4. 6 Confusion matrix Uji Coba Ke-6.....	41
Tabel 4. 7 Confusion matrix Uji Coba Ke-7	41
Tabel 4. 8 Confusion matrix Uji Coba Ke-8.....	42
Tabel 4. 9. Confusion matrix Uji Coba Ke-9.....	42
Tabel 4. 10 Confusion matrix Uji Coba Ke-10	43
Tabel 4. 11 Confusion matrix Uji Coba Ke-11	44
Tabel 4. 12 Confusion matrix Uji Coba Ke-12.....	44
Tabel 4. 13 Confusion matrix Uji Coba Ke-13.....	45
Tabel 4. 14. Confusion matrix Uji Coba Ke-14	45
Tabel 4. 15. Confusion matrix Uji Coba Ke-15	46
Tabel 4. 16 Confusion matrix Uji Coba Ke-16.....	46
Tabel 4. 17 Confusion matrix Uji Coba Ke-17	47
Tabel 4. 18 Confusion matrix Uji Coba Ke-18.....	47
Tabel 4. 19 Confusion matrix Uji Coba Ke-19	48
Tabel 4. 20 Confusion matrix Uji Coba Ke-20.....	49
Tabel 4. 21 Confusion matrix Uji Coba Ke-21	49
Tabel 4. 22 Confusion matrix Uji Coba Ke-22.....	50
Tabel 4. 23 Confusion matrix Uji Coba Ke-23.....	50
Tabel 4. 24 Confusion matrix Uji Coba Ke-24.....	51
Tabel 4. 25 Confusion matrix Uji Coba Ke-25.....	52
Tabel 4. 26 Confusion matrix Uji Coba Ke-26.....	52
Tabel 4. 27 Confusion matrix Uji Coba Ke-27	53
Tabel 4. 28 Confusion matrix Uji Coba Ke-28.....	54
Tabel 4. 29 Confusion matrix Uji Coba Ke-29	54
Tabel 4. 30 Confusion matrix Uji Coba Ke-30.....	55
Tabel 4. 31 Confusion matrix Uji Coba Ke-31	55
Tabel 4. 32 Confusion matrix Uji Coba Ke-32.....	56
Tabel 4. 33 Confusion matrix Uji Coba Ke-33	56
Tabel 4. 34. Confusion matrix Uji Coba Ke-34.....	57
Tabel 4. 35. Confusion matrix Uji Coba Ke-35	57
Tabel 4. 36 Confusion matrix Uji Coba Ke-36.....	58
Tabel 4. 37 Analisa pengaruh Balancing pada metode Random Forest	60

Tabel 4. 38 Analisa pengaruh Normalisasi pada metode Random Forest	60
Tabel 4. 39 Analisa pengaruh n_estimator pada metode Random Forest.....	61
Tabel 4. 40 Analisa pengaruh max_depth pada metode Random Forest.....	62
Tabel 4. 41 Analisa pengaruh Normalisasi pada metode Adaboost	63
Tabel 4. 42 Analisa pengaruh Balancing pada metode Adaboost.....	64
Tabel 4. 43 Analisa pengaruh n_estimator pada metode Adaboost.....	65

ABSTRAK

Muttaqin, Nafiah Nur. 2024. **Klasifikasi Penyakit Diabetes Menggunakan Metode *Random Forest* dan *Adaboost***. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Okta Qomaruddin Aziz, M.Kom, (II) Khadijah Fahmi Hayati Holle, M.Kom.

Kata Kunci: *Adaboost*, Diabetes, Klasifikasi, *Random Forest*

Diabetes termasuk penyakit kronis yang ditandai oleh kadar gula darah tinggi akibat gangguan produksi atau fungsi insulin, yang dapat menyebabkan komplikasi serius jika tidak ditangani. Klasifikasi diabetes diperlukan untuk membantu deteksi dini dan pengambilan keputusan klinis yang lebih tepat guna mencegah komplikasi jangka panjang. Penelitian ini membandingkan metode *Random Forest* dan *Adaboost* untuk meningkatkan akurasi prediksi diabetes. Dataset yang digunakan adalah data diabetes penduduk Afrika-Amerika pedesaan, yang diproses melalui interpolasi, normalisasi, dan *Balancing* menggunakan *SMOTE*. Hasil penelitian menunjukkan bahwa *Random Forest* memiliki performa lebih baik dengan akurasi hingga 98%, unggul dalam *precision*, *recall*, dan *F1-score* dibandingkan *Adaboost* yang mencapai akurasi 96%. Analisis *feature importance* mengidentifikasi kadar glukosa sebagai faktor paling signifikan, diikuti oleh fitur usia. Penelitian ini mendukung kesimpulan bahwa metode *Random Forest* adalah metode yang lebih efektif untuk prediksi diabetes dan pengambilan keputusan yang lebih akurat.

ABSTRACT

Muttaqin, Nafiah Nur. 2024. **Classification of Diabetes Using *Random Forest* and *Adaboost* Methods**. Thesis. Department of Informatics Engineering, Faculty of Science and Technology, Maulana Malik Ibrahim State Islamic University Malang. Supervisors : (I) Okta Qomaruddin Aziz, M.Kom, (II) Khadijah Fahmi Hayati Holle, M.Kom.

Key words: *Classification, Diabetes, Random Forest, Adaboost.*

Diabetes is a chronic disease characterized by high blood sugar levels due to impaired insulin production or function, which can lead to serious complications if left untreated. Diabetes classification is needed to aid early detection and more informed clinical decision-making to prevent long-term complications. This research compares Random Forest and Adaboost methods to improve the accuracy of diabetes prediction. The dataset used is diabetes data of rural African-American population, which is processed through interpolation, normalization, and balancing using SMOTE. The results showed that Random Forest performed better with up to 98% accuracy, excelling in precision, recall, and F1-score compared to Adaboost which achieved 96% accuracy. Feature importance analysis identified glucose level as the most significant factor, followed by age. This research supports the conclusion that the Random Forest method is a more effective method for diabetes prediction and more accurate decision making.

مستخلص البحث

و تعزيز التكيف (Random Forest) متقين، نافعة نور. 2024. تصنيف مرض السكري باستخدام طريقة غابة عشوائية البحث الجامعي. قسم الهندسة المعلوماتية، كلية العلوم والتكنولوجيا بجامعة مولانا مالك إبراهيم (Adaboost) الإسلامية الحكومية مالانج. المشرف الأول: أوكتا قمر الدين عزيز، الماجستير. المشرف الثاني: خديجة فهمي حياتي. هولي، الماجستير.

الكلمات الرئيسية: تعزيز تكيف، مرض سكري، تصنيف، غابة عشوائية

مرض السكري هو مرض مزمن يتميز بارتفاع مستويات السكر في الدم بسبب ضعف إنتاج الأنسولين أو وظيفته، والذي يمكن أن يؤدي إلى مضاعفات خطيرة إذا ترك دون علاج. هناك حاجة إلى تصنيف مرض السكري للمساعدة في الكشف المبكر واتخاذ قرارات سريرية أكثر ملاءمة لمنع المضاعفات طويلة الأمد. قارن هذا البحث بين طريقة غابة عشوائية و تعزيز التكيف لتحسين دقة التنبؤ بمرض السكري. كانت مجموعة البيانات المستخدمة هي بيانات مرض السكري للسكان الريفيين الأمريكيين من أصل أفريقي، والتي تمت معالجتها من خلال الاستيفاء والتطبيع والموازنة باستخدام تقنية الإفراط في أخذ عينات الأقليات الاصطناعية أظهرت النتائج أن طريقة غابة عشوائية تتمتع بأداء أفضل بدقة تصل إلى 98٪، متفوقة في الدقة والاستدعاء ودرجة (SMOTE) ف1 مقارنة بطريقة تعزيز التكيف التي تحقق دقة بنسبة 96٪. حدد تحليل ميزة الأهمية مستويات الجلوكوز على أنها العامل الأكثر أهمية، تليها ميزة العمر. يدعم هذا البحث الاستنتاج القائل بأن طريقة الغابة العشوائية هي طريقة أكثر فاعلية للتنبؤ بمرض السكري واتخاذ قرارات أكثر دقة.

BAB I

PENDAHULUAN

1.1 Latar Belakang

Diabetes adalah salah satu dari penyakit kronis yang paling umum di dunia, mempengaruhi jutaan orang tanpa memandang usia atau jenis kelamin menurut (Arania dkk. 2021). Penyakit ini ditandai oleh kadar gula darah yang tinggi akibat masalah dengan produksi atau fungsi insulin, yang menyebabkan gejala seperti kelelahan, penurunan berat badan, dan penglihatan kabur menurut (T.Eltrikanawati and Fedillah Nurhafifah 2023). Diabetes yang tidak terdiagnosis dan tidak ditangani dengan baik dapat mengakibatkan penurunan kualitas hidup, peningkatan risiko komplikasi serius, seperti penyakit jantung, kerusakan saraf, dan bahkan kematian (Manurung and Darungan 2021).

Setiap penyakit pasti ada obatnya, sebagaimana telah dijelaskan dalam Asy-Syu'ara ayat 80 yang berbunyi :

وَإِذَا مَرَضْتُ فَهُوَ يَشْفِينِي ﴿٨٠﴾

“Apabila aku sakit, Dialah yang menyembuhkanku.” (QS. Asy Syu'ara: 80).

Dari ayat di atas dijelaskan bahwa Allah yang akan menyembuhkan manusia apabila sakit. Allah yang berkuasa untuk menyembuhkan penyakit apa saja yang diderita oleh seseorang. Namun demikian, manusia sebagai ciptaan-Nya juga harus berusaha dan mencari cara untuk sembuh. Ayat ini mengajarkan bahwa meskipun kita percaya akan kekuasaan Allah dalam menyembuhkan, kita tetap memiliki tanggung jawab untuk menjaga kesehatan dan berusaha mencari pengobatan yang

tepat. Dengan demikian, kesadaran dan tindakan mencegah penyakit diabetes menjadi bentuk ikhtiar kita dalam menjaga amanah kesehatan yang diberikan oleh Allah SWT (Wiji dkk., 2023).

Dalam beberapa tahun terakhir, perkembangan teknologi informasi dan komputasi telah membuka peluang baru untuk meningkatkan diagnosis dan manajemen penyakit, termasuk diabetes penjelasan menurut (Gunawan dkk. 2022). *Machine Learning*, telah menunjukkan potensi besar dalam mengklasifikasikan berbagai penyakit berdasarkan data medis yang kompleks dan beragam (Cahyanti dkk., 2023). *Random Forest* dan *Adaboost* merupakan dua algoritma *Machine Learning* yang telah banyak digunakan dan diakui dalam bidang ini karena keunggulannya dalam mengolah data yang kompleks dan memberikan hasil yang akurat (Marsuhandi dkk., 2020).

Random Forest, sebuah ensemble learning method, yang bekerja dengan membuat beberapa pohon keputusan selama *training* dan mengoutputkan kelas yang merupakan model dari kelas-kelas output (klasifikasi) atau mean prediction (regresi) dari masing-masing pohon individu (Kari dkk., 2021). Metode ini dikenal tangguh terhadap *overfitting* dan efektif dalam menangani dataset yang memiliki banyak variable (Syukron & Subekti, 2018).

Adaboost (Adaptive Boosting) adalah teknik boosting yang bekerja dengan menggabungkan beberapa model sederhana, biasanya disebut *Weak Learners*, untuk membentuk model yang kuat (Ding dkk., 2022). Metode ini secara iteratif menyesuaikan bobot model untuk memperbaiki kesalahan prediksi dari iterasi sebelumnya, sehingga meningkatkan akurasi secara signifikan (Jasman dkk., 2022).

Meskipun kedua metode ini memiliki kelebihan masing-masing, studi komparatif yang mendalam untuk menentukan metode mana yang lebih efektif dalam klasifikasi penyakit diabetes masih terbatas (Candra Permana & Dewi Patwari, 2021). Oleh karena itu, penelitian ini berfokus pada perbandingan hasil dari penerapan metode *Random Forest* dan *Adaboost* dalam mendeteksi penyakit diabetes. Dengan demikian, penelitian ini diharapkan dapat memberikan wawasan dan pengetahuan yang lebih baik tentang kelebihan dan kekurangan masing-masing metode.

Usia telah diidentifikasi sebagai salah satu penyebab utama yang mempengaruhi risiko pengembangan diabetes mellitus tipe 2. Penelitian oleh Susilawati dan Rista Rahmawati pada tahun 2021 di Puskesmas Tugu, Kecamatan Cimanggis, Depok, mengungkapkan bahwa usia memiliki pengaruh signifikan terhadap kejadian diabetes melitus tipe 2. Mereka menemukan bahwa seiring dengan bertambahnya usia, resiko seseorang untuk mengembangkan diabetes meningkat, terutama pada individu yang memiliki faktor risiko lain seperti hipertensi dan riwayat keluarga. Temuan ini sejalan dengan hasil penelitian (Arania dkk., 2021), yang melihatkan hasil prevalensi diabetes yang lebih banyak pada kelompok usia yang lebih tua, akibat perubahan metabolisme serta penurunan fungsi insulin yang terjadi seiring bertambahnya usia. Penelitian-penelitian ini menunjukkan bahwa perhatian terhadap faktor usia sangat penting dalam upaya pencegahan dan pengelolaan diabetes mellitus tipe 2.

Penelitian ini penting dilakukan mengingat kebutuhan yang mendesak akan metode diagnostik yang lebih cepat, akurat, dan efisien dalam manajemen penyakit

diabetes. Dengan demikian, hasil dari penelitian ini diharapkan tidak hanya memperkaya literatur ilmiah tetapi juga memberikan manfaat praktis bagi tenaga medis dalam meningkatkan kualitas diagnosis pasien diabetes.

1.2 Pernyataan Masalah

1. Bagaimana hasil perbandingan performa metode *Random Forest* dan *Adaboost* dalam klasifikasi penyakit diabetes berdasarkan hasil akurasi?
2. Bagaimana pengaruh normalisasi dan *Balancing* data pada metode *Random Forest* dan *Adaboost* dalam klasifikasi penyakit diabetes?
3. Apakah fitur usia yang dinyatakan sebagai fitur berpengaruh dalam penelitian oleh Susilawati dan Rista Rahmawati juga terbukti sebagai fitur paling berpengaruh dalam penelitian ini?

1.3 Batasan Masalah

1. Data yang digunakan adalah Diabetes dataset. Data yang diambil dari data penduduk Afrika-Amerika yang merupakan data sekunder didapat dari website Kaggle.com dan memiliki 19 atribut serta 403 *record* data.
2. Menggunakan atribut chol, stab.glu, hdl, ratio, glyhb, location, age, gender, height, weight, frame, bp.1s, bp.1d,bp.2s,bp.2d, waist, hip, time.ppn pada data Diabetes Dataset.
3. Penelitian ini meneliti pengaruh fitur usia berdasarkan hasil penelitian yang dilakukan oleh Susilawati dan Rista Rahmawati dengan judul “Hubungan Usia, Jenis Kelamin, dan Hipertensi dengan Kejadian Diabetes Mellitus Tipe 2 di Puskesmas Tugu Kecamatan Cimanggis Kota Depok”.

1.4 Tujuan Penelitian

1. Membandingkan performa metode *Random Forest* dan *Adaboost* dalam klasifikasi penyakit diabetes berdasarkan hasil akurasi.
2. Mengukur pengaruh normalisasi dan *Balancing* pada performa metode *Random Forest* dan *Adaboost* dalam klasifikasi penyakit diabetes berdasarkan hasil akurasi
3. Membuktikan apakah fitur usia yang dinyatakan signifikan dalam penelitian oleh Susilawati dan Rista Rahmawati juga terbukti signifikan memengaruhi dalam penelitian ini.

1.5 Manfaat Penelitian

1. Memberikan manfaat untuk klasifikasi diabetes.
2. Memberikan wawasan tentang metode *Random Forest Classifier*.
3. Memberikan wawasan tentang metode *Adaboost*.
4. Hasil dari penelitian ini dapat digunakan sebagai landasan pada penelitian selanjutnya.

BAB II

STUDI PUSTAKA

2.1 Klasifikasi Penyakit

Dalam penelitian yang dilakukan oleh (Depari dkk. 2022) tentang penerapan metode *Random Forest*, *Decision Tree*, & *Naïve Bayes* untuk mendeteksi penyakit Jantung. Dalam penelitian yang bertujuan untuk membandingkan hasil performa dari metode *Random Forest*, *Decision Tree*, & *Naïve Bayes* dengan menggunakan kumpulan data pasien penyakit jantung “Personal Key Indicators of Heart Disease”. Penelitian ini menunjukkan bahwa hasil akurasi dari metode *Random Forest* sebesar 75%, *Decision Tree* sebesar 71%, dan *Naïve Bayes* sebesar 72%. Dari hasil penelitian ini terlihat bahwa performa metode *Random Forest* merupakan metode yang paling baik diantara metode yang lainnya.

Penelitian yang telah dilakukan oleh (Sumwiza dkk. 2023) yang berjudul "Enhanced Cardiovascular Disease Prediction Model Using *Random Forest* Algorithm". Penelitian ini menggunakan metode ensemble dengan algoritma *Random Forest* untuk meningkatkan akurasi dalam mengklasifikasikan penyakit kardiovaskular. Data yang digunakan adalah Heart Disease Dataset, yang berisi 1025 entri dengan 14 atribut. Selama tahap *Preprocessing*, outlier dihapus untuk meningkatkan representasi fitur yang relevan dengan penyakit tersebut. Penelitian ini mencapai akurasi 99%. Perbedaan utama terletak pada data yang digunakan, yaitu menggunakan Heart Disease Dataset, sementara penelitian ini menggunakan Heart Attack Analysis & Prediction.

Penelitian oleh (Erdiansyah dkk., 2022) yang berjudul “Komparasi Metode *K-Nearest Neighbor* dan *Random Forest* Dalam Prediksi Akurasi Klasifikasi Pengobatan Penyakit Kutil”. Penelitian ini membahas tentang penerapan Metode *K-Nearest Neighbor* dan *Random Forest* untuk melihat tingkat akurasi pada prediksi keberhasilan dari pengobatan penyakit kutil. Dalam penelitian menggunakan data dari *UCI Machine Learning Repository* dengan jumlah sebanyak 90 record data, 7 atribut dan 1 kelas atribut. Hasil dari pengujian menggunakan *K-Nearest Neighbor* memperoleh akurasi 76.78 %, kemudian metode *Random Forest* memperoleh tingkat akurasi 86.56 %. Dari pengujian tersebut menunjukkan metode *Random Forest* memperoleh hasil yang lebih unggul dibandingkan menggunakan *K-Nearest Neighbor*.

Penelitian yang diteliti oleh (Abdurrahman, 2022) tentang diabetes dengan judul “Klasifikasi Penyakit Diabetes Melitus Menggunakan *Adaboost Classifier*”. *Adaboost (Adaptive Boosting)*, yang berfungsi untuk membangun "strong Classifier" dengan menggabungkan beberapa "weak Classifier". Dataset yang digunakan dalam penelitian berasal dari *UCI Machine Learning Repository*, dengan 768 record, 8 variabel prediktor, dan 1 variabel target. Dari hasil penelitian, *Adaboost Classifier* menghasilkan akurasi 80.09% setelah imputasi mean dan 76.19% setelah imputasi mean. Namun, imputasi modus tidak mengubah hasil dari dataset asli, dikarenakan *Adaboost* sangat sensitif jika terdapat missing values yang dikenali sebagai modus dan digantikan dengan NaN oleh Python (Abdurrahman, 2022).

Penelitian yang dilakukan oleh (Mortara dkk., 2023) dengan judul “Perbandingan Algoritma C4.5 dan Adaptive Boosting dalam Klasifikasi Penyakit Alzheimer” membandingkan performa algoritma C4.5 dan Adaptive Boosting (*Adaboost*) menggunakan data Alzheimer dari Kaggle, yang terdiri dari 373 sampel data dan terdiri dari 15 atribut. Hasil penelitian menunjukkan bahwa *Adaboost* memiliki akurasi lebih baik dibandingkan C4.5. Pada metode percentage split (80% data latih, 20% data uji), *Adaboost* memperoleh akurasi 89%, sedangkan C4.5 88%. Pada metode k-fold cross-validation ($k = 10$), *Adaboost* mencapai akurasi 91,15%, sedangkan C4.5 hanya 89,27%. Algoritma *Adaboost* terbukti lebih unggul, khususnya pada metode k-fold cross-validation.

Tabel 2. 1 Klasifikasi Penyakit

No.	Judul Penelitian	Peneliti	Metode	Hasil Penelitian
1.	Penerapan Metode <i>Random Forest</i> , <i>Decision Tree</i> , dan Naïve Bayes untuk Mendeteksi Penyakit Jantung	Depari dkk. (2022)	<i>Random Forest</i> , <i>Decision Tree</i> , Naïve Bayes	<i>Random Forest</i> (75%), <i>Decision Tree</i> (71%), Naïve Bayes (72%) – <i>Random Forest</i> terbaik
2.	Enhanced Cardiovascular Disease Prediction Model Using <i>Random Forest</i> Algorithm	Sumwiza dkk. (2023)	<i>Random Forest</i>	Akurasi 99% dengan <i>Preprocessing</i> dan Heart Disease Dataset
3.	Komparasi Metode K-Nearest Neighbor dan <i>Random Forest</i> Dalam Prediksi Akurasi Klasifikasi Pengobatan Penyakit Kutil	"Erdiansyah, Lubis, Erwansyah (2022)"	"K-Nearest Neighbor, <i>Random Forest</i> "	" <i>Random Forest</i> (86.56%), K-Nearest Neighbor (76.78%) – <i>Random Forest</i> lebih baik"
4.	Klasifikasi Penyakit Diabetes Melitus Menggunakan <i>Adaboost</i> Classifier	Abdurrahman (2022)	<i>Adaboost</i> Classifier	Akurasi 80.09% setelah imputasi mean
5.	Perbandingan Algoritma C4.5 dan Adaptive Boosting dalam Klasifikasi Penyakit Alzheimer	Mortara dkk. (2023)	C4.5, Adaptive Boosting (<i>Adaboost</i>)	<i>Adaboost</i> memperoleh akurasi 91,15%, sedangkan C4.5 memperoleh akurasi 89,27%.

Tabel 2.1 menyajikan ringkasan penelitian terdahulu yang menggunakan berbagai metode dalam klasifikasi penyakit. Tabel ini mencakup judul penelitian, peneliti, metode yang digunakan, dan hasil penelitian masing-masing. Dalam Tabel ini, ditampilkan beberapa penelitian yang mengaplikasikan algoritma *Random Forest* dan *Adaboost* dalam mendeteksi dan mengklasifikasikan berbagai penyakit, termasuk penyakit jantung, kutil, alzheimer, dan diabetes melitus. Informasi yang terdapat dalam Tabel ini memberikan gambaran mengenai efektivitas metode yang digunakan dalam konteks kesehatan, serta hasil akurasi yang dicapai oleh masing-masing penelitian.

2.2 Diabetes

Diabetes Mellitus merupakan kondisi medis kronis yang dideteksi dari tingginya kadar gula dalam darah menurut (Dewi dkk. 2023). Kondisi ini terjadi ketika tubuh seseorang tidak mampu memproduksi cukup insulin dan tidak dapat menggunakannya dengan efektif. Insulin dan hormon dihasilkan oleh pankreas, berperan penting dalam membantu sel-sel tubuh menyerap glukosa dari darah untuk diubah menjadi energi menurut (Gita and Utomo 2022). Tanpa insulin yang cukup, glukosa menumpuk dalam darah, yang pada akhirnya bisa menyebabkan berbagai masalah kesehatan serius. Diabetes tidak hanya berdampak pada metabolisme tubuh tetapi juga memiliki efek negatif pada kesehatan jantung, saraf, dan organ lainnya (P. C. Simarmata dkk., 2021).

Jumlah penderita penyakit diabetes terus mengalami peningkatan di seluruh dunia, dengan jutaan orang terdiagnosis setiap tahunnya. Menurut laporan dari Organisasi Kesehatan Dunia (WHO), tingkat kejadian diabetes meningkat di

hampir semua negara, terutama di negara berkembang (Lovic dkk., 2020). Dampak kesehatan dari diabetes sangat signifikan, termasuk meningkatnya risiko penyakit jantung, stroke, kerusakan saraf, gagal ginjal, dan gangguan penglihatan. Selain komplikasi-komplikasi tersebut, diabetes juga bisa mempengaruhi kualitas hidup penderita dalam jangka panjang (Ferawati & Hadi Sulisty, 2020).

Beberapa faktor risiko utama dapat berkontribusi terhadap perkembangan diabetes. Faktor genetik merupakan salah satu yang paling penting, di mana riwayat keluarga dengan diabetes bisa meningkatkan risiko seseorang. Selain faktor genetik, gaya hidup seperti pola makan yang kurang baik, kurang gerak, dan kelebihan berat badan juga sangat mempengaruhi. Faktor lain seperti usia, etnisitas tertentu, dan kondisi medis lainnya seperti tekanan darah tinggi dan kadar kolesterol yang tinggi juga bisa meningkatkan risiko (Nasution dkk., 2021).

Pencegahan diabetes bisa dilakukan dengan menjalani gaya hidup yang lebih sehat, seperti makan makanan yang seimbang, berolahraga setiap hari, dan menjaga berat badan yang ideal. Bagi mereka yang sudah terdiagnosis diabetes, pengelolaan yang baik sangat penting untuk mencegah komplikasi lebih lanjut. Pengelolaan diabetes meliputi pemantauan rutin kadar glukosa darah, penggunaan obat atau insulin sesuai petunjuk dokter, dan edukasi mengenai cara mengelola diabetes sendiri (Sandrawati & Somadayo, 2023).

2.3 *Machine Learning*

Machine Learning adalah merupakan salah satu komponen utama dalam kecerdasan buatan yang memungkinkan sistem untuk belajar dari data dan

mengambil keputusan atau membuat prediksi tanpa perlu diprogram secara eksplisit. Esensi dari pembelajaran mesin terletak pada kemampuannya dalam mengenali pola serta hubungan dalam data, yang selanjutnya dimanfaatkan untuk menghasilkan keputusan yang lebih akurat di masa mendatang. Teknologi ini sangat berguna di berbagai bidang, termasuk medis, di mana ia dapat mengelola data kompleks dan besar untuk menciptakan model prediksi yang akurat (Diana dkk. 2023).

Teknik *Machine Learning* sangat berharga dalam menemukan faktor risiko yang mungkin tidak jelas dengan analisis data konvensional. Dengan kemampuannya untuk memproses data yang beragam dan rumit, *Machine Learning* dapat mengidentifikasi pola yang tidak terlihat dengan metode tradisional, membantu dalam pencegahan dan manajemen penyakit. Sebagai contoh, teknik ini bisa mengungkap faktor risiko untuk penyakit kronis berdasarkan data medis yang luas, mendukung pengembangan strategi pencegahan yang lebih tepat (Telaumbanua dkk., 2020).

Di dunia medis, berbagai algoritma *Machine Learning* digunakan, termasuk regresi, *Decision Tree*, dan *neural network*. Masing-masing algoritma ini memiliki kekuatan dan kekurangan, sehingga pemilihan yang tepat bergantung pada jenis data dan tujuan analisis. Jaringan saraf tiruan, misalnya, sangat efektif untuk data gambar medis, sedangkan regresi lebih cocok untuk data numerik dan prediksi (Jatnika Fahmi Idris dkk., 2024).

2.4 Balancing Data

Ketidakseimbangan data (*Class imbalance*) adalah tantangan umum dalam melakukan analisis data, terutama pada klasifikasi. Ketika kelas minoritas memiliki jumlah sampel yang jauh lebih sedikit dibandingkan dengan kelas mayoritas, model *machine learning* cenderung bias di kelas mayoritas. Oleh karena itu, *Balancing data* menjadi langkah penting untuk memastikan performa model yang adil dan akurat.

Synthetic Minority Over-sampling Technique (SMOTE) merupakan metode yang sering diaplikasikan untuk mengatasi problem ini. SMOTE bekerja dengan membuat sampel sintetis pada kelas minoritas melalui interpolasi linier antara sampel asli. Proses ini menjaga distribusi kelas minoritas sembari memperluas jumlah sampel untuk meningkatkan representasinya. Dengan demikian, metode ini dapat membantu meningkatkan performa model dalam mengklasifikasikan kelas minoritas.

Penelitian oleh (Chen dkk., 2020) menunjukkan bahwa SMOTE efektif dalam meningkatkan akurasi model klasifikasi pada dataset yang tidak seimbang, terutama pada algoritma seperti k-Nearest Neighbors dan Random Forest. Di sisi lain, penelitian lain menemukan bahwa meskipun SMOTE meningkatkan akurasi keseluruhan, ia dapat memperkenalkan korelasi antar-sampel dalam kelas minoritas yang dapat memengaruhi analisis statistik dan performa model tertentu, seperti logistic regression atau discriminant analysis.

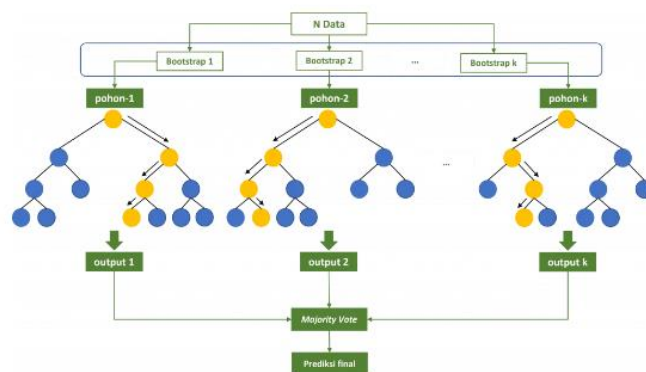
Dalam konteks penelitian ini, SMOTE diaplikasikan untuk menangani ketidakseimbangan data pada atribut target diabetes. Dengan demikian, proses

Balancing data ini diharapkan dapat meningkatkan performa model *Random Forest* dan *Adaboost* dalam mengklasifikasikan data pasien diabetes.

2.5 Random Forest

Random Forest adalah salah satu metode ensemble learning yang prosesnya menggabungkan beberapa pohon keputusan (*decision tree*) untuk meningkatkan hasil akurasi prediksi dan mengurangi *overfitting*. Setiap pohon keputusan dalam *Random Forest* dibangun dari sampel acak data pelatihan dan subset acak dari fitur. Keunggulan utama *Random Forest* adalah kemampuannya menangani data dengan jumlah fitur yang besar dan sifatnya yang robust terhadap data yang hilang atau nilai pencilan (Marsuhandi dkk., 2020).

Random Forest bekerja dengan membuat beberapa pohon keputusan dari sampel data yang berbeda (*bootstrap* sampling) dan menggunakan rata-rata prediksi dari semua pohon untuk menentukan hasil akhir. Proses ini membantu mengurangi variabilitas dan meningkatkan stabilitas model. Setiap pohon dalam hutan menghasilkan satu prediksi, dan hasil akhir ditentukan oleh mayoritas suara dari semua pohon (Suci Amaliah dkk. 2022).



Gambar 2. 1 Desain model *Random Forest*.

Sumber: <mailto:https://sainsdata.id/machine-learning/893/random-forest-untuk-model-klasifikasi-menggunakan-scikitlearn-python/>

Algoritma *Random Forest Classifier* dimulai dengan memuat data pelatihan ke dalam memori. Setelah itu, data tersebut dibagi menjadi beberapa subset acak menggunakan teknik bootstrap. Teknik ini memungkinkan pengambilan sampel dengan penggantian dari data pelatihan asli, sehingga setiap subset dapat mengandung data yang berulang (Marsuhandi dkk., 2020). Untuk setiap subset data yang dihasilkan, sebuah pohon keputusan dibuat. Pohon keputusan ini dibangun dengan membatasi kedalaman pohon atau jumlah maksimum fitur yang dapat digunakan untuk setiap split guna menghindari *overfitting*. Pada setiap node dalam pohon, subset fitur acak dipilih untuk mencari split terbaik, menambah variasi dan mengurangi korelasi antar pohon. Proses ini diulang sebanyak $n_estimator$ kali, yang merupakan jumlah pohon yang diinginkan dalam model ensemble (Tamba & -, 2022).

Setelah semua pohon keputusan dibuat, model siap untuk menentukan prediksi terhadap sampel baru. Prediksi ini dilakukan dengan mengumpulkan hasil dari setiap pohon keputusan. Untuk setiap pohon, data baru diklasifikasikan dan hasil prediksinya dicatat. Setelah semua hasil prediksi dari setiap pohon terkumpul, dilakukan agregasi prediksi dengan menghitung jumlah prediksi untuk setiap kelas dari semua pohon keputusan. Terakhir, data baru diklasifikasikan ke dalam kelas dengan jumlah prediksi terbanyak (Marsuhandi dkk., 2020). Artinya, kelas yang paling sering diprediksi oleh pohon-pohon keputusan akan menjadi kelas akhir untuk data baru, mengikuti prinsip mayoritas suara dalam ensemble learning.

2.6 Adaboost

Adaboost (Adaptive Boosting) adalah algoritma boosting yang dilakukan untuk menggabungkan beberapa model prediktif sederhana (*Weak Learners*) untuk membentuk model yang lebih kuat (Hornýák & Iantovics, 2023). Setiap model sederhana dilatih secara berurutan, dengan fokus pada data yang sebelumnya sulit diprediksi. *Adaboost* menyesuaikan bobot sampel berdasarkan kesalahan prediksi, sehingga model akhir lebih akurat dalam mengklasifikasikan data yang kompleks (Ding dkk., 2022).

Proses pelatihan *Adaboost* dimulai dengan memberikan bobot yang sama pada semua sampel data. Setelah model pertama dilatih, bobot dari sampel yang salah diprediksi ditingkatkan, sementara bobot dari sampel yang benar diprediksi diturunkan. Model selanjutnya dilatih dengan mempertimbangkan bobot yang telah disesuaikan ini. Proses ini diulangi hingga jumlah model yang ditentukan tercapai, dan prediksi akhir diperoleh dari kombinasi tertimbang dari semua model (Purnajiwa Arimbawa & Sanjaya ER, 2020).

2.6 Feature Importance

Feature importance adalah teknik yang digunakan untuk menentukan kontribusi setiap fitur dalam model prediktif terhadap hasil yang diinginkan (Chen dkk., 2020). Dalam penelitian ini, *feature importance* digunakan untuk menganalisis pengaruh atribut-atribut tertentu, seperti usia, terhadap klasifikasi penyakit diabetes.

Metode *feature importance* sering kali diterapkan dalam algoritma seperti *Random Forest*, yang secara otomatis menghitung nilai penting dari setiap fitur selama proses pelatihan. Fitur dengan nilai penting yang lebih tinggi dianggap lebih berpengaruh dalam memprediksi hasil dibandingkan fitur dengan nilai penting yang lebih rendah (Harmaja dkk., 2023).

Dalam studi yang dilakukan oleh (K. B. Simarmata dkk., 2022) mengenai faktor-faktor yang mempengaruhi persetujuan pinjaman di sektor teknologi finansial, *feature importance* digunakan untuk mengidentifikasi atribut-atribut utama, seperti pendapatan tahunan dan pembayaran bulanan, yang berkontribusi signifikan terhadap keputusan persetujuan. Dengan memahami fitur mana yang paling berpengaruh, peneliti dapat membantu perusahaan membuat keputusan yang lebih akurat dan efisien.

2.7 Faktor Usia terhadap Klasifikasi Diabetes

Usia telah diidentifikasi sebagai salah satu sebab penting dalam resiko pengembangan penyakit diabetes. Penelitian oleh (Susilawati and Rahmawati 2021) dalam studi mereka berjudul "Hubungan Usia, Jenis Kelamin, dan Hipertensi dengan Kejadian Diabetes Mellitus Tipe 2 di Puskesmas Tugu Kecamatan Cimanggis Kota Depok" menunjukkan bahwa usia memiliki pengaruh signifikan terhadap kejadian diabetes mellitus tipe 2. Mereka menemukan bahwa seiring bertambahnya usia, risiko seseorang untuk mengembangkan diabetes juga meningkat, terutama pada individu yang memiliki faktor risiko lainnya seperti hipertensi dan riwayat keluarga.

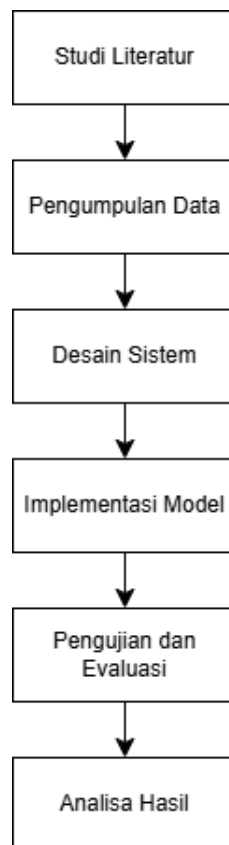
Penelitian lain oleh (Arania dkk. 2021) juga mendukung temuan ini, di mana mereka mengidentifikasi bahwa kelompok usia yang lebih tua memiliki prevalensi diabetes yang lebih tinggi. Mereka menemukan bahwa perubahan metabolisme dan penurunan fungsi insulin seiring bertambahnya usia berkontribusi pada peningkatan risiko diabetes.

BAB III

DESAIN DAN IMPLEMENTASI

3.1 Tahapan Penelitian

Tahapan penelitian ini alah alur utama yang menggambarkan proses penelitian ini. Gambar 3.1 adalah metode dan langkah - langkah yang akan digunakan agar penulisan terstruktur dengan baik.



Gambar 3. 1 Tahapan Penelitian

3.2 Pengumpulan Data

Data yang digunakan merupakan data sekunder. Dataset yang digunakan dalam penelitian ini adalah "Diabetes" yang merupakan data *open access* data web dari situs Kaggle dengan alamat [kaggle.com/datasets/rashikrahmanpritom/heart-](https://www.kaggle.com/datasets/rashikrahmanpritom/heart-)

[attack-analysis-prediction-dataset/data](#). Dataset ini mencakup 403 data individu dan terdiri dari 19 atribut. Pada penelitian ini digunakan 17 fitur dimana 1 atributnya yaitu Glybh yang merupakan variabel output dan kolom id tidak digunakan karena tidak mempengaruhi hasil penelitian ini.

Atribut-atribut tersebut menyajikan berbagai informasi penting yang dibutuhkan untuk menganalisis risiko diabetes.

Tabel 3. 1 Penjelasan Fitur

	Fitur	Keterangan
1.	Id	Identifikasi unik untuk setiap pasien
2.	Chol	Kadar kolesterol dalam darah (mg/dl)
3.	Stab.glu	Kadar glukosa darah stabil (mg/dl)
4.	HDL	High-Density Lipoprotein, "kolesterol baik" (mg/dl)
5.	Ratio	Rasio antara HDL dan kolesterol total
6.	Glyhb	Hemoglobin terglikosilasi (HbA1c) (%)
7.	Location	Lokasi geografis pasien
8.	Age	Usia pasien (tahun)
9.	Gender	Jenis kelamin pasien (male/female)
10.	Height	Tinggi badan pasien
11.	Weight	Berat badan pasien
12.	Frame	Ukuran kerangka tubuh (small, medium, large)
13.	BP.1s	Tekanan darah sistolik pertama (mmHg)
14.	BP.1d	Tekanan darah diastolik pertama (mmHg)
15.	BP.2s	Tekanan darah sistolik kedua (mmHg)
16.	BP.2d	Tekanan darah diastolik kedua (mmHg)
17.	Waist	Lingkar pinggang pasien (inci)
18.	Hip	Lingkar pinggul pasien (inci)
19.	Time.ppn	Waktu pasca-prandial (menit)

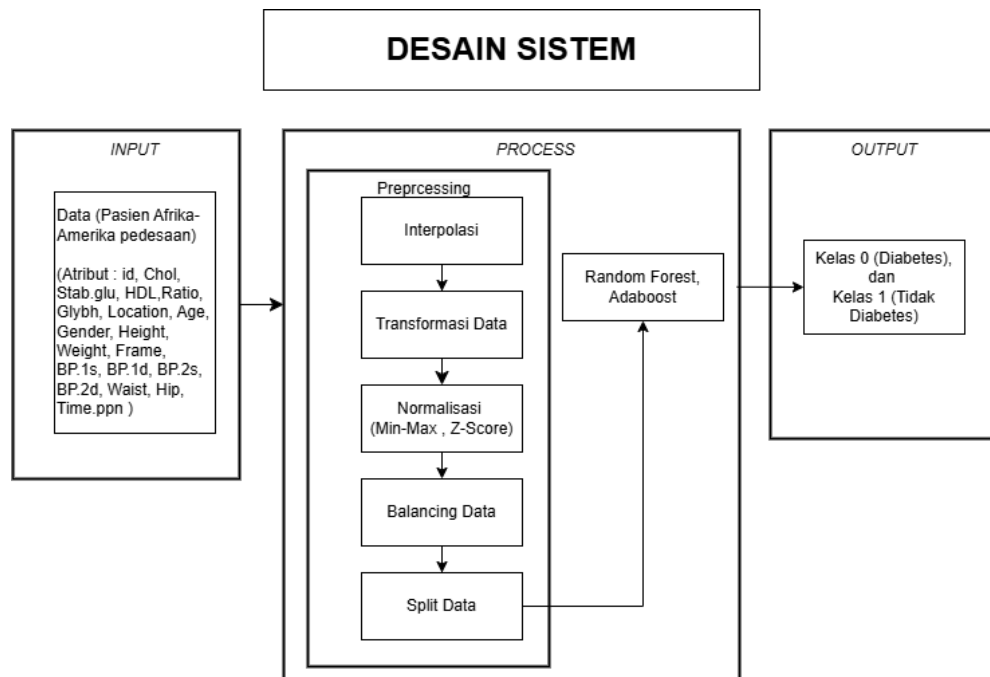
Tabel 3. 2 Contoh Dataset Penyakit Diabetes

Id	chol	Stab.glu	hdl	ratio	glyhb	...	bp.1s	bp.1d	bp.2s	bp.2d	waist	hip	Time.ppn
1	203	82	56	35.999...	43.099..	...	118	59			29	38	720
2	165	97	24	65.000..	44.400..	...	112	68			46	48	360
3	228	92	37	88.999..	46.399..	...	190	92	185	92	49	57	180
4	78	93	12	35.999..	46.300..	...	110	50			33	38	480
5	249	90	28	48.000..	77.199..	...	138	80			44	41	300
...
400	296	369	46	64.00..	161.100..	...	138	94	130	94	35	39	210
401	284	89	54	53.00..	43.899..	...	140	100	146	102	32	43	180
402	194	269	38	50.9..	136.300..	...	120	70			33	40	20
403	199	76	52	37.99..	44.899..	...	120	78			41	48	255
404	159	88	79	37.99..	49.899..	...	100	72			49	58	900

Dalam dataset ini, dari total 19 atribut, 18 fitur digunakan untuk model klasifikasi. Atribut yang dikecualikan adalah Glyhb, yang berfungsi sebagai variabel output atau target dalam klasifikasi penyakit diabetes. Penggunaan 18 fitur ini bertujuan untuk membangun model yang akurat dalam memprediksi kemungkinan seseorang menderita diabetes berdasarkan karakteristik yang terdapat dalam dataset.

3.3 Desain Sistem

Desain sistem yang digunakan dalam penelitian ini terdiri dari beberapa tahap sebagaimana telah ditunjukkan pada Gambar 3.2.



Gambar 3. 2 Desain sistem

Penelitian ini menggunakan bahasa Python dengan dataset "Diabetes" dari Kaggle, yang berisi data pasien Afrika-Amerika pedesaan. Tahapan *Preprocessing* dimulai dengan transformasi data kategori menjadi numerik menggunakan encoding dan one hot encoding, penanganan missing values menggunakan mean (interpolasi), dan normalisasi data menggunakan metode *Min-Max* dan *Z-Score* untuk memastikan skala data seragam. Selanjutnya, dilakukan penyeimbangan kelas menggunakan SMOTE dan pembagian dataset sebesar 80% data training dan 20% data testing. Setelah *Preprocessing*, kemudian diterapkan dua model klasifikasi, yaitu *Random Forest* dan *Adaboost*, dari hasil akurasi kedua model kemudian dibandingkan untuk menentukan algoritma mana yang lebih baik dalam klasifikasi diabetes.

3.4 Preprocessing

Preprocessing data bertujuan untuk memastikan bahwa data yang digunakan dalam algoritma berkualitas tinggi dan siap untuk diproses. Setelah data terkumpul, selanjutnya dilakukan *Preprocessing* yang meliputi:

3.4.1 Interpolasi

Pada penelitian ini, missing values ditangani dengan mengganti nilai yang hilang menggunakan mean. Metode yang digunakan untuk menangani missing value dalam penelitian ini adalah imputasi dengan mean. Imputasi mean dilakukan dengan menggantikan nilai yang hilang pada suatu atribut dengan rata-rata dari semua nilai yang tersedia pada atribut tersebut.

$$Mean = \frac{\sum x_i}{n} \quad (3.1)$$

Keterangan:

x_i : nilai ke-i dalam dataset,
 $\sum x_i$: jumlah seluruh nilai dalam dataset,
 n : jumlah total nilai atau elemen dalam dataset.

3.4.2 Transformasi Data

Transformasi data dilakukan untuk mengubah data kategori menjadi data numerik supaya dapat diolah oleh algoritma pembelajaran mesin. Dalam penelitian ini, data kategori diubah menjadi numerik menggunakan label encoding untuk kategori dengan dua nilai, seperti 'location' dan 'gender', dan one-hot encoding untuk kategori dengan lebih dari dua nilai, seperti 'frame'. Label encoding mengubah kategori menjadi angka unik, sementara one-hot encoding mengubah setiap kategori menjadi kolom biner (Potdar dkk., 2017).

3.4.3 Normalisasi Data

Normalisasi data adalah proses untuk menyamakan skala seluruh fitur dalam dataset sehingga tidak ada fitur yang mendominasi yang lain dalam proses pembelajaran model. Dengan tujuan agar memastikan semua fitur berkontribusi secara proposional dalam model pembelajaran mesin. Ini penting karena fitur dengan skala yang sangat berbeda dapat mendominasi algoritma dan dapat mempengaruhi akurasi. Dua metode normalisasi yang akan digunakan di penelitian ini adalah *Min-Max Normalization* dan *Z-Score Normalization*. Berikut rumus normalisasi menggunakan *Min-Max* dan *Z-Score* (Mohammed, 2020).

Min-Max Normalization: metode ini mengubah nilai dalam kolom tersebut ke dalam rentang 0 sampai 1. Berikut rumus normalisasi *Min-Max* :

$$x_{new} = \frac{x - \min(X)}{\max(X) - \min(X)} \quad (3.2)$$

Keterangan :

x_{new} : Nilai baru setelah normalisasi.
 x : Nilai asli yang akan dinormalisasi.
 $\min(X)$: Nilai minimum dari keseluruhan fitur X .
 $\max(X)$: Nilai maksimum dari keseluruhan fitur X .

Z-Score Normalization: metode ini mengubah nilai fitur menjadi skor standar (*Z-Score*) yang menyatakan berapa standar deviasi suatu nilai dari mean.

Rumus yang digunakan adalah:

$$z = \frac{x - \mu}{\sigma} \quad (3.3)$$

Keterangan :

z : Nilai *Z-Score* yang dihasilkan.
 x : Nilai asli yang akan dinormalisasi.
 μ : Mean (rata-rata) dari fitur yang bersangkutan.
 σ : Standar deviasi dari fitur yang bersangkutan.

Normalisasi ini memastikan bahwa setiap fitur berkontribusi secara proporsional dalam proses pelatihan model, dan tidak ada fitur yang mendominasi.

3.4.4 Synthetic Minority Oversampling Technique (SMOTE)

Pada tahap *Preprocessing*, dilakukan teknik *Synthetic Minority Oversampling Technique (SMOTE)* untuk mengatasi ketidakseimbangan data yang dapat mempengaruhi kinerja model klasifikasi. Dalam penelitian ini, dataset memiliki ketidakseimbangan signifikan, dengan 338 sampel pada kelas 0 (tidak diabetes) dan 65 sampel pada kelas 1 (diabetes). SMOTE menghasilkan sampel sintesis baru untuk kelas minoritas (kelas 1), yang membantu mengurangi perbedaan jumlah antara kedua kelas, sehingga meningkatkan akurasi model dalam mengklasifikasikan kedua kelas tersebut. Berikut langkah-langkah dalam menerapkan teknik SMOTE (Mohammed, 2020).

- a. Memilih secara acak satu sampel dari data minoritas.
- b. Mengidentifikasi tetangga terdekat (k) dari sampel minoritas yang terpilih.
- c. Membuat sampel data sintesis baru dengan menggabungkan sampel minoritas terpilih dengan tetangga terdekatnya menggunakan kombinasi berbobot, sesuai dengan persamaan yang relevan, dengan menggunakan persamaan berikut:

$$x_{new} = x_i + \alpha (x_j - x_i) \quad (3.4)$$

Keterangan :

- x_{new} : Instance sintesis baru yang dihasilkan.
- x_i : sampel kelas minoritas yang dipilih.
- x_j : Salah satu tetangga terdekat dari x_i , yang juga merupakan sampel dari kelas minoritas.
- α : bilangan acak rentang 0 dan 1.

3.4 Split Data

Setelah tahap preprocessing dan eksplorasi data selesai dilakukan, langkah berikutnya adalah membagi dataset menjadi dua bagian: data pelatihan (*training set*) dan data pengujian (*testing set*). Data pelatihan digunakan untuk membangun dan melatih model *Machine Learning*, sedangkan data pengujian berfungsi untuk menilai performa model pada data baru yang belum pernah diakses sebelumnya. Dalam penelitian ini, data dibagi menggunakan rasio 8:2, di mana 80% dari dataset dialokasikan untuk pelatihan dan 20% sisanya untuk pengujian..

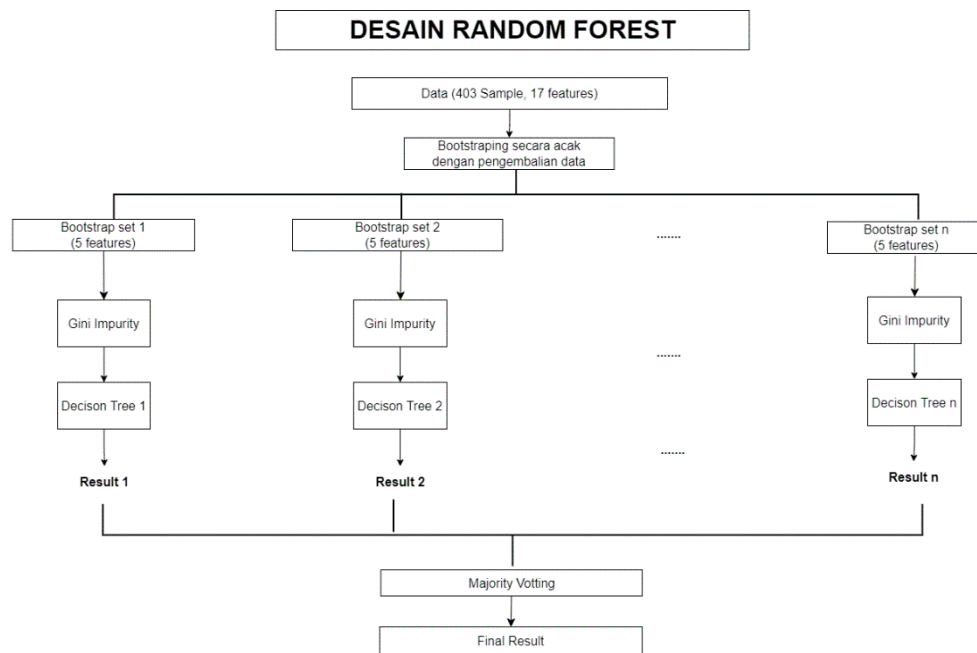
3.5 Implementasi Model

Setelah proses pembagian data dilakukan, langkah selanjutnya adalah mengimplementasikan model pembelajaran mesin yang telah dipilih. Pada tahap ini, model akan dilatih menggunakan data pelatihan yang telah disediakan, dan kemudian diuji dengan data pengujian untuk mengevaluasi kinerjanya. Pemilihan model yang tepat dan penerapan parameter yang sesuai sangat berpengaruh terhadap hasil akhir prediksi. Dalam penelitian ini, dua model yang digunakan adalah *Random Forest* dan *Adaboost*, yang masing-masing termasuk dalam metode ensemble. Berikut adalah implementasi dari kedua model tersebut.

3.5.1 Random Forest

Pada Gambar 3.2 menjelaskan desain Model *Random Forest*, dilakukan setelah data dibagi menjadi dua bagian utama, yaitu data pelatihan (*training*) dan data pengujian (*testing*), beberapa parameter penting dalam algoritma *Random Forest* adalah metode *ensemble* yang menggunakan beberapa *Decision Tree* untuk

klasifikasi. Algoritma ini bekerja dengan membuat banyak *Decision Tree* selama proses pelatihan, dan hasil akhir ditentukan berdasarkan *voting* dari prediksi masing-masing *tree*.



Gambar 3. 3 Desain Model *Random Forest*

Dalam implementasi model, setelah dilakukan *Preprocessing* fitur yang digunakan menjadi 19 fitur dan 1 label untuk membangun model *Random Forest*. Berikut adalah langkah-langkah algoritma *Random Forest* (Jackins dkk., 2021) :

- a. Mengambil sampel dari dataset beberapa kali dengan pengembalian (*Bootstrap Aggregating (Bagging)*). Pemilihan fitur diambil secara acak dapat ditentukan dengan menghitung \sqrt{m} dimana m adalah jumlah fitur selain fitur target (Mahmuda, 2024).
- b. Membuat pohon keputusan dengan memilih fitur-fitur secara acak dari setiap subset data berdasarkan kriteria gini.

$$Gini_{(S_i)} = 1 - \sum_{i=1}^{c-1} p_i^2 \quad (3.5)$$

Keterangan :

p_i : probabilitas setiap kelas pada node
 c : jumlah kelas

- c. Dari fitur yang dipilih secara acak tersebut kemudian mencari titik split terbaik berdasarkan kriteria Gini.

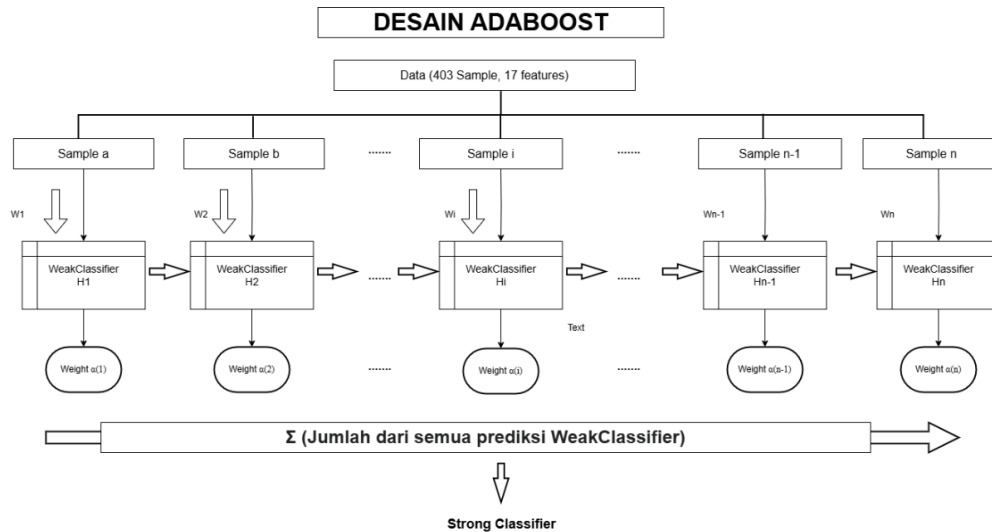
$$Gini_{Split} = \sum_{i=1}^{c-1} \left(\frac{n_i}{n}\right) \times Gini_{(S_i)} \quad (3.6)$$

Keterangan :

c : jumlah kelas
 n_i : total data pada satu kelas
 n : total seluruh data

- d. Proses pemilihan fitur dan mencari titik split terbaik berlanjut untuk setiap node dalam pohon sampai pohon selesai dibangun.
- e. Melakukan klasifikasi melalui suara terbanyak (voting) dari gabungan pohon yang ada.

3.5.2 Adaboost



Gambar 3. 4 Desain Model Adaboost

Pada tahap ini, algoritma *Adaboost* diterapkan untuk membangun model klasifikasi. *Adaboost (Adaptive Boosting)* adalah ensemble learning method yang membentuk model kuat dengan menggabungkan beberapa model lemah (*Weak Learners*), biasanya *decision stumps (Decision Tree* dengan satu node). Setiap model lemah difokuskan pada kesalahan yang dibuat oleh model sebelumnya dengan memberikan bobot lebih pada data yang sulit diklasifikasikan.

Berikut adalah Langkah-langkah algoritma *Adaboost* (Wang & Sun, 2021):

- a. **Inisialisasi Bobot Data:** Langkah pertama dalam *Adaboost* adalah memberikan bobot awal yang sama untuk setiap sampel data. Jika terdapat n sampel data, maka setiap sampel i akan memiliki bobot awal yang dapat ditentukan menggunakan persamaan 3.7.

$$w_i = \frac{1}{N} \quad (3.7)$$

Keterangan :

- w_i : bobot sampel ke- i pada iterasi pertama.
 N : jumlah total sampel

Bobot ini akan digunakan untuk menentukan seberapa penting setiap sampel dalam proses pelatihan model.

- b. **Pelatihan Model Klasifikasi:** Pada setiap iterasi, model lemah (misalnya, pohon keputusan dengan kedalaman satu) dilatih menggunakan data dengan bobot yang telah diinisialisasi. Setelah pelatihan, model akan menghasilkan prediksi untuk setiap sampel.
- c. **Menghitung Error Model:** Setelah model dilatih, hitung tingkat kesalahan (error) pada data pelatihan. Perhitungan eror model ditunjukkan pada persamaan 3.8.

$$\varepsilon_t = \frac{\sum_i w_i \cdot I[h_t(x_i) \neq y_i]}{\sum_i w_i} \quad (3.8)$$

Keterangan :

- ε_t : Tingkat kesalahan (*error rate*) dari model pada iterasi ke- t .
 \sum_i : Penjumlahan untuk semua data pelatihan i .
 w_i : Bobot dari sampel ke- i .
 $I[h_t(x_i) \neq y_i]$: Fungsi indikator yang bernilai 1 jika prediksi model salah, dan 0 jika benar.
 $\sum_i w_i$: Total bobot, digunakan sebagai penyebut untuk menghitung rata-rata tertimbang dari kesalahan.

- d. **Menghitung Bobot Model:** Setelah tingkat kesalahan diketahui, bobot model tersebut dihitung menggunakan persamaan 3.9.

$$\alpha_t = \frac{1}{2} \log \left(\frac{1 - \varepsilon_t}{\varepsilon_t} \right) \quad (3.9)$$

Keterangan :

- α_t : Bobot dari model pada iterasi ke- t , menunjukkan seberapa baik model dalam mengklasifikasikan data.
 ε_t : Tingkat kesalahan dari model pada iterasi ke- t .
 \log : Fungsi logaritma natural yang digunakan untuk menghitung bobot.

Bobot ini menggambarkan seberapa baik model tersebut dalam mengklasifikasikan data.

- e. **Memperbarui Bobot Data:** Bobot data kemudian diperbarui untuk fokus lebih pada sampel yang sulit diklasifikasikan. Perhitungan untuk memperbarui bobot data ditunjukkan pada persamaan 3.10.

$$w_{i,t+1} = w_{i,t} \cdot \exp(\alpha t \cdot I [h_t (x_i) \neq y_i]) \quad (3.10)$$

Keterangan:

- $w_{i,t+1}$: Bobot dari sampel ke- i pada iterasi ke- $t + 1$.
 $w_{i,t}$: Bobot dari sampel ke- i pada iterasi ke- t .
 \exp : Fungsi eksponensial yang digunakan untuk menghitung perubahan bobot.
 αt : Bobot dari model pada iterasi ke- t , yang menunjukkan seberapa baik model dalam mengklasifikasikan data.
 $I [h_t (x_i) \neq y_i]$: Fungsi indikator yang bernilai 1 jika prediksi model salah, dan 0 jika benar.

- f. **Pengulangan:** Proses pelatihan diulang untuk sejumlah iterasi T . Pada setiap iterasi, model baru dilatih dengan bobot yang diperbarui, dan bobot model baru dihitung.
- g. **Prediksi Akhir:** Prediksi akhir untuk data uji dihasilkan dengan menggabungkan prediksi dari semua model yang dilatih, masing-masing dengan bobotnya. Prediksi akhir dapat ditentukan dari persamaan 3.11.

$$\text{Prediksi Akhir} = \text{sign}(\sum_{t=1}^T \alpha_t \cdot h_t(x)) \quad (3.11)$$

Keterangan :

- $h_t(x)$: Prediksi dari model lemah (*weak learner*) ke- t untuk sampel x , di mana $h_t(x)$ bernilai +1 untuk kelas positif atau -1 untuk kelas negatif.
 α_t : Bobot dari model lemah ke- t , yang dihitung berdasarkan tingkat kesalahan model tersebut. Model dengan kinerja lebih baik akan memiliki bobot lebih tinggi.
 $\sum_{t=1}^T$: Menunjukkan penjumlahan dari seluruh model lemah (dari iterasi 1 hingga T).
 sign : Fungsi tanda yang menentukan prediksi akhir. Jika hasil penjumlahan bernilai positif, prediksi akhir adalah +1 (kelas positif). Jika bernilai negatif, prediksi akhir adalah -1 (kelas negatif). Dengan demikian, kelas yang memiliki jumlah bobot terbesar akan dipilih sebagai prediksi akhir.

3.6 Evaluasi Model

Setiap model akan dievaluasi menggunakan confusion matrix, yaitu Tabel yang digunakan untuk menunjukkan jumlah data uji yang diklasifikasikan dengan benar maupun salah. *Confusion matrix* ini terdiri dari empat komponen utama: *True Positive* (TP), *True Negative* (TN), *False Positive* (FP), dan *False Negative* (FN). *True Positive* (TP) mengacu pada jumlah data dari kelas positif yang diklasifikasikan dengan benar sebagai kelas positif. *True Negative* (TN) adalah jumlah data dari kelas negatif yang diklasifikasikan dengan benar sebagai kelas negatif. *False Positive* (FP) adalah data dari kelas negatif yang salah diklasifikasikan sebagai kelas positif, sedangkan *False Negative* (FN) merupakan data dari kelas positif yang salah diklasifikasikan sebagai kelas negatif (Normawati & Prayogi, 2021).

Tabel 3. 3 *Confusion matrix*

Kelas Sebenarnya	Kelas Prediksi	
	Positif	Negatif
Positif	True Positive (TP)	False Negative (FN)
Negatif	False Positive (FP)	True Negative (TN)

Persamaan Confusion Matrix yang digunakan untuk menghitung performa akurasi pada Persamaan 3.12, sementara untuk menghitung performa presisi terdapat pada Persamaan 3.13, performa recall pada Persamaan 3.14, dan f-measure pada Persamaan 3.15.(Rohman & Arifianto, 2021).

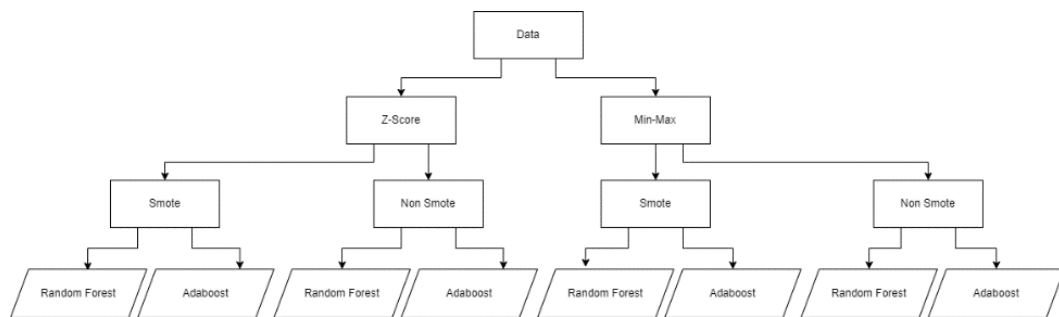
$$Akurasi(\%) = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (3.12)$$

$$Presisi(\%) = \frac{TP}{TP+FP} \times 100\% \quad (3.13)$$

$$Recall(\%) = \frac{TP}{TP+FN} \times 100\% \quad (3.14)$$

$$F1 - Score = \frac{2 \times \text{Presisi} \times \text{Recall}}{\text{Presisi} + \text{Recall}} \quad (3.15)$$

3.7 Skenario Pengujian



Gambar 3. 5 Skenario Pengujian

Pada Gambar 3.5 ini menunjukkan skenario uji coba pada penelitian ini. Langkah pertama setelah data dilakukan perubahan kategori ke numerik dan pengisian data pada missing value (interpolasi), kemudian pengolahan menggunakan dua metode normalisasi, yaitu *Z-Score* dan *Min-Max*. Kemudian diproses dengan metode SMOTE dan non-SMOTE untuk menangani ketidakseimbangan data. Hasil dari kombinasi normalisasi dan teknik penyeimbangan ini digunakan untuk melatih dua algoritma, yaitu *Random Forest* dan *Adaboost*. Uji coba ini bertujuan untuk memahami pengaruh *Preprocessing* data dan metode *Balancing* terhadap kinerja algoritma *Random Forest* dan *Adaboost* dalam berbagai kondisi.

Tabel 3. 4 Skenario Pengujian

Uji Coba	Normalisasi	Balancing	Metode
1	<i>Z-Score</i>	SMOTE	<i>Random Forest</i> (n=50, d=5)
2	<i>Z-Score</i>	SMOTE	<i>Random Forest</i> (n=50, d=7)

Uji Coba	Normalisasi	Balancing	Metode
3	<i>Z-Score</i>	SMOTE	<i>Random Forest (n=50, d=10)</i>
4	<i>Z-Score</i>	SMOTE	<i>Random Forest (n=100, d=5)</i>
5	<i>Z-Score</i>	SMOTE	<i>Random Forest (n=100, d=7)</i>
6	<i>Z-Score</i>	SMOTE	<i>Random Forest (n=100, d=10)</i>
7	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=50, d=5)</i>
8	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=50, d=7)</i>
9	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=50, d=10)</i>
10	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=100, d=5)</i>
11	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=100, d=7)</i>
12	<i>Z-Score</i>	Non-SMOTE	<i>Random Forest (n=100, d=10)</i>
13	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=50, d=5)</i>
14	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=50, d=7)</i>
15	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=50, d=10)</i>
16	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=100, d=5)</i>
17	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=100, d=7)</i>
18	<i>Min-Max</i>	SMOTE	<i>Random Forest (n=100, d=10)</i>
19	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=50, d=5)</i>
20	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=50, d=7)</i>
21	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=50, d=10)</i>
22	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=100, d=5)</i>
23	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=100, d=7)</i>
24	<i>Min-Max</i>	Non-SMOTE	<i>Random Forest (n=100, d=10)</i>
25	<i>Z-Score</i>	SMOTE	<i>Adaboost (n=30)</i>
26	<i>Z-Score</i>	SMOTE	<i>Adaboost (n=50)</i>
27	<i>Z-Score</i>	SMOTE	<i>Adaboost (n=70)</i>
28	<i>Z-Score</i>	Non-SMOTE	<i>Adaboost (n=30)</i>

Uji Coba	Normalisasi	Balancing	Metode
29	<i>Z-Score</i>	Non-SMOTE	<i>Adaboost</i> (n=50)
30	<i>Z-Score</i>	Non-SMOTE	<i>Adaboost</i> (n=70)
31	<i>Min-Max</i>	SMOTE	<i>Adaboost</i> (n=30)
32	<i>Min-Max</i>	SMOTE	<i>Adaboost</i> (n=50)
33	<i>Min-Max</i>	SMOTE	<i>Adaboost</i> (n=70)
34	<i>Min-Max</i>	Non-SMOTE	<i>Adaboost</i> (n=30)
35	<i>Min-Max</i>	Non-SMOTE	<i>Adaboost</i> (n=50)
36	<i>Min-Max</i>	Non-SMOTE	<i>Adaboost</i> (n=70)

Tabel 3.4 adalah skenario penelitian untuk membandingkan performa algoritma *Random Forest* dan *Adaboost* dalam klasifikasi diabetes. Model 1 hingga 24 menggunakan *Random Forest* dengan variasi *n_estimator* (50 dan 100) dan *max_depth* (5, 7, dan 10). Model 25 hingga 36 menggunakan *Adaboost* dengan *n_estimator* 30, 50, dan 70. Setiap model juga diuji dengan kombinasi metode normalisasi (*Z-Score* dan *Min-Max*) serta teknik *Balancing* (SMOTE dan Non-SMOTE). Variasi ini dirancang untuk menentukan konfigurasi optimal dalam meningkatkan akurasi klasifikasi diabetes.

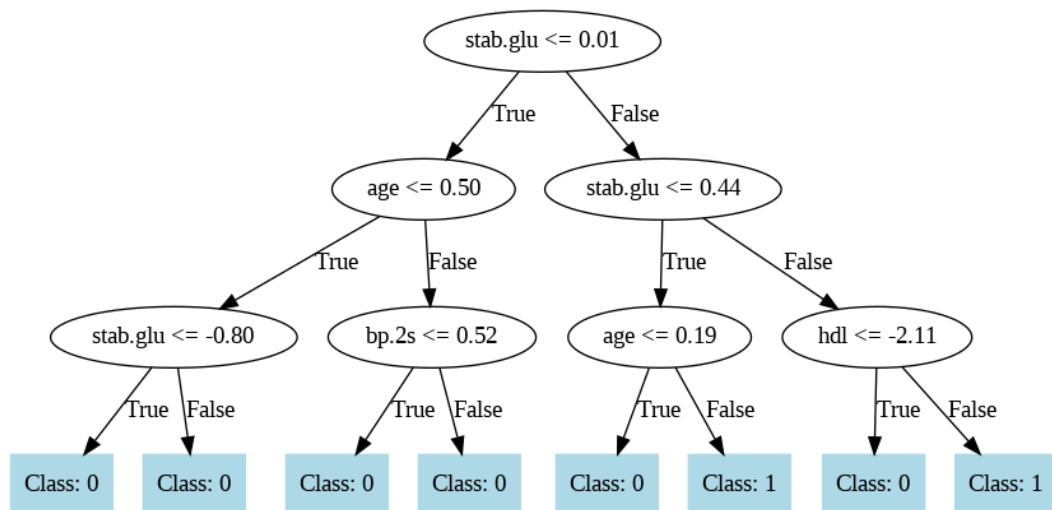
BAB IV

UJI COBA DAN PEMBAHASAN

Pada bagian ini merupakan hasil pengujian model untuk klasifikasi penyakit diabetes menggunakan algoritma *Random Forest* dan *Adaboost*.

4.1 Hasil Training

Pada Gambar 4.1 adalah visualisasi dari salah satu pembangunan pohon keputusan yang dibangun menggunakan metode *Random Forest*.



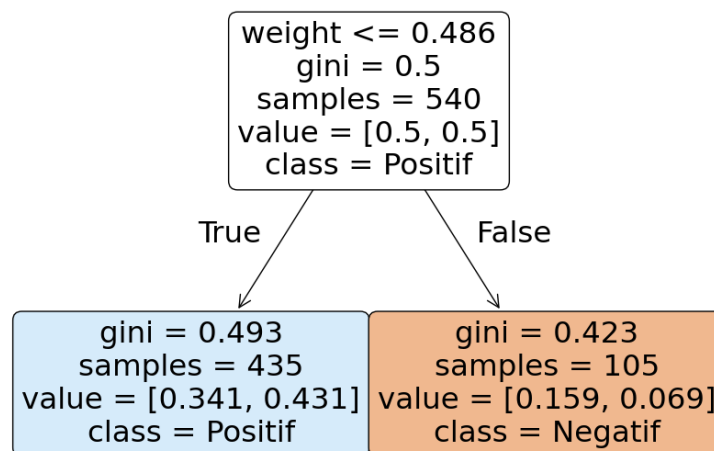
Gambar 4. 1 Visualisasi Salah satu hasil pembangunan Pohon Keputusan

Gambar 4.1 menunjukkan salah satu pohon keputusan yang dibangun dengan menggunakan metode *Gini Impurity*. *Gini Impurity* adalah ukuran yang digunakan untuk memilih fitur yang paling efektif dalam memisahkan data. Pada setiap simpul (node), fitur dengan nilai *Gini Impurity* terkecil akan dipilih untuk membagi data lebih lanjut. Semakin kecil nilai *Gini Impurity*, semakin baik fitur

tersebut dalam memisahkan kelas, sehingga fitur yang lebih informatif ditempatkan di posisi atas pohon.

Pada pohon keputusan ini, pada setiap simpul, model memeriksa nilai fitur tertentu dan membandingkannya dengan *threshold* yang ditentukan di simpul tersebut. Jika nilai fitur kurang dari / sama dengan *threshold*, data akan diteruskan ke cabang kiri, jika lebih besar, data akan diteruskan ke cabang kanan. Proses ini berulang hingga mencapai simpul daun yang tidak dapat dibagi lagi, yang menunjukkan keputusan akhir. Di simpul daun terakhir, keputusan akhir diberikan, yang menunjukkan apakah individu dalam data uji ini termasuk dalam kelas 0 (bukan diabetes) atau kelas 1 (diabetes).

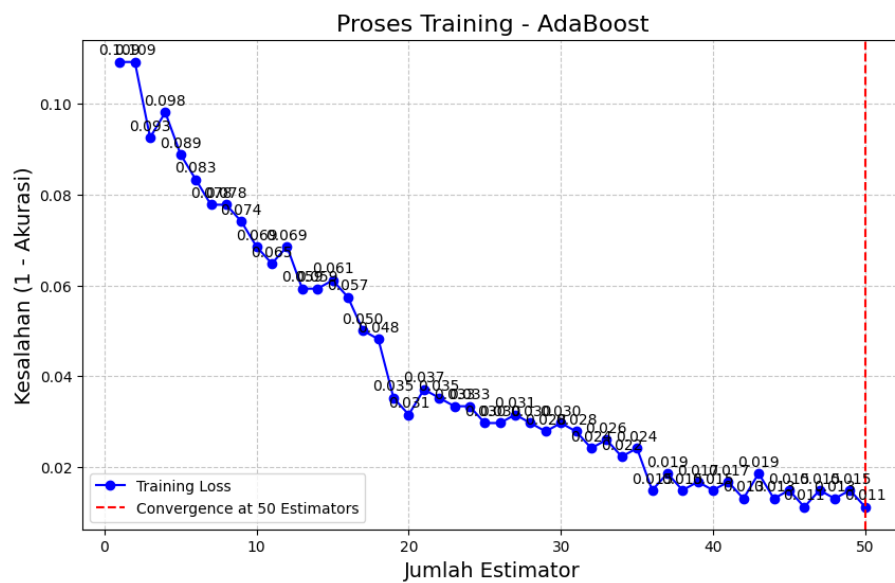
Decision Tree Ke-50 dalam Model AdaBoost



Gambar 4. 2 Visualisasi *Decision Stump* pada *Adaboost*

Pada Gambar 4.2 ditampilkan visualisasi dari salah satu *Decision Stump* yang dibangun dalam model *Adaboost*. *Decision Stump* adalah model pohon keputusan dengan kedalaman sangat terbatas, yaitu hanya memiliki satu level pembagian (satu simpul internal dan dua simpul daun). Pada visualisasi ini, pohon keputusan hanya melakukan pembagian berdasarkan satu fitur (misalnya *weight*)

dengan sebuah *threshold* tertentu. Pada model *Adaboost*, setiap iterasi model berfokus pada data yang salah diklasifikasikan oleh iterasi sebelumnya dan memberikan bobot yang lebih tinggi pada data tersebut. *Decision Stump* ini dioptimalkan untuk menghasilkan prediksi yang lebih baik dengan meningkatkan fokus pada sampel yang lebih sulit diprediksi.



Gambar 4. 3 Error pada *Model Adaboost*

Gambar 4.3 menunjukkan Error atau kesalahan pada model *Adaboost* saat proses training. Error ini dihitung berdasarkan prediksi yang dihasilkan oleh model. Grafik ini menggambarkan bagaimana error berkurang seiring dengan bertambahnya jumlah estimator yang digunakan dalam model *Adaboost*. Pada grafik, sumbu x menunjukkan jumlah estimator (iteration), sementara sumbu y menunjukkan tingkat kesalahan. Seiring bertambahnya jumlah *Decision Stumps*, model *Adaboost* semakin fokus pada data yang salah klasifikasi sebelumnya, yang menyebabkan penurunan tingkat kesalahan. Penurunan error ini mencerminkan peningkatan performa model selama proses training.

4.2 Hasil Uji Coba Model *Random Forest*

Pada model ini menggunakan *Random Forest*. Pada pengujian ini dilakukan normalisasi *Z-Score*, normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan tanpa menggunakan *Balancing*. Pada setiap model menggunakan kombinasi parameter yaitu *n_estimator* dan *max_depth*.

4.2.1 Model 1

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter *n_estimator* = 50 dan *max_depth* = 5. Hasil pengujian yang didapat sebagai berikut. Evaluasi Matrik dari pengujian ini dilihat pada Tabel 4.1.

Tabel 4. 1 *Confusion matrix* Uji Coba Ke-1

TP	FP	TN	FN
72	5	57	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 95%, precision 94%, *recall* 97%, dan *f1-score* 95%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* 95%, *recall macro average* 95%, dan *f1-score macro average* 95%.

4.2.2 Model 2

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter *n_estimator* = 50 dan *max_depth* = 7. Hasil pengujian yang didapat sebagai berikut. Matrik Keputusan dari pengujian ini dapat dilihat pada Tabel 4.2.

Tabel 4. 2 Confusion matrix Uji Coba Ke-2

TP	FP	TN	FN
72	3	59	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 96%, presisi sebesar 96%, *recall* sebesar 97%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 96%, *recall macro average* sebesar 96%, dan *f1-score macro average* sebesar 96%.

4.2.3 Model 3

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.3.

Tabel 4. 1 *Confusion matrix* Uji Coba Ke-3

TP	FP	TN	FN
72	1	61	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 98%, presisi sebesar 99%, *recall* sebesar 97%, dan *f1-score* sebesar 98%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 98%, *recall macro average* sebesar 98%, dan *f1-score macro average* sebesar 98%.

4.2.4 Model 4

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 100$ dan

$max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.4.

Tabel 4. 2 *Confusion matrix* Uji Coba Ke-4

TP	FP	TN	FN
70	3	59	4

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 95%, presisi sebesar 96%, *recall* sebesar 95%, dan *f1-score* sebesar 95%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 95%, *recall macro average* sebesar 95%, dan *f1-score macro average* sebesar 95%.

4.2.5 Model 5

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.5.

Tabel 4. 3. *Confusion matrix* Uji Coba Ke-5

TP	FP	TN	FN
72	4	58	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 96%, presisi sebesar 95%, *recall* sebesar 97%, dan *f1-score* sebesar 96%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 96%, *recall macro average* sebesar 95%, dan *f1-score macro average* sebesar 96%.

4.2.6 Model 6

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.6.

Tabel 4. 4 *Confusion matrix* Uji Coba Ke-6

TP	FP	TN	FN
71	0	62	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 98%, presisi 100%, *recall* 96%, dan *f1-score* 98%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 98%, *recall macro average* sebesar 98%, dan *f1-score macro average* sebesar 98%.

4.2.7 Model 7

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.7.

Tabel 4. 5 *Confusion matrix* Uji Coba Ke-7

TP	FP	TN	FN
65	12	3	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 84%, *precision* sebesar 75%, *recall* sebesar 20%, dan *f1-score* sebesar 32%. Selain itu terdapat beberapa hasil lain yang didapatkan

meliputi presisi *macro average* sebesar 80%, *recall macro average* sebesar 59%, dan *f1-score macro average* sebesar 61%.

4.2.8 Model 8

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.8.

Tabel 4. 6 *Confusion matrix* Uji Coba Ke-8

TP	FP	TN	FN
65	11	4	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 85%, presisi sebesar 86%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 83%, *recall macro average* sebesar 63%, dan *f1-score macro average* sebesar 66%.

4.2.9 Model 9

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.9.

Tabel 4. 7. *Confusion matrix* Uji Coba Ke-9

TP	FP	TN	FN
65	9	6	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 88%, presisi sebesar 88%, *recall* sebesar 98%, dan *f1-score* sebesar 93%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 87%, *recall macro average* sebesar 69%, dan *f1-score macro average* sebesar 74%.

4.2.10 Model 10

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.10.

Tabel 4. 8 *Confusion matrix* Uji Coba Ke-10

TP	FP	TN	FN
65	10	5	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 86%, presisi sebesar 87%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 85%, *recall macro average* sebesar 66%, dan *f1-score macro average* sebesar 70%.

4.2.11 Model 11

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.11.

Tabel 4. 9 *Confusion matrix* Uji Coba Ke-11

TP	FP	TN	FN
65	12	3	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 84%, presisi sebesar 84%, *recall* sebesar 98%, dan *f1-score* sebesar 91%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 80%, *recall macro average* sebesar 59%, dan *f1-score macro average* sebesar 61%.

4.2.12 Model 12

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.12.

Tabel 4. 10 *Confusion matrix* Uji Coba Ke-12

TP	FP	TN	FN
65	10	5	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 86%, presisi sebesar 87%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 85%, *recall macro average* sebesar 66%, dan *f1-score macro average* sebesar 70%.

4.2.13 Model 13

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 50$ dan $max_depth = 5$.

Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.13.

Tabel 4. 11 *Confusion matrix* Uji Coba Ke-13

TP	FP	TN	FN
72	7	55	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 91%, *recall* sebesar 97%, dan *f1-score* sebesar 94%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 94%, *recall macro average* sebesar 93%, dan *f1-score macro average* sebesar 93%.

4.2.14 Model 14

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 50$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.14.

Tabel 4. 12. *Confusion matrix* Uji Coba Ke-14

TP	FP	TN	FN
71	1	61	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 97%, presisi sebesar 99%, *recall* sebesar 96%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 97%, *recall macro average* sebesar 97%, dan *f1-score macro average* sebesar 97%.

4.2.15 Model 15

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 50$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.15.

Tabel 4. 13. *Confusion matrix* Uji Coba Ke-15

TP	FP	TN	FN
70	0	62	4

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 97%, presisi sebesar 100%, *recall* sebesar 95%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 97%, *recall macro average* sebesar 97%, dan *f1-score macro average* sebesar 97%.

4.2.16 Model 16

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 100$ dan $max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.16

Tabel 4. 14 *Confusion matrix* Uji Coba Ke-16

TP	FP	TN	FN
71	2	60	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 96%, presisi sebesar 97%, *recall* sebesar 96%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan

meliputi presisi *macro average* sebesar 96%, recall *macro average* sebesar 96%, dan f1-score *macro average* sebesar 96%.

4.2.17 Model 17

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 100$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.17.

Tabel 4. 15 *Confusion matrix* Uji Coba Ke-17

TP	FP	TN	FN
70	0	62	4

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 97%, presisi sebesar 100%, *recall* sebesar 95%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 97%, recall *macro average* sebesar 97%, dan f1-score *macro average* sebesar 97%.

4.2.18 Model 18

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan $n_estimator = 100$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.18.

Tabel 4. 16 *Confusion matrix* Uji Coba Ke-18

TP	FP	TN	FN
72	1	61	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 98%, presisi sebesar 99%, *recall* sebesar 97%, dan *f1-score* sebesar 98%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 98%, *recall macro average* sebesar 98%, dan *f1-score macro average* sebesar 98%.

4.2.19 Model 19

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.19.

Tabel 4. 17 *Confusion matrix* Uji Coba Ke-19

TP	FP	TN	FN
65	11	4	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 85%, presisi sebesar 86%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 83%, *recall macro average* sebesar 63%, dan *f1-score macro average* sebesar 66%.

4.2.20 Model 20

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.20.

Tabel 4. 18 *Confusion matrix* Uji Coba Ke-20

TP	FP	TN	FN
65	9	6	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 88%, presisi sebesar 88%, *recall* sebesar 98%, dan *f1-score* sebesar 93%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 87%, *recall macro average* sebesar 69%, dan *f1-score macro average* sebesar 74%.

4.2.21 Model 21

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 50$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.21.

Tabel 4. 19 *Confusion matrix* Uji Coba Ke-21

TP	FP	TN	FN
65	10	5	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 86%, presisi sebesar 87%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 85%, *recall macro average* sebesar 66%, dan *f1-score macro average* sebesar 70%.

4.2.22 Model 22

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 5$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.22.

Tabel 4. 20 *Confusion matrix* Uji Coba Ke-22

TP	FP	TN	FN
65	9	6	1

Berdasarkan hasil perhitungan dengan skenario perbandingan data 8:2 didapatkan akurasi sebesar 88%, presisi sebesar 88%, *recall* sebesar 98%, dan *f1-score* sebesar 93%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 87%, *recall macro average* sebesar 69%, dan *f1-score macro average* sebesar 74%.

4.2.23 Model 23

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 7$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.23.

Tabel 4. 21 *Confusion matrix* Uji Coba Ke-23

TP	FP	TN	FN
64	11	4	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 84%, presisi sebesar 85%, *recall* sebesar 97%, dan *f1-score* sebesar 91%. Selain itu terdapat beberapa hasil lain yang didapatkan

meliputi presisi *macro average* sebesar 76%, recall *macro average* sebesar 62%, dan f1-score *macro average* sebesar 64%.

4.2.24 Model 24

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 100$ dan $max_depth = 10$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.24.

Tabel 4. 22 *Confusion matrix* Uji Coba Ke-24.

TP	FP	TN	FN
65	10	5	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 86%, presisi sebesar 87%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 85%, recall *macro average* sebesar 66%, dan f1-score *macro average* sebesar 70%.

Secara keseluruhan, hasil ini menunjukkan bahwa pemilihan parameter $n_estimators$, max_depth , serta metode *Balancing* data mempengaruhi performa model *Random Forest*, di mana penggunaan SMOTE dan max_depth yang lebih tinggi terbukti memberikan hasil yang lebih baik pada sebagian besar metrik evaluasi.

4.3 Hasil Uji Coba Algoritma *Adaboost*

Pada model ini menggunakan algoritma *Adaboost*. Pada pengujian algoritma ini dilakukan normalisasi *Z-Score*, normalisasi *Min-Max*, *Balancing*

menggunakan SMOTE dan tanpa menggunakan *Balancing*. Pada setiap skenario menggunakan kombinasi parameter yaitu $n_estimator$ (30, 50, dan 100).

4.3.1 Model 25

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 30$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.25.

Tabel 4. 23 *Confusion matrix* Uji Coba Ke-25

TP	FP	TN	FN
70	6	56	4

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 92%, *recall* sebesar 95%, dan *f1-score* sebesar 93%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 93%, *recall macro average* sebesar 92%, dan *f1-score macro average* sebesar 93%.

4.3.2 Model 26

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 50$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.26.

Tabel 4. 24 *Confusion matrix* Uji Coba Ke-26

TP	FP	TN	FN
72	3	59	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 96%, presisi sebesar 96%, *recall* sebesar 97%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 96%, *recall macro average* sebesar 96%, dan *f1-score macro average* sebesar 96%.

4.3.3 Model 27

Pada uji coba model ini menggunakan normalisasi *Z-Score*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 70$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.27.

Tabel 4. 25 *Confusion matrix* Uji Coba Ke-27

TP	FP	TN	FN
71	4	58	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 95%, presisi sebesar 95%, *recall* sebesar 96%, dan *f1-score* sebesar 95%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 95%, *recall macro average* sebesar 95%, dan *f1-score macro average* sebesar 95%.

4.3.4 Model 28

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 30$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.28.

Tabel 4. 26 *Confusion matrix* Uji Coba Ke-28

TP	FP	TN	FN
65	6	9	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 86%, presisi sebesar 87%, *recall* sebesar 98%, dan *f1-score* sebesar 92%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 85%, *recall macro average* sebesar 66%, dan *f1-score macro average* sebesar 70%.

4.3.5 Model 29

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 50$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.29.

Tabel 4. 27 *Confusion matrix* Uji Coba Ke-29

TP	FP	TN	FN
65	5	10	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 93%, *recall* sebesar 98%, dan *f1-score* sebesar 96%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 92%, *recall macro average* sebesar 83%, dan *f1-score macro average* sebesar 86%.

4.3.6 Model 30

Pada uji coba model ini menggunakan normalisasi *Z-Score* dan menggunakan parameter $n_estimator = 70$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.30.

Tabel 4. 28 *Confusion matrix* Uji Coba Ke-30

TP	FP	TN	FN
65	5	10	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 91%, *recall* sebesar 98%, dan *f1-score* sebesar 96%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 92%, *recall macro average* sebesar 83%, dan *f1-score macro average* sebesar 86%.

4.3.7 Model 31

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 30$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.31.

Tabel 4. 29 *Confusion matrix* Uji Coba Ke-31

TP	FP	TN	FN
72	3	59	2

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 96%, presisi sebesar 96%, *recall* sebesar 97%, dan *f1-score* sebesar 97%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 96%, *recall macro average* sebesar 96%, dan *f1-score macro average* sebesar 96%.

4.3.8 Model 32

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 50$. Hasil

pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.32.

Tabel 4. 30 *Confusion matrix* Uji Coba Ke-32

TP	FP	TN	FN
71	5	57	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 94%, presisi sebesar 93%, *recall* sebesar 96%, dan *f1-score* sebesar 95%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 94%, *recall macro average* sebesar 94%, dan *f1-score macro average* sebesar 94%.

4.3.9 Model 33

Pada uji coba model ini menggunakan normalisasi *Min-Max*, *Balancing* menggunakan SMOTE dan menggunakan parameter $n_estimator = 70$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.33.

Tabel 4. 31 *Confusion matrix* Uji Coba Ke-33

TP	FP	TN	FN
71	6	56	3

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 92%, *recall* sebesar 96%, dan *f1-score* sebesar 94%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 94%, *recall macro average* sebesar 93%, dan *f1-score macro average* sebesar 93%.

4.3.10 Model 34

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 30$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.34.

Tabel 4. 32. *Confusion matrix* Uji Coba Ke-34

TP	FP	TN	FN
65	6	9	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 91%, presisi sebesar 92%, *recall* sebesar 98%, dan *f1-score* sebesar 95%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 91%, *recall macro average* sebesar 79%, dan *f1-score macro average* sebesar 83%.

4.3.11 Model 35

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 50$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.35.

Tabel 4. 33. *Confusion matrix* Uji Coba Ke-35

TP	FP	TN	FN
65	5	10	1

Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 93%, *recall* sebesar 98%, dan *f1-score* sebesar 96%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 92%, *recall macro average* sebesar 83%, dan *f1-score macro average* sebesar 86%.

4.3.12 Model 36

Pada uji coba model ini menggunakan normalisasi *Min-Max* dan menggunakan parameter $n_estimator = 70$. Hasil pengujian yang didapat sebagai berikut. *Confusion matrix* dari uji coba ini dapat dilihat pada Tabel 4.36.

Tabel 4. 34 *Confusion matrix* Uji Coba Ke-36

TP	FP	TN	FN
65	5	10	1

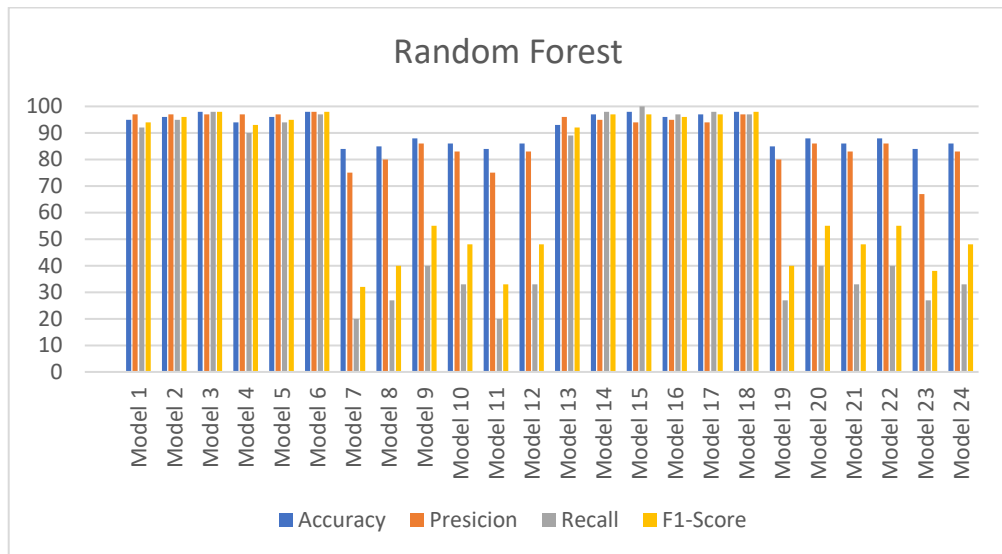
Berdasarkan hasil pengujian menggunakan rasio perbandingan data 8:2 menghasilkan akurasi sebesar 93%, presisi sebesar 93%, *recall* sebesar 98%, dan *f1-score* sebesar 96%. Selain itu terdapat beberapa hasil lain yang didapatkan meliputi presisi *macro average* sebesar 92%, *recall macro average* sebesar 83%, dan *f1-score macro average* sebesar 86%.

4.4 Pembahasan

Pada bagian ini menjabarkan hasil uji coba untuk mengetahui skenario yang paling optimal dalam melatih *Random Forest* dan *Adaboost*.

4.4.1 Analisa Hasil Uji Coba Algoritma *Random Forest*

Berdasarkan hasil uji coba model *Random Forest* pada Gambar 4.4, terdapat variasi performa pada metrik evaluasi accuracy, precision, recall, dan F1-score. Penggunaan metode normalisasi, teknik *Balancing*, serta variasi nilai $n_estimators$ dan max_depth memberikan dampak yang berbeda terhadap performa masing-masing model.



Gambar 4. 4 Grafik Hasil Uji Coba *Random Forest*

Pada Gambar 4.4 merupakan grafik hasil uji coba menggunakan metode *Random Forest*. Hasil evaluasi menunjukkan bahwa parameter yang paling berpengaruh terhadap akurasi, presisi, recall, dan F1-score adalah *max_depth* dan teknik *Balancing*. Peningkatan kedalaman pohon hingga tingkat tertentu (*max_depth = 10*) secara konsisten meningkatkan performa model. Selain itu, penggunaan SMOTE secara signifikan meningkatkan akurasi dan keseimbangan antara presisi dan recall, sehingga sangat direkomendasikan untuk dataset yang tidak seimbang. Teknik normalisasi memberikan dampak kecil, sedangkan peningkatan jumlah *n_estimators* memiliki pengaruh minimal terhadap hasil akhir model.

Berikut merupakan hasil analisa performa model *Random Forest* berdasarkan parameter yang digunakan:

A. Pengaruh *Balancing* pada Metode *Random Forest*

Balancing data merupakan langkah penting dalam menangani dataset yang tidak seimbang. Tabel berikut menunjukkan performa model *Random Forest* dengan *Balancing* SMOTE dan non-SMOTE.

Tabel 4. 35 Analisa pengaruh *Balancing* pada metode *Random Forest*

<i>Balancing</i>	Accuracy	Presisi	Recall	F1-Score
Non-smote	86	81	31	45
Smote	96	96	96	96

Dari Tabel 4.37, terlihat bahwa *Balancing* menggunakan SMOTE memberikan performa yang jauh lebih baik di semua metrik evaluasi dibandingkan dengan non-SMOTE. Teknik SMOTE berhasil meningkatkan nilai recall dari 31% menjadi 96%, yang berkontribusi signifikan pada peningkatan F1-score. Hal ini menunjukkan bahwa teknik SMOTE efektif dalam mengatasi ketidakseimbangan data, sehingga model mampu mendeteksi kelas positif dengan lebih baik.

B. Pengaruh Normalisasi pada Metode *Random Forest*

Normalisasi data bertujuan untuk menyelaraskan skala antar fitur dalam dataset. Tabel berikut membandingkan performa model dengan normalisasi *Z-Score* dan *Min-Max*.

Tabel 4. 36 Analisa pengaruh Normalisasi pada metode *Random Forest*

Normalisasi	Accuracy	Presisi	Recall	F1-Score
<i>Z-Score</i>	91	89	62	69
<i>Min-Max</i>	92	88	68	74

Tabel 4.38 menunjukkan bahwa normalisasi *Min-Max* menghasilkan performa yang lebih baik dibandingkan *Z-Score*, terutama pada nilai recall dan F1-score. Normalisasi *Min-Max* meningkatkan recall dari 62% menjadi 68%, yang berdampak pada peningkatan F1-score menjadi 74%. Hal ini menunjukkan bahwa *Min-Max* lebih sesuai untuk skala data pada model *Random Forest* dibandingkan *Z-Score*.

C. Pengaruh Jumlah *n_estimator* pada Metode *Random Forest*

Parameter *n_estimator* menentukan jumlah pohon dalam ensemble *Random Forest*. Tabel berikut menunjukkan performa model berdasarkan jumlah *n_estimator*.

Tabel 4. 37 Analisa pengaruh *n_estimator* pada metode *Random Forest*

n_estimator	Accuracy	Presisi	Recall	F1-Score
50	91	89	63	70
100	91	88	63	71

Tabel 4.39 menunjukkan bahwa peningkatan *n_estimator* dari 50 menjadi 100 memberikan sedikit peningkatan pada F1-score, yaitu dari 70% menjadi 71%. Hal ini mengindikasikan bahwa lebih banyak pohon dalam ensemble membantu model menghasilkan prediksi yang lebih konsisten, meskipun peningkatan performa relatif kecil.

D. Pengaruh *max_depth* pada Metode *Random Forest*

Parameter *max_depth* menentukan kedalaman maksimum pohon dalam *Random Forest*. Tabel berikut menunjukkan pengaruhnya terhadap performa model.

Tabel 4. 38 Analisa pengaruh max_depth pada metode *Random Forest*

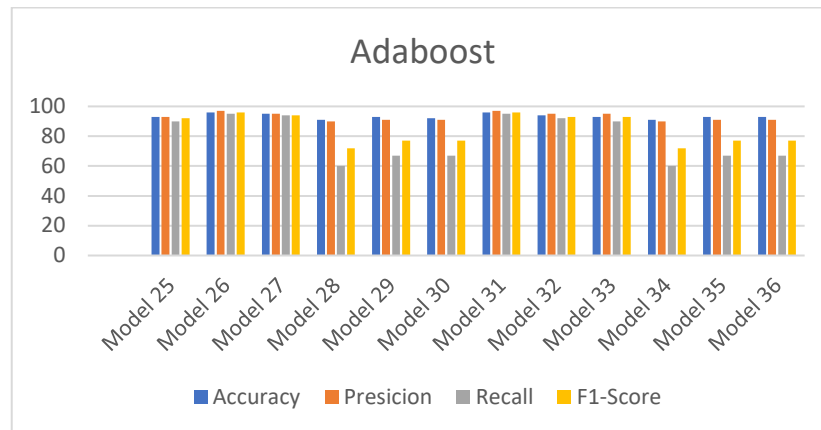
Max_depth	Accuracy	Presisi	Recall	F1-Score
5	90	89	61	69
7	91	86	62	69
10	92	90	67	74

Tabel 4.40 terlihat bahwa peningkatan max_depth dari 5 hingga 10 memberikan dampak positif pada performa model. Kedalaman maksimum pohon sebesar 10 menghasilkan nilai F1-score tertinggi (74%), menunjukkan bahwa kedalaman pohon yang lebih besar memungkinkan model menangkap lebih banyak pola dalam data. Namun, kedalaman yang terlalu besar dapat meningkatkan risiko *overfitting*, sehingga parameter ini perlu disesuaikan dengan kompleksitas dataset.

Secara keseluruhan, hasil ini menunjukkan bahwa pemilihan parameter n_estimators, max_depth, serta metode *Balancing* data sangat mempengaruhi performa model *Random Forest*, di mana penggunaan SMOTE dan n_estimators yang lebih tinggi terbukti memberikan hasil yang lebih baik pada sebagian besar metrik evaluasi.

4.4.2 Analisa Hasil Uji Coba Algoritma *Adaboost*

Berdasarkan hasil uji coba model *Adaboost* yang ditampilkan pada Gambar 4.5, terlihat bahwa performa metrik evaluasi yaitu *accuracy*, *precision*, *recall*, dan *F1-score* mengalami variasi yang relatif stabil di sebagian besar model. Model dengan kombinasi *Balancing* menggunakan SMOTE dan non-SMOTE serta nilai n_estimators yang berbeda menunjukkan pengaruh yang cukup konsisten terhadap performa keseluruhan.



Gambar 4. 5 Grafik Hasil Uji Coba *Adaboost*

Pada Gambar 4.5 menunjukkan grafik hasil uji coba metode *Adaboost*. Dari penelitian ini menunjukkan bahwa metode *Adaboost* dengan parameter $n_estimators = 50$ memberikan hasil performa yang optimal. Penerapan teknik *Balancing* seperti SMOTE terbukti efektif dalam meningkatkan keseimbangan antara presisi dan recall, sehingga memberikan dampak signifikan terhadap performa model. Di sisi lain, penggunaan teknik normalisasi sama-sama menghasilkan yang optimal.

Berikut merupakan hasil analisa performa model *Adaboost* berdasarkan parameter yang digunakan:

A. Pengaruh Normalisasi pada Metode *Adaboost*

Normalisasi bertujuan untuk menyelaraskan skala antar fitur dalam dataset. Tabel berikut membandingkan performa model menggunakan normalisasi *Z-Score* dan *Min-Max*:

Tabel 4. 39 Analisa pengaruh Normalisasi pada metode *Adaboost*

Normalisasi	Akurasi	Precision	Recall	F1-Score
<i>Min-Max</i>	93	93	79	85
<i>Z-Score</i>	93	93	79	85

Hasil Tabel 4.41 menunjukkan bahwa normalisasi *Z-Score* dan *Min-Max* memberikan performa yang serupa pada semua metrik evaluasi. Hal ini menunjukkan bahwa kedua metode normalisasi ini sama-sama cocok untuk dataset yang digunakan dalam model *Adaboost*.

B. Pengaruh *Balancing* pada metode *Adaboost*

Balancing data adalah langkah krusial dalam menangani dataset yang tidak seimbang. Tabel berikut menunjukkan performa model *Adaboost* dengan *Balancing* SMOTE dan tanpa *Balancing* (non-SMOTE).

Tabel 4. 40 Analisa pengaruh *Balancing* pada metode *Adaboost*

Normalisasi	Akurasi	Precision	Recall	F1-Score
smote	95	95	93	94
non-smote	92	91	65	75

Dari Tabel 4.43, terlihat bahwa *Balancing* menggunakan SMOTE memberikan peningkatan signifikan pada nilai recall, dari 65% menjadi 93%. Hal ini berdampak pada peningkatan F1-score dari 75% menjadi 94%. Dengan demikian, SMOTE terbukti efektif dalam mengatasi ketidakseimbangan data, membuat model lebih mampu mendeteksi kelas minoritas tanpa mengorbankan metrik lainnya.

C. Pengaruh *n_estimator* pada Metode *Adaboost*

Parameter *n_estimators* menentukan jumlah iterasi boosting dalam model *Adaboost*. Tabel berikut menunjukkan performa model berdasarkan jumlah *n_estimators*:

Tabel 4. 41 Analisa pengaruh n_estimator pada metode *Adaboost*

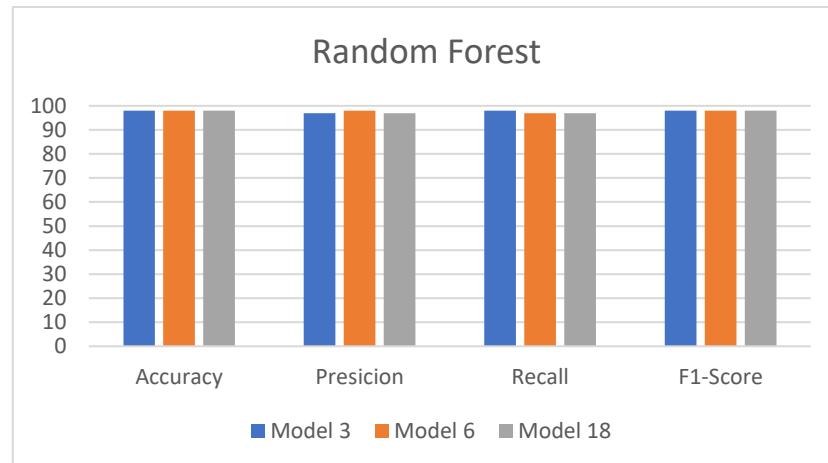
Normalisasi	Akurasi	Precision	Recall	F1-Score
30	92,75	92,5	76,25	83
50	94	93,5	80,25	85,75
70	93,25	93	79,5	85,25

Dari Tabel 4.43 terlihat bahwa peningkatan jumlah n_estimators dari 30 menjadi 50 memberikan peningkatan pada recall dan F1-score. Namun, peningkatan lebih lanjut menjadi 70 tidak memberikan keuntungan yang signifikan, menunjukkan bahwa jumlah n_estimators sebesar 50 sudah optimal untuk model ini.

Secara keseluruhan performa model *Adaboost* sangat dipengaruhi oleh *Balancing* data, normalisasi, dan jumlah n_estimators. *Balancing* menggunakan SMOTE secara signifikan meningkatkan recall dari 65% menjadi 93% dan F1-score dari 75% menjadi 94%, menunjukkan bahwa SMOTE efektif dalam menangani ketidakseimbangan data. Normalisasi *Z-Score* dan *Min-Max* memberikan hasil yang sama, dengan recall dan F1-score masing-masing sebesar 79% dan 85%. Untuk parameter n_estimators, hasil paling optimal dicapai pada nilai 50 dengan F1-score 85.75%, sesuai dengan default *Adaboost* yang menggunakan 50 iterasi, menunjukkan bahwa jumlah ini sudah cukup untuk mencapai keseimbangan antara akurasi dan efisiensi komputasi.

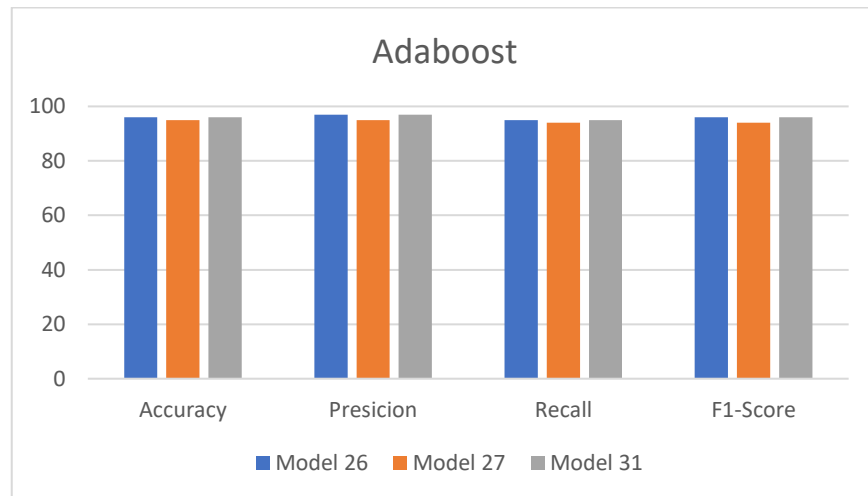
4.3.3 Perbandingan *Random Forest* dan *Adaboost*

Berikut merupakan analisis perbandingan performa tiga model terbaik dari metode *Random Forest* dan *Adaboost* berdasarkan empat metrik utama: accuracy, precision, recall, dan F1-score.



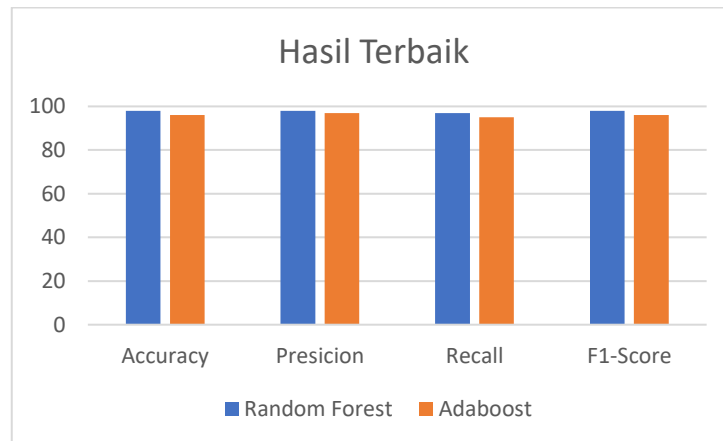
Gambar 4. 6 3 model terbaik *Random Forest*

Gambar 4.6 menampilkan performa dari 3 model terbaik pada metode *Random Forest*, yaitu Model 3, Model 6, dan Model 18. Hasil evaluasi performa menunjukkan bahwa ketiga model *Random Forest* memiliki akurasi tinggi, yaitu 98%. Model 6 unggul dalam presisi (98%), menunjukkan kemampuan terbaik dalam mengurangi false positive. Sementara itu, Model 3 memiliki recall tertinggi (98%), yang berarti lebih baik dalam mendeteksi kasus positif. Model 18 menunjukkan performa seimbang dengan nilai presisi dan recall masing-masing 97%, serta F1-Score 98% seperti model lainnya.



Gambar 4. 7 3 model terbaik *Adaboost*

Gambar 4.7 menunjukkan performa dari tiga model terbaik pada metode *Adaboost*, yaitu Model 26, Model 27, dan Model 31. Model-model ini memiliki accuracy sebesar 96%, sedikit lebih rendah dibandingkan dengan *Random Forest*. Precision pada Model 31 mencapai angka tertinggi (97.5%), sedangkan Model 27 stabil di angka 94.5%, menunjukkan bahwa Model 31 lebih efektif dalam menghindari prediksi positif yang salah. Pada metrik recall, Model 26 dan Model 31 sama-sama mencapai nilai 95%, menunjukkan bahwa keduanya cukup baik dalam mendeteksi kelas positif, meskipun performanya sedikit lebih rendah dibandingkan *Random Forest*. F1-score pada ketiga model ini berada di kisaran 95%-96%, mencerminkan performa *Adaboost* yang stabil meskipun tidak sebaik *Random Forest* dalam keseimbangan antara precision dan recall.



Gambar 4. 8 Hasil Terbaik *Random Forest* dan *Adaboost*

Dari Gambar 4.8 dapat disimpulkan bahwa metode *Random Forest* unggul di seluruh metrik utama dibandingkan dengan *Adaboost*, terutama pada metrik presisi, recall dan F1-score. Grafik ini mendukung kesimpulan bahwa *Random Forest* adalah metode yang lebih optimal untuk penelitian ini.

4.4.3 Analisa *Feature Importance*

Berikut ini Gambar 4.9 merupakan hasil feature importance yang telah diurutkan dari yang paling berpengaruh pada penelitian ini.

```

Feature Importance :
stab.glu: 0.1246
age: 0.0713
chol: 0.0667
bp.1s: 0.0660
height: 0.0659
hip: 0.0609
hdl: 0.0603
bp.1d: 0.0588
waist: 0.0583
time.ppn: 0.0557
ratio: 0.0535
weight: 0.0535
bp.2s: 0.0508
frame: 0.0468
bp.2d: 0.0414
location: 0.0385
gender: 0.0270

```

Gambar 4. 9 Hasil *Feature Importance*

Pada penelitian yang dilakukan (Susilawati & Rahmawati, 2021), fitur usia dinyatakan signifikan dalam memengaruhi kejadian diabetes mellitus tipe 2. Dalam penelitian ini, hasil perhitungan feature importance menunjukkan bahwa fitur usia memiliki nilai importance yang cukup tinggi, berada di peringkat kedua setelah "stab.glu" (kadar glukosa stabil) dengan nilai sebesar 0.0713. Ini menunjukkan bahwa usia merupakan salah satu faktor yang signifikan dalam model klasifikasi diabetes yang digunakan dalam penelitian ini, meskipun masih berada di bawah fitur glukosa.

Keberadaan usia sebagai fitur dengan importance yang cukup tinggi ini menunjukkan bahwa faktor usia memiliki kontribusi yang kuat dalam klasifikasi diabetes pada model ini, konsisten dengan temuan dari penelitian tersebut. Namun, faktor glukosa tetap menjadi fitur yang paling dominan, yang mungkin disebabkan oleh hubungan fisiologis langsung antara kadar glukosa dan diabetes.

Dari hasil penelitian yang telah dilakukan, perbandingan antara algoritma *Random Forest* dan *Adaboost* dalam klasifikasi diabetes menunjukkan adanya perbedaan dalam efektivitas keduanya. Setiap algoritma memiliki kelebihan dan kekurangannya masing-masing dalam hal akurasi, ketelitian, dan kemampuan mengidentifikasi faktor-faktor yang mempengaruhi klasifikasi diabetes.

Pemilihan algoritma yang tepat mencerminkan pentingnya kesesuaian antara metode yang digunakan dengan data yang dianalisis. Sebagaimana firman Allah SWT dalam Q.S. Al-Anbiya ayat 33:

وَهُوَ الَّذِي خَلَقَ اللَّيْلَ وَالنَّهَارَ وَالشَّمْسَ وَالْقَمَرَ كُلٌّ فِي فَلَكٍ يَسْبَحُونَ ﴿٣٣﴾

“Dan Dialah yang telah menciptakan malam dan siang, matahari dan bulan; semua itu beredar di dalam falaknya masing-” (QS. Al Anbiya: 33).

Menurut Tafsir As-Sa'di, ayat ini menjelaskan bahwa Allah telah memelihara matahari dan bintang, masing masing beredar dalam garis edar yang ditentukan (Ash-Shiddieqy, 2011). Dalam konteks penelitian ini, konsep keteraturan dan keseimbangan tersebut menjadi inspirasi dalam proses pemilihan algoritma dan pengaturan parameter. Sama seperti orbit yang diatur dengan teliti oleh Allah SWT, pemilihan metode seperti *Random Forest* dan *Adaboost*, serta penyesuaian parameter yang optimal, dilakukan untuk menghasilkan keseimbangan antara kompleksitas model dan akurasi hasil. Dengan pendekatan yang sistematis dan terukur, diharapkan model yang dibangun dapat meningkatkan keakuratan dalam klasifikasi diabetes, sehingga mendukung tujuan penelitian secara menyeluruh. Hal ini mengajarkan pentingnya kehati-hatian dalam setiap proses untuk mencapai hasil yang optimal dan bermanfaat.

Hal ini juga sejalan dengan firman Allah dalam Q.S. Al-Furqan ayat 2:

الَّذِي لَهُ مُلْكُ السَّمَوَاتِ وَالْأَرْضِ وَمَآ يَتَّخِذُ وَلَدًا ۚ وَمَآ يَكُنْ لَهُ شَرِيكٌ فِي الْمُلْكِ وَخَلَقَ كُلَّ شَيْءٍ

فَقَدَرَهُ ۗ تَقْدِيرًا ﴿٢﴾

“Dia telah menciptakan segala sesuatu, lalu menetapkan ukuran-ukurannya dengan tepat.” (Q.S. Al-Furqan [25]: 2)

Menurut Tafsir Ibnu Katsir, ayat ini menegaskan bahwa Allah adalah satu-satunya pencipta segala sesuatu di langit dan bumi, serta menentukan ukuran-

ukurannya dengan sangat teliti dan presisi. Dalam penelitian ini, pengaturan parameter dan pemilihan algoritma yang tepat diharapkan dapat mencapai keseimbangan antara akurasi dan efisiensi dalam klasifikasi diabetes, sehingga hasil yang diperoleh menjadi lebih optimal (Pokhrel, 2024).

Ayat ini menegaskan bahwa segala sesuatu diciptakan dengan presisi dan keseimbangan. Dalam penelitian ini, pengaturan parameter dan pemilihan algoritma yang tepat diharapkan dapat mencapai keseimbangan antara akurasi dan efisiensi dalam klasifikasi diabetes, sehingga hasil yang diperoleh dapat lebih optimal. Penelitian ini diharapkan dapat menjadi pengingat bagi manusia akan pentingnya menjaga kesehatan secara berkelanjutan. Kesehatan merupakan aspek yang sangat berharga dalam kehidupan dan dalam Islam dianggap sebagai anugerah yang harus dihargai oleh setiap individu. Nabi Muhammad SAW mengingatkan kita tentang pentingnya kesehatan dan waktu luang, sebagaimana beliau bersabda:

نِعْمَتَانِ مَعْبُودٌ فِيهِمَا كَثِيرٌ مِنَ النَّاسِ الصِّحَّةُ وَالْفَرَاغُ

“Dua kenikmatan yang sering dilupakan oleh kebanyakan manusia adalah kesehatan dan waktu luang.” (HR. Al-Bukhari: 6412, at-Tirmidzi: 2304, Ibnu Majah: 4170)

Ibnu Bathal menjelaskan bahwa seseorang tidak dikatakan memiliki waktu luang hingga ia memiliki kesehatan yang baik. Apabila seseorang memiliki kedua hal tersebut, ia seharusnya lebih semangat dalam memanfaatkan nikmat tersebut dan tidak lalai dalam bersyukur atas karunia Allah SWT. Bersyukur diwujudkan dengan menjalankan segala perintah-Nya dan menjauhi larangan-Nya. Mereka

yang tidak melakukannya dianggap sebagai orang yang terperdaya (Asqalani, 2002).

Melalui penelitian klasifikasi penderita diabetes berdasarkan data rekam medis menggunakan metode *Random Forest* dan *Adaboost* ini, diharapkan manusia dapat merenungkan betapa berharganya nikmat kesehatan yang diberikan oleh Allah SWT. Kesehatan adalah salah satu nikmat terbesar yang diberikan-Nya kepada umat manusia, dan dengan diberikan nikmat tersebut, kita diingatkan untuk senantiasa bersyukur dengan cara menaati segala perintah-Nya serta menjauhi segala larangan-Nya.

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Penelitian ini bertujuan untuk mengevaluasi performa metode *Random Forest* dan *Adaboost* dalam klasifikasi penyakit diabetes, serta menganalisis pengaruh teknik *balancing data*, normalisasi, dan fitur penting seperti usia terhadap hasil klasifikasi. Berdasarkan hasil analisis, beberapa kesimpulan utama dapat disampaikan sebagai berikut:

1. Perbandingan Performa Metode *Random Forest* dan *Adaboost*

Berdasarkan hasil penelitian ini dapat disimpulkan bahwa metode *Random Forest* secara keseluruhan menunjukkan performa yang lebih unggul sebesar 2% dibandingkan *Adaboost*. *Random Forest* mencapai akurasi hingga 98%, sedangkan *Adaboost* mencapai 96%. *Random Forest* lebih efektif menghasilkan performa dibandingkan dengan *Adaboost*, meskipun memiliki performa yang stabil.

2. Pengaruh Normalisasi dan *Balancing Data* pada *Random Forest* dan *Adaboost*

Penggunaan teknik *Balancing data* seperti *SMOTE* terbukti sangat efektif dalam meningkatkan performa model, khususnya pada *Random Forest*. Teknik normalisasi juga memberikan pengaruh signifikan, di mana normalisasi *Min-Max* lebih efektif dibandingkan *Z-Score* dalam meningkatkan *recall* dan *F1-score*, terutama pada *Random Forest*. Parameter model seperti *max_depth* yang optimal pada nilai 10 memberikan performa terbaik untuk *Random Forest*,

sedangkan peningkatan jumlah pohon (*n_estimators*) tidak mempengaruhi pada matrix evaluasi. Pada *Adaboost*, SMOTE juga berhasil dalam meningkatkan performa model, dengan normalisasi *Min-Max* dan *Z-Score* memberikan hasil serupa yang artinya jika menggunakan *Balancing* akan meningkatkan peforma model. Parameter jumlah iterasi (*n_estimators*) yang optimal pada nilai 50 menunjukkan keseimbangan antara akurasi dan efisiensi komputasi pada model *Adaboost*.

3. Pengaruh Fitur Usia dalam Klasifikasi

Dari analisis *feature importance*, fitur "stab.glu" (kadar glukosa stabil) dan usia memiliki pengaruh terbesar dalam menentukan hasil klasifikasi. Hal ini mendukung penelitian sebelumnya yang menunjukkan bahwa kadar glukosa dan usia adalah faktor utama dalam prediksi diabetes.

Secara keseluruhan, metode *Random Forest* yang dikombinasikan dengan teknik *Balancing* SMOTE dan normalisasi *Min-Max* merupakan pendekatan terbaik untuk klasifikasi penyakit diabetes berdasarkan dataset penelitian ini.

5.2 Saran

Berdasarkan proses dan hasil penelitian ini, peneliti menyadari bahwa penelitian ini masih banyak kekurangan. Maka dari itu, diharapkan pada penelitian selanjutnya agar dapat melakukan peningkatan dan perbaikan agar hasil yang didapatkan menjadi lebih baik. Beberapa saran untuk penelitian selanjutnya yaitu:

1. Penggunaan dataset yang lain dari sumber yang berbeda agar dapat menilai kemampuan model dalam mengklasifikasikan pada populasi yang berbeda.

2. Penggunaan *Balancing* yang berbeda seperti *Random Oversampling*, *Random Undersampling*, ADASYN (*Adaptive Synthetic Sampling*) dan metode lainnya.
3. Eksperimen dengan metode yang lain seperti *Naïve Bayes*, *Support Vector Machine* (SVM), dan algoritma lainnya.

DAFTAR PUSTAKA

- Abdurrahman, G. (2022). Klasifikasi Penyakit Diabetes Melitus Menggunakan Adaboost Classifier. *JUSTINDO (Jurnal Sistem Dan Teknologi Informasi Indonesia)*, 7(1), 59–66. <https://doi.org/10.32528/justindo.v7i1.4949>
- Arania, R., Triwahyuni, T., Esfandiari, F., & Nugraha, F. R. (2021). HUBUNGAN ANTARA USIA, JENIS KELAMIN, DAN TINGKAT PENDIDIKAN DENGAN KEJADIAN DIABETES MELLITUS DI KLINIK MARDI WALUYO LAMPUNG TENGAH. *Jurnal Medika Malahayati*, 5(3), 146–153. <https://doi.org/10.33024/jmm.v5i3.4200>
- Ash-Shiddieqy, T. M. H. (2011). *Tafsir AL-Quranul Majid An-Nur Jilid 3*. https://books.google.co.id/books?id=7Ks_EAAAQBAJ&newbks=1&newbks_redir=0&lpg=PA238&dq=tafsir surat al furqan&hl=id&pg=PR6#v=onepage&q=tafsir surat al furqan&f=false
- Asqalani, I. H. Al. (2002). *Fathul Baari syarah : Sahih Al-Bukhari*.
- Cahyanti, F. L. D., Sarasati, F., Astuti, W., & Firasari, E. (2023). KLASIFIKASI DATA MINING DENGAN ALGORITMA MACHINE LARNING UNTUK PREDIKSI PENYAKIT LIVER. *Technologia : Jurnal Ilmiah*, 14(2), 134. <https://doi.org/10.31602/tji.v14i2.10093>
- Candra Permana, B. A., & Dewi Patwari, I. K. (2021). Komparasi Metode Klasifikasi Data Mining Decision Tree dan Naïve Bayes Untuk Prediksi Penyakit Diabetes. *Infotek : Jurnal Informatika Dan Teknologi*, 4(1), 63–69. <https://doi.org/10.29408/jit.v4i1.2994>
- Chen, R. C., Dewi, C., Huang, S. W., & Caraka, R. E. (2020). Selecting critical features for data classification based on machine learning methods. *Journal of Big Data*, 7(1). <https://doi.org/10.1186/s40537-020-00327-4>
- Depari, D. H., Widiastiwi, Y., & Santoni, M. M. (2022). Perbandingan Model Decision Tree, Naive Bayes dan *Random Forest* untuk Prediksi Klasifikasi Penyakit Jantung. *Informatik : Jurnal Ilmu Komputer*, 18(3), 239. <https://doi.org/10.52958/iftk.v18i3.4694>
- Dewi, P., Azizah, M., Rendowaty, A., Sri Wahyuni, Y., & Pranata, L. (2023). Edukasi tentang Diabetes Mellitus dan Pemeriksaan Biomedis Kadar Gula Darah Pada Ibu Rumah Tangga. *Health Community Service*, 1(1), 46–50. <https://doi.org/10.47709/hcs.v1i1.3358>
- Diana, R., Warni, H., & Sutabri, T. (2023). PENGGUNAAN TEKNOLOGI MACHINE LEARNING UNTUK PELAYANAN MONITORING KEGIATAN BELAJAR MENGAJAR PADA SMK BINA SRIWIJAYA PALEMBANG. *JUTEKIN (Jurnal Teknik Informatika)*, 11(1). <https://doi.org/10.51530/jutekin.v11i1.709>
- Ding, Y., Zhu, H., Chen, R., & Li, R. (2022). An Efficient AdaBoost Algorithm

with the Multiple Thresholds Classification. *Applied Sciences*, 12(12), 5872. <https://doi.org/10.3390/app12125872>

Erdiansyah, U., Irmansyah Lubis, A., & Erwansyah, K. (2022). Komparasi Metode K-Nearest Neighbor dan *Random Forest* Dalam Prediksi Akurasi Klasifikasi Pengobatan Penyakit Kulit. *JURNAL MEDIA INFORMATIKA BUDIDARMA*, 6(1), 208. <https://doi.org/10.30865/mib.v6i1.3373>

Ferawati, F., & Hadi Sulisty, A. A. (2020). HUBUNGAN ANTARA KEJADIAN KOMPLIKASI DENGAN KUALITAS HIDUP PENDERITA DIABETES MELLITUS TIPE 2 PADA PASIEN PROLANIS DI WILAYAH KERJA PUSKESMAS DANDER. *Jurnal Ilmiah Keperawatan Stikes Hang Tuah Surabaya*, 15(2), 269–277. <https://doi.org/10.30643/jiksht.v15i2.80>

Gita, A. P., & Utomo, D. E. (2022). Literature Review: Hubungan Pengetahuan Terhadap Kepatuhan Diet Diabetes Militus. *MAHESA : Malahayati Health Student Journal*, 2(1), 25–32. <https://doi.org/10.33024/mahesa.v2i1.5818>

Gunawan, G., Rahmawati, A., Suhada, S., Hidayatulloh, T., & Wintana, D. (2022). Optimasi Linear Sampling dan Information Gain pada Algoritma Decision Tree Untuk Diagnosis Penyakit Diabetes. *MULTINETICS*, 7(2), 124–131. <https://doi.org/10.32722/multinetics.v7i2.3796>

Gunawan, S., & Rahmawati, R. (2021). Hubungan Usia, Jenis Kelamin dan Hipertensi dengan Kejadian Diabetes Mellitus Tipe 2 di Puskesmas Tugu Kecamatan Cimanggis Kota Depok Tahun 2019. *ARKESMAS (Arsip Kesehatan Masyarakat)*, 6(1), 15–22. <https://doi.org/10.22236/arkesmas.v6i1.5829>

Harmaja, O. J., Prasetia, I., Hutagalung, Y. V., & Sirait, H. A. (2023). Comparison of Ensemble Learning Algorithm in Classifying Early Diagnostic of Diabetes. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima (JUSIKOM PRIMA)*, 7(1), 218–231. <https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v7i1.4054>

Hornyák, O., & Iantovics, L. B. (2023). AdaBoost Algorithm Could Lead to Weak Results for Data with Certain Characteristics. *Mathematics*, 11(8), 1801. <https://doi.org/10.3390/math11081801>

Jackins, V., Vimal, S., Kaliappan, M., & Lee, M. Y. (2021). AI-based smart prediction of clinical disease using *Random Forest* classifier and Naive Bayes. In *Journal of Supercomputing* (Vol. 77, Issue 5, pp. 5198–5219). <https://doi.org/10.1007/s11227-020-03481-x>

Jasman, T. Z., Fadhlullah, M. A., Pratama, A. L., & Rismayani, R. (2022). Analisis Algoritma Gradient Boosting, Adaboost dan Catboost dalam Klasifikasi Kualitas Air. *Jurnal Teknik Informatika Dan Sistem Informasi*, 8(2). <https://doi.org/10.28932/jutisi.v8i2.4906>

Jatnika Fahmi Idris, Rafid Ramadhani, & Muhammad Malik Mutoffar. (2024).

KLASIFIKASI PENYAKIT KANKER PARU MENGGUNAKAN PERBANDINGAN ALGORITMA MACHINE LEARNING. *Jurnal Media Akademik (JMA)*, 2(2). <https://doi.org/10.62281/v2i2.145>

Kari, T., N, L., A, S. B., R, D., Jagannatha, K. B., & Natarajan, S. (2021). An Accelerated Approach to Parallel Ensemble Techniques Targeting Healthcare and Environmental Applications. *2020 3rd International Conference on Energy, Power and Environment: Towards Clean Energy Technologies*, 1–6. <https://doi.org/10.1109/ICEPE50861.2021.9404519>

Lovic, D., Piperidou, A., Zografou, I., Grassos, H., Pittaras, A., & Manolis, A. (2020). The Growing Epidemic of Diabetes Mellitus. *Current Vascular Pharmacology*, 18(2), 104–109. <https://doi.org/10.2174/1570161117666190405165911>

Mahmuda, S. (2024). Implementasi Metode *Random Forest* pada Kategori Konten Kanal Youtube. *Jurnal Jendela Matematika*, 2(01), 21–31. <https://doi.org/10.57008/jjm.v2i01.633>

Manurung, R. F., & Darungan, T. S. (2021). GAMBARAN KUALITAS HIDUP PASIEN DIABETES MELITUS TIPE 2 PADA DI PUSKESMAS TELADAN KOTA MEDAN. *Jurnal Kedokteran Ibnu Nafis*, 10(2), 154–159. <https://doi.org/10.30743/jkin.v10i2.200>

Marsuhandi, A. H., Soleh, A. M., Wijayanto, H., & Domiri, D. D. (2020). PEMANFAATAN ENSEMBLE LEARNING DAN PENGINDERAAN JAUH UNTUK PENGKLASIFIKASIAN JENIS LAHAN PADI. *Seminar Nasional Official Statistics*, 2019(1), 188–195. <https://doi.org/10.34123/semnasoffstat.v2019i1.247>

Mohammed, A. J. (2020). Improving Classification Performance for a Novel Imbalanced Medical Dataset using SMOTE Method. *International Journal of Advanced Trends in Computer Science and Engineering*, 9(3), 3161–3172. <https://doi.org/10.30534/ijatcse/2020/104932020>

Mortara, A. A., Permatasari, M., Desiani, A., Andriani, Y., & Arhami, M. (2023). Perbandingan Algoritma C4.5 dan Adaptive Boosting dalam Klasifikasi Penyakit Alzheimer. *Jurnal Teknologi Dan Informasi*, 13(2), 196–207. <https://doi.org/10.34010/jati.v13i2.10525>

Nasution, F., Andilala, A., & Siregar, A. A. (2021). FAKTOR RISIKO KEJADIAN DIABETES MELLITUS. *Jurnal Ilmu Kesehatan*, 9(2), 94. <https://doi.org/10.32831/jik.v9i2.304>

Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *Jurnal Sains Komputer & Informatika (J-SAKTI)*, 5(2), 697–711.

Pokhrel, S. (2024). No TitleEΛENH. *Ayan*, 15(1), 37–48.

Potdar, K., S., T., & D., C. (2017). A Comparative Study of Categorical Variable

Encoding Techniques for Neural Network Classifiers. *International Journal of Computer Applications*, 175(4), 7–9. <https://doi.org/10.5120/ijca2017915495>

Purnajiwa Arimbawa, I. G. A., & Sanjaya ER, N. A. (2020). Penerapan Metode Adaboost Untuk Multi-Label Classification Pada Dokumen Teks. *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*, 9(1), 127. <https://doi.org/10.24843/JLK.2020.v09.i01.p13>

Rohman, M. A., & Arifianto, D. (2021). Penerapan Metode Euclidean Probability dan Confusion Matrix dalam Diagnosa Penyakit Koi Application of the Euclidean Probability and Confusion Matrix Methods in the Diagnosis of Koi Disease. *Jurnal Smart Teknologi*, 2(2), 122–130.

Sandrawati, S., & Somadayo, N. A. S. (2023). Penyuluhan Perubahan Gaya Hidup dan Pola Makan Yang Baik untuk Pencegahan Diabetes Mellitus Di Puskesmas Kalumata Kota Ternate. *Pekan : Jurnal Pengabdian Kesehatan*, 2(2), 10–14. <https://doi.org/10.33387/pekan.v2i2.7236>

Simarmata, K. B., Hartomo, K. D., & Hartomo, K. D. (2022). Analisa Rekomendasi Fitur Persetujuan Pinjaman Perusahaan Financial Technology Menggunakan Metode Random Forest. *JATISI (Jurnal Teknik Informatika Dan Sistem Informasi)*, 9(3), 2055–2070. <https://doi.org/10.35957/jatisi.v9i3.2258>

Simarmata, P. C., Hayati, K., Sitepu, S. D. E. U., Hutahuruk, R., Butar-butur, R. A., & Cahya, N. (2021). PENYULUHAN BRISK WALKING TERHADAP PENURUNAN KADAR GULA DARAH PADA PASIEN DIABETES MELITUS. *JURNAL PENGKAS KESTRA (JPK)*, 1(1), 106–109. <https://doi.org/10.35451/jpk.v1i1.743>

Suci Amaliah, Nusrang, M., & Aswi, A. (2022). Penerapan Metode *Random Forest* Untuk Klasifikasi Varian Minuman Kopi di Kedai Kopi Konijiwa Bantaeng. *VARIANSI: Journal of Statistics and Its Application on Teaching and Research*, 4(3), 121–127. <https://doi.org/10.35580/variansiunm31>

Sumwiza, K., Twizere, C., Rushingabigwi, G., Bakunzibake, P., & Bamurigire, P. (2023). Enhanced cardiovascular disease prediction model using *Random Forest* algorithm. *Informatics in Medicine Unlocked*, 41, 101316. <https://doi.org/10.1016/j.imu.2023.101316>

Syukron, A., & Subekti, A. (2018). Penerapan Metode Random Over-Under Sampling dan *Random Forest* Untuk Klasifikasi Penilaian Kredit. *Jurnal Informatika*, 5(2), 175–185. <https://doi.org/10.31311/ji.v5i2.4158>

T.Eltrikanawati, T. E., & Fedillah Nurhafifah, B. (2023). EDUKASI DIABETES MELITUS DAN PEMERIKSAAN KADAR GLUKOSA DARAH. *Sambulu Gana : Jurnal Pengabdian Masyarakat*, 2(2), 64–70. https://doi.org/10.56338/sambulu_gana.v2i2.3542

Tamba, S. P., & -, E. (2022). PREDIKSI PENYAKIT GAGAL JANTUNG

DENGAN MENGGUNAKAN RANDOM FOREST. *Jurnal Sistem Informasi Dan Ilmu Komputer Prima (JUSIKOM PRIMA)*, 5(2), 176–181. <https://doi.org/10.34012/jurnalsisteminformasidanilmukomputer.v5i2.2445>

Telaumbanua, F. D., Hulu, P., Nadeak, T. Z., Lumbantong, R. R., & Dharma, A. (2020). Penggunaan Machine Learning Di Bidang Kesehatan. *Jurnal Teknologi Dan Ilmu Komputer Prima (JUTIKOMP)*, 2(2). <https://doi.org/10.34012/jutikomp.v2i2.657>

Wang, W., & Sun, D. (2021). The improved AdaBoost algorithms for imbalanced data classification. *Information Sciences*, 563, 358–374. <https://doi.org/10.1016/j.ins.2021.03.042>

Wiji, N., Futihat, R. A., & Ahamd, M. S. (2023). LIVING QUR'AN DI RUMAH SAKIT ISLAM PURWOKERTO (Implementasi Surat Asy-Syu'ara Ayat 80 dalam Pelayanan Divisi Bimbingan Rohani Islam). *Qudwah Qur'aniyah : Jurnal Ilmu Al-Qur'an Dan Tafsir*, 1, 29–44.