PROJEK AKHIR UAS

BIG DATA AND DATA MINING (ST168)

KLASIFIKASI PENYAKIT ALZHEIMER MENGGUNAKAN ALGORITMA EXTREME GRADIENT BOOSTING

Dosen Pengampu:

Abd. Mizwar A. Rahim, M.Kom

Disusun oleh:

22.11.5260

M. IKHSAN

BDDM 4

PROGRAM STUDI S1 INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS AMIKOM YOGYAKARTA

2025

1. PENDAHULUAN

Topik klasifikasi penyakit Alzheimer menggunakan algoritma Extreme Gradient Boosting (XGBoost) sangat penting untuk diteliti karena Alzheimer merupakan salah satu penyebab utama demensia di seluruh dunia, yang berdampak signifikan pada individu, keluarga, dan sistem kesehatan masyarakat. Menurut data dari Organisasi Kesehatan Dunia (WHO), jumlah penderita demensia diperkirakan akan mencapai 152 juta pada tahun 2050, dengan Alzheimer sebagai penyebab utama[1]. Penelitian yang berfokus pada pengembangan metode klasifikasi yang lebih akurat dan efisien dapat membantu dalam diagnosis dini, yang sangat penting untuk intervensi terapeutik yang lebih efektif[2].

Penggunaan algoritma pembelajaran mesin, seperti XGBoost, dalam klasifikasi Alzheimer menawarkan potensi untuk meningkatkan akurasi diagnosis dengan memanfaatkan data yang kompleks, seperti citra MRI dan informasi biomarker. Penelitian menunjukkan bahwa model pembelajaran mesin dapat mengidentifikasi pola yang tidak terlihat oleh analisis tradisional, sehingga memungkinkan deteksi dini yang lebih baik[3]. Sebagai contoh, pendekatan berbasis pembelajaran mendalam telah menunjukkan hasil yang menjanjikan dalam segmentasi hippocampus dan klasifikasi citra MRI, yang merupakan area penting dalam diagnosis Alzheimer[4], [5]. Dengan memanfaatkan teknik seperti XGBoost, penelitian ini dapat mengoptimalkan pemilihan fitur dan meningkatkan sensitivitas serta spesifisitas dalam klasifikasi penyakit[6].

Dampak dari penelitian ini tidak hanya terbatas pada peningkatan akurasi diagnosis, tetapi juga pada pengurangan beban ekonomi dan sosial yang ditimbulkan oleh penyakit Alzheimer. Biaya perawatan untuk pasien Alzheimer sangat tinggi, dan diagnosis yang lebih cepat dapat mengurangi biaya tersebut dengan memungkinkan perencanaan perawatan yang lebih baik dan lebih awal[7]. elain itu, dengan meningkatkan pemahaman tentang mekanisme penyakit melalui analisis data yang lebih mendalam, penelitian ini dapat berkontribusi pada pengembangan terapi baru dan strategi pencegahan[8], [9].

Secara keseluruhan, penelitian tentang klasifikasi Alzheimer menggunakan algoritma XGBoost adalah langkah penting dalam upaya untuk mengatasi tantangan yang ditimbulkan oleh penyakit ini. Dengan memanfaatkan kemajuan dalam teknologi pembelajaran mesin dan neuroimaging, kita dapat berharap untuk mencapai kemajuan signifikan dalam diagnosis dan manajemen Alzheimer, yang pada gilirannya dapat meningkatkan kualitas hidup pasien dan mengurangi dampak sosial dari penyakit ini[10].

I. Tujuan Penelitian

1. Mengembangkan model klasifikasi berbasis machine learning dengan algoritma Extreme Gradient Boosting untuk klasifikasi Alzheimer berdasarkan fitur-fitur relevan yang telah dipilih.
2. Mengatasi ketidakseimbangan data dengan teknik machine learning untuk hasil prediksi yang lebih akurat.
3. Mengeksplorasi bagaimana pemrosesan data kompleks dengan algoritma XGBoost dapat meningkatkan deteksi dini Alzheimer.
4. Membuktikan efektivitas model XGBoost dalam aplikasi data medis lainnya sebagai dasar untuk pengembangan sistem diagnostik berbasis machine learning yang lebih luas.

II. Metode

Metode yang digunakan adalah metode eksperimental berbasis data sekunder dengan pendekatan machine learning. Fokus penelitian ini adalah pada pengembangan dan evaluasi model klasifikasi berbasis algoritma Extreme Gradient Boosting (XGBoost) untuk mendeteksi penyakit Alzheimer. Langkah-langkah utama dalam metode ini meliputi:

1. Pengumpulan Data
   Dataset medis terkait Alzheimer diperoleh dari sumber sekunder, yaitu diperoleh dari platform Kaggle.
2. Preprocessing Data
   Preprocessing ini dilakukan untuk menjaga kualitas data. Dilakukan beberapa langkah utama, yaitu memeriksa tipe data untuk memastikan kesesuaian format, menghapus fitur yang tidak penting untuk meningkatkan efisiensi model, mengganti nama kolom agar lebih informatif, dan mengecek nilai missing value.
3. Exploratory Data Analysis (EDA)
   Analisis data eksplorasi dilakukan untuk memahami distribusi data, hubungan antar fitur, dan pola penting yang ada dalam dataset. Ada beberapa visualisasi data dalam proyek ini yaitu bar chart, pie chart, box plot, dan histogram untuk memberikan wawasan lebih mendalam tentang dataset.
4. Seleksi Fitur
   Tahap ini bertujuan memilih fitur-fitur paling relevan untuk meningkatkan akurasi dan efisiensi model. Metode seleksi berbasis algoritma Random Forest digunakan untuk menentukan fitur penting.
5. Modeling
   Model klasifikasi dikembangkan menggunakan algoritma Extreme Gradient Boosting (XGBoost). Proses ini melibatkan pelatihan model untuk mengidentifikasi pola dalam data dan menghasilkan prediksi.
6. Evaluasi Model
   Model yang dikembangkan dievaluasi menggunakan classification report dan confusion matrix. Classification report memberikan metrik penting seperti precision, recall, dan F1-score untuk masing-masing kelas. Sementara itu, confusion matrix digunakan untuk menganalisis hasil prediksi secara mendetail, menunjukkan jumlah True Positive, False Positive, True Negative, dan False Negative.

2. PROFILE DATASET

I. Sumber Dataset

Dataset yang digunakan dalam proyek ini diperoleh dari platform Kaggle, dikenal sebagai Alzheimer's Disease Dataset. Dataset ini tersedia secara publik, dibuat oleh Rabie El Kharoua, dataset ini terakhir diperbarui 7 bulan yang lalu. dan dapat diakses melalui tautan Kaggle berikut ini: https://www.kaggle.com/datasets/rabieelkharoua/alzheimers-disease-dataset.

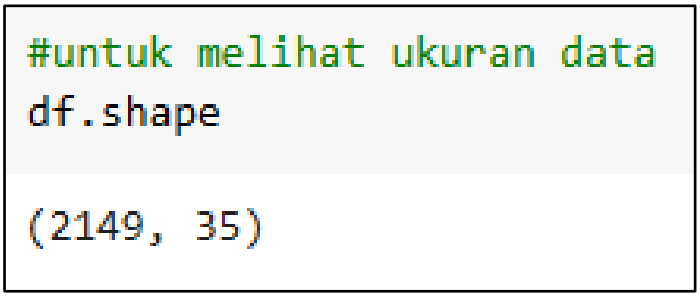
Dataset ini digunakan untuk melakukan analisis dan prediksi terkait penyakit Alzheimer, dengan fokus pada berbagai faktor yang memengaruhi kondisi tersebut. Dataset mencakup sejumlah parameter input, seperti informasi demografis usia dan jenis kelamin, riwayat kesehatan, gaya hidup pasien, dan faktor lain yang berhubungan dengan risiko Alzheimer. Kolom diagnosis dalam dataset ini menentukan apakah pasien terdiagnosis penyakit Alzheimer (1) atau tidak (0). Setiap baris data memuat informasi yang relevan tentang seorang pasien, yang dapat digunakan untuk analisis lanjutan dan pengembangan model prediktif. Berikut ini Tabel 1 yang menggambarkan informasi fitur pada dataset.

Tabel 1. Dataset

No	Fitur	Keterangan
1	PatientID	Identifikasi unik untuk setiap pasien
2	Age	Usia pasien
3	Gender	Jenis kelamin pasien (Male/Female)
4	Ethnicity	Ras atau etnis pasien
5	EducationLevel	Tingkat pendidikan pasien
6	BMI	Indeks massa tubuh pasien (kategori berat badan)
7	Smoking	Kebiasaan merokok (Yes/No)
8	AlcoholConsumption	Konsumsi alkohol (None/Moderate/High)
9	PhysicalActivity	Tingkat aktivitas fisik pasien (Active/Sedentary)
10	DietQuality	Kualitas diet pasien (Good/Average/Poor)
11	SleepQuality	Kualitas tidur pasien (Good/Average/Poor)
12	FamilyHistoryAlzheimers	Riwayat keluarga dengan penyakit Alzheimer (Yes/No)
13	CardiovascularDisease	Adanya penyakit kardiovaskular (Yes/No)
14	Diabetes	Adanya riwayat diabetes (Yes/No)
15	Depression	Adanya riwayat depresi (Yes/No)
16	HeadInjury	Riwayat cedera kepala (Yes/No)
17	Hypertension	Adanya hipertensi (Yes/No)
18	SystolicBP	Tekanan darah sistolik pasien (mmHg)
19	DiastolicBP	Tekanan darah diastolik pasien (mmHg)
20	CholesterolTotal	Total kolesterol pasien (mg/dL)
21	CholesterolLDL	Kadar kolesterol LDL pasien (mg/dL)
22	CholesterolHDL	Kadar kolesterol HDL pasien (mg/dL)
23	CholesterolTriglycerides	Kadar trigliserida pasien (mg/dL)
24	MMSE	Skor MMSE (Mini-Mental State Examination) pasien
25	FunctionalAssessment	Penilaian fungsional pasien (Good/Moderate/Poor)
26	MemoryComplaints	Keluhan terkait ingatan (Yes/No)
27	BehavioralProblems	Masalah perilaku pasien (Yes/No)
28	ADL	Tingkat kemandirian pasien dalam aktivitas sehari-hari
29	Confusion	Adanya kebingungan atau linglung (Yes/No)
30	Disorientation	Disorientasi waktu, tempat, atau orang (Yes/No)

31	PersonalityChanges	Perubahan kepribadian (Yes/No)
32	DifficultyCompletingTasks	Kesulitan menyelesaikan tugas sehari-hari (Yes/No)
33	Forgetfulness	Tingkat kelupaan pasien
34	Diagnosis	Hasil diagnosis pasien (1: Alzheimer, 0: Tidak Alzheimer)
35	DoctorInCharge	Dokter yang bertanggung jawab terhadap pasien

Dataset ini terdiri dari 2.149 sampel data yang mencakup 35 fitur, yang dirancang untuk memberikan informasi penting terkait kesehatan pasien dan gejala penyakit Alzheimer. Tipe data dalam dataset ini bervariasi, dengan mayoritas bertipe numerik int64 dan float64, yang mencerminkan data kuantitatif yang dapat diukur. Jumlah sampel data dan fitur dari dataset dapat dilihat pada gambar 1.



Gambar 1. Jumlah Data Pada Dataset

3. DATA PREPROCESSING

Proses pre-processing dilakukan untuk memastikan kualitas dataset yang akan digunakan dalam analisis. Berdasarkan evaluasi awal, pemeriksaan tipe data dilakukan untuk memastikan kesesuaian format data. Hal ini sangat krusial karena kesalahan dalam tipe data dapat mempengaruhi hasil model dan analisis yang dilakukan. Dengan memastikan bahwa setiap kolom memiliki tipe data yang sesuai, model dapat bekerja dengan optimal tanpa mengalami kesalahan interpretasi. Hasil tipe datanya dapat dilihat pada gambar 2.

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 2149 entries, 0 to 2148
Data columns (total 35 columns):
 #   Column                     Non-Null Count  Dtype
---  ------                     --------------  -----
 0   PatientID                  2149 non-null   int64
 1   Age                        2149 non-null   int64
 2   Gender                     2149 non-null   int64
 3   Ethnicity                  2149 non-null   int64
 4   EducationLevel             2149 non-null   int64
 5   BMI                        2149 non-null   float64
 6   Smoking                    2149 non-null   int64
 7   AlcoholConsumption         2149 non-null   float64
 8   PhysicalActivity           2149 non-null   float64
 9   DietQuality                2149 non-null   float64
 10  SleepQuality               2149 non-null   float64
 11  FamilyHistoryAlzheimers    2149 non-null   int64
 12  CardiovascularDisease      2149 non-null   int64
 13  Diabetes                   2149 non-null   int64
 14  Depression                 2149 non-null   int64
 15  HeadInjury                 2149 non-null   int64
 16  Hypertension               2149 non-null   int64
 17  SystolicBP                 2149 non-null   int64
 18  DiastolicBP                2149 non-null   int64
 19  CholesterolTotal           2149 non-null   float64
 20  CholesterolLDL             2149 non-null   float64
 21  CholesterolHDL             2149 non-null   float64
 22  CholesterolTriglycerides   2149 non-null   float64
 23  MMSE                       2149 non-null   float64
 24  FunctionalAssessment       2149 non-null   float64
 25  MemoryComplaints           2149 non-null   int64
 26  BehavioralProblems         2149 non-null   int64
 27  ADL                        2149 non-null   float64
 28  Confusion                  2149 non-null   int64
 29  Disorientation             2149 non-null   int64
 30  PersonalityChanges         2149 non-null   int64
 31  DifficultyCompletingTasks  2149 non-null   int64
 32  Forgetfulness              2149 non-null   int64
 33  Diagnosis                  2149 non-null   int64
 34  DoctorInCharge             2149 non-null   object
dtypes: float64(12), int64(22), object(1)
memory usage: 587.7+ KB
```

Gambar 2. Identifikasi Type Data pada Dataset

Selanjutnya, fitur yang tidak penting dihapus dari dataset. Proses ini bertujuan untuk meningkatkan efisiensi model dengan mengurangi kompleksitas data. Fitur yang dianggap tidak relevan yaitu ‘PatientID’ dan ‘DoctorInCharge’. Fitur ‘PatientID’ dianggap tidak penting karena fungsinya hanya sebagai pengidentifikasi unik untuk setiap pasien dan tidak memberikan informasi relevan terkait dengan diagnosis atau prediksi penyakit Alzheimer. Sedangkan fitur ‘DoctorInCharge’ dihilangkan karena tidak berkontribusi pada analisis data medis yang berkaitan dengan kondisi pasien, dan lebih merupakan informasi administratif yang tidak memengaruhi hasil klasifikasi model. Dengan menghapus kedua fitur ini, dataset menjadi lebih fokus pada variabel yang memiliki pengaruh langsung terhadap penyakit yang sedang diteliti. Kode untuk menghapus fitur ‘patienID’ dan ‘DoctorInCharge’ dapat dilihat pada Gambar 3.

```
# Drop fitur yang engga penting
df = df.drop(['PatientID', 'DoctorInCharge'], axis=1)
df.head(10)
```

Gambar 3. Identifikasi Type Data pada Dataset

Dalam tahap pre-processing ini, dilakukan juga untuk menggantikan nama fitur untuk meningkatkan interpretabilitas dataset. Bebera fitur diganti namanya agar lebih informatif dan mudah dipahami, dari yang awalannya huruf kapital diganti jadi huruf kecil, dan juga yang menggunakan spasi diganti pake '_'. Kode untuk mengganti nama kolom dapat dilihat pada Gambar 4.

```
# Mengganti nama kolom
df = df.rename(columns={
    'Age': 'age',
    'Gender': 'gender',
    'Ethnicity': 'ethnicity',
    'Smoking': 'smoking',
    'AlcoholConsumption': 'alcohol_consumption',
    'PhysicalActivity': 'physical_activity',
    'DietQuality': 'diet_quality',
    'SleepQuality': 'sleep_quality',
    'FamilyHistoryAlzheimers': 'family_history_AD',
    'CardiovascularDisease': 'cardiovascular_disease',
    'Diabetes': 'diabetes',
    'Depression': 'depression',
    'HeadInjury': 'head_injury',
    'Hypertension': 'hypertension',
    'SystolicBP': 'systolic_BP',
    'DiastolicBP': 'diastolic_BP',
    'CholesterolTotal': 'total_cholesterol',
    'CholesterolLDL': 'LDL_cholesterol',
    'CholesterolTriglycerides': 'Triglycerides',
    'FunctionalAssessment': 'functional_assessment',
    'MemoryComplaints': 'memory_complaints',
    'BehavioralProblems': 'behavioral_problems',
    'Confusion': 'confusion',
    'Disorientation': 'disorientation',
    'PersonalityChanges': 'personality_changes',
    'DifficultyCompletingTasks': 'task_completion_difficulty',
    'Forgetfulness': 'forgetfulness',
    'Diagnosis': 'diagnosis',
    'CholesterolHDL': 'HDl_cholesterol',
    'EducationLevel': 'education_level'
})

df.head()
```

Gambar 4. Kode Mengganti Nama Kolom

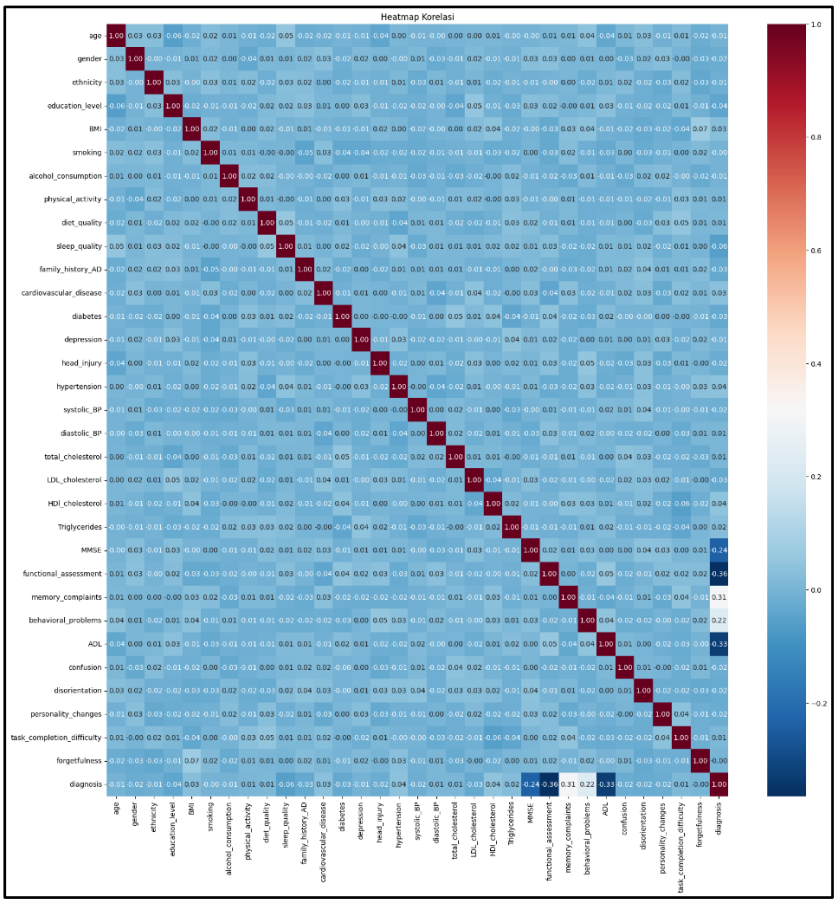
Terakhir, langkah pengecekan nilai missing value sangat penting untuk memastikan integritas data. Nilai yang hilang dapat mengganggu proses pelatihan model dan menghasilkan prediksi yang tidak akurat. Pada dataset ini, hasil pemeriksaan menunjukkan bahwa tidak terdapat missing value pada semua fitur. Hal ini memastikan bahwa data yang digunakan untuk pelatihan model sudah lengkap dan berkualitas, sehingga tidak memerlukan langkah imputasi atau penghapusan entri. Ketiadaan nilai yang hilang ini mendukung proses analisis lebih efisien dan memberikan dasar yang kuat untuk pengembangan model klasifikasi. Hasil identifikasi nilai kosong dapat dilihat pada Gambar 5.

	0
age	0
gender	0
ethnicity	0
education_level	0
BMI	0
smoking	0
alcohol_consumption	0
physical_activity	0
diet_quality	0
sleep_quality	0
family_history_AD	0
cardiovascular_disease	0
diabetes	0
depression	0
head_injury	0
hypertension	0
systolic_BP	0
diastolic_BP	0
total_cholesterol	0
LDL_cholesterol	0
HDL_cholesterol	0
Triglycerides	0
MMSE	0
functional_assessment	0
memory_complaints	0
behavioral_problems	0

Gambar 5. Identifikasi Missing Value pada Dataset

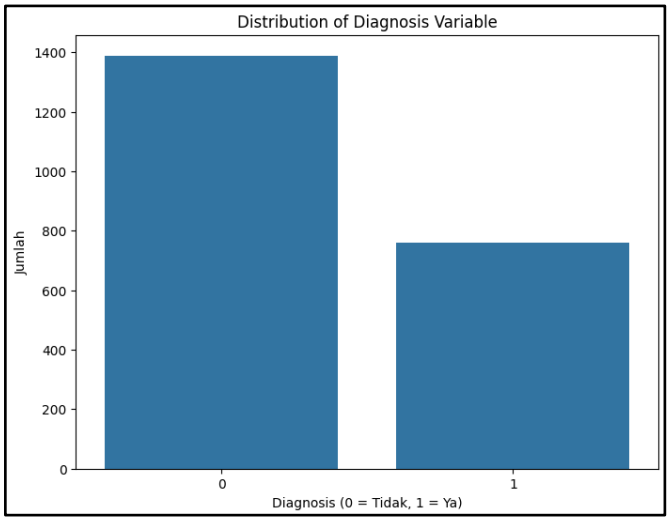
4. EXPLORATORY DATA ANALYSIS

Sebagai bagian dari eksplorasi data, matriks korelasi antar fitur dihitung dan divisualisasikan menggunakan heatmap. Matriks korelasi ini membantu memahami hubungan linier antara variabel-variabel dalam dataset. Secara umum, sebagian besar fitur memiliki korelasi yang lemah atau mendekati nol satu sama lain, menunjukkan hubungan linier yang lemah. Sebagai contoh, fitur diagnosis menunjukkan korelasi yang paling signifikan dengan functional_assessment dengan nilai(-0,36), mengindikasikan bahwa penilaian fungsional individu cenderung lebih rendah untuk individu dengan diagnosis positif. Selain itu, memory_complaints memiliki korelasi positif sebesar 0,31 dengan diagnosis, menunjukkan bahwa keluhan terkait memori cenderung meningkat pada individu dengan diagnosis positif. Korelasi ini dapat memberikan wawasan tentang peran keluhan memori dalam diagnosis. Di sisi lain, sebagian besar fitur seperti smoking, alcohol_consumption, dan physical_activity menunjukkan korelasi rendah dengan fitur lain, mencerminkan hubungan yang lebih independen terhadap variabel lain dalam dataset. Misalnya, korelasi antara smoking dan diagnosis hanya sebesar (-0,004), menunjukkan bahwa merokok mungkin tidak memiliki hubungan linier yang signifikan dengan diagnosis. Selain itu, fitur-fitur seperti diet_quality dan sleep_quality memiliki korelasi yang relatif kecil dengan sebagian besar fitur lainnya, menunjukkan bahwa mereka mungkin lebih independen. diet_quality menunjukkan korelasi positif kecil dengan physical_activity dengan nilai(0,01) dan diagnosis (0,01). Tidak ditemukan masalah multikolinearitas yang signifikan dalam dataset ini, karena tidak ada pasangan fitur yang memiliki korelasi mendekati 1. Hal ini memastikan bahwa fitur-fitur dalam dataset cukup independen dan tidak redundan. Hasil heatmap korelasi dapat dilihat pada gambar 6.



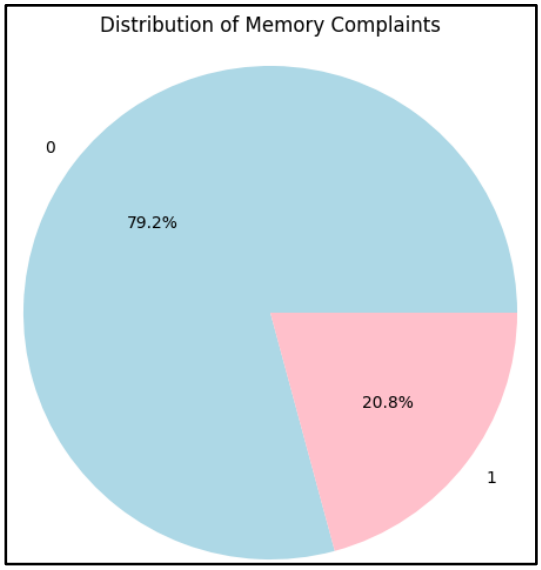
Gambar 6. Heatmap Korelasi

Berdasarkan hasil visualisasi pada bar chart ini, terlihat bahwa jumlah pasien yang tidak didiagnosis menderita Alzheimer (label diagnosis 0) lebih banyak dibandingkan dengan pasien yang positif menderita Alzheimer (label diagnosis 1). Hal ini menunjukkan adanya ketidakseimbangan dalam data, di mana proporsi pasien negatif lebih dominan dibandingkan dengan pasien positif. Hasil distribusi fitur diagosis dapat dilihat pada gambar 7.



Gambar 7. Bar Chart Fitur Diagnosis

Selanjutnya pada diagram pie chart ini, ditampilkan distribusi keluhan memori (memory complaints) pada dataset yang dianalisis. Diagram menunjukkan bahwa sebagian besar data, sebesar 79,2%, tidak memiliki keluhan memori (label 0), sedangkan sisanya, sebesar 20,8%, memiliki keluhan memori (label 1). Distribusi ini menunjukkan bahwa mayoritas individu dalam dataset tidak mengalami keluhan terkait memori, yang dapat menjadi indikasi bahwa sebagian besar populasi memiliki kondisi memori yang normal atau tidak memiliki gejala yang signifikan. Namun, keberadaan 20,8% individu yang memiliki keluhan memori perlu diperhatikan, karena kelompok ini dapat menjadi bagian penting dalam analisis lebih lanjut, terutama jika terkait dengan risiko atau diagnosis penyakit tertentu, seperti Alzheimer. Hasil pie chart dapat dilihat pada gambar 8.



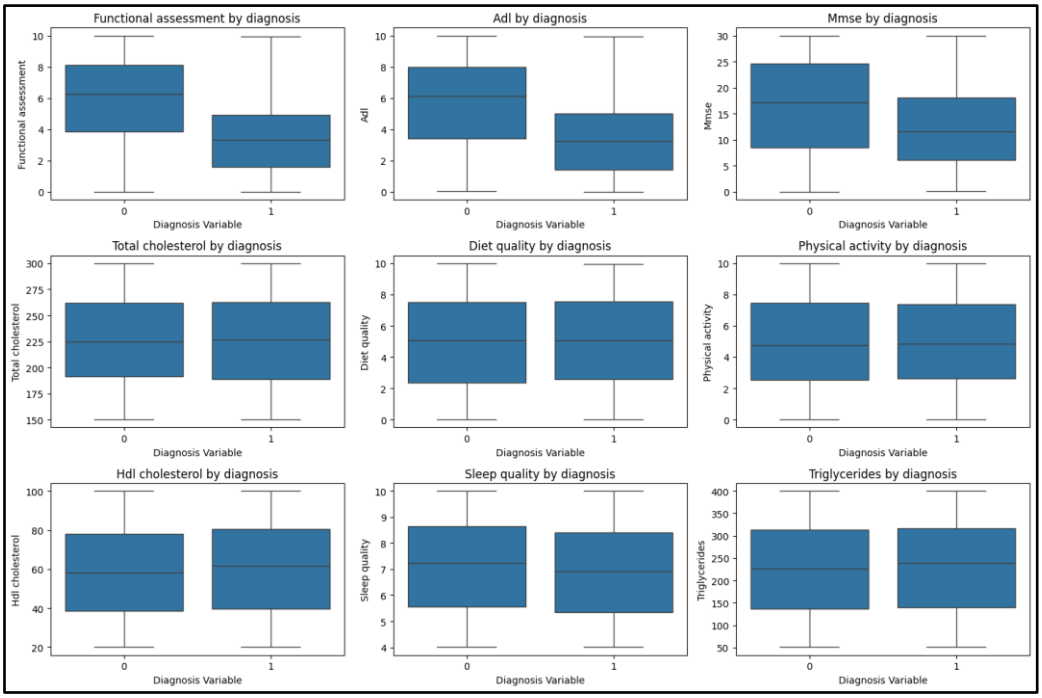
Gambar 8. Pie Chart Memory Complaints

Kemudian saya membuat visualisai dari boxt plot untuk mendeteksi outlier, hasil visualisasi dari diagram boxplot menunjukkan beberapa temuan menarik terkait perbedaan variabel antara diagnosis 0 menunjukkan pasien yang tidak terdiagnosis Alzheimer, sedangkan diagnosis 1 menunjukkan pasien yang positif Alzheimer. Pada variabel Functional Assessment, Diagnosis 1 memiliki rentang nilai yang lebih rendah dibandingkan Diagnosis 0, Median Functional Assessment untuk pasien dengan diagnosis 0 berada disekitar angka 6, sedangkan untuk diagnosis 1, median berada sekitar angka 3. Hal ini menunjukkan bahwa individu dengan Diagnosis 1 cenderung memiliki fungsi kognitif atau fisik yang lebih buruk. Hal serupa terlihat pada variabel ADL (Activities of Daily Living), distribusi pada diagnosis 0 menunjukkan rentang nilai ADL yang lebih tinggi dibandingkan diagnosis 1. Median ADL untuk pasien dengan diagnosis 0 berada di sekitar angka 6, sedangkan untuk diagnosis 1, median berada di sekitar angka 3, mengindikasikan bahwa mereka mungkin lebih kesulitan dalam menjalankan aktivitas sehari-hari.

Selain itu, rentang nilai MMSE (Mini-Mental State Examination) untuk Diagnosis 1 juga lebih rendah dibandingkan Diagnosis 0, yang mencerminkan adanya penurunan fungsi kognitif pada individu dengan Alzheimer. Aktivitas fisik dan kualitas tidur pada Diagnosis 1 juga cenderung lebih rendah dibandingkan Diagnosis 0, meskipun perbedaannya tidak terlalu mencolok.

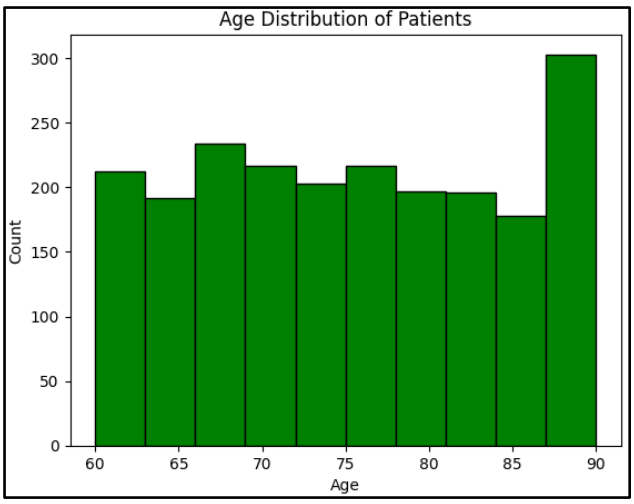
Namun, beberapa variabel metabolik, seperti kolesterol total, kolesterol HDL, dan trigliserida, tidak menunjukkan perbedaan yang signifikan antara kedua kelompok diagnosis. Kualitas diet juga tampak serupa pada kedua kelompok, menunjukkan bahwa faktor-faktor ini mungkin tidak secara langsung terkait dengan kondisi diagnosis atau dampaknya kurang terlihat dalam data ini.

Secara keseluruhan, individu dengan Diagnosis 1 menunjukkan penurunan fungsi kognitif, fisik, dan keseharian dibandingkan dengan Diagnosis 0, sementara faktor metabolik dan kualitas diet relatif tidak terpengaruh. Temuan ini mendukung hipotesis bahwa Alzheimer atau kondisi terkait lebih memengaruhi aspek kognitif dan gaya hidup dibandingkan faktor metabolik. Hasil box plot dapat dilihat pada gambar 9.



Gambar 9. Visualisasi Box Plot

Terakhir, saya membuat visualisasi histogram, diagram histogram ini menggambarkan distribusi usia pasien dalam dataset. Rentang usia pasien berada antara 60 hingga 90 tahun, yang dibagi menjadi 10 interval (bin). Dari diagram, terlihat bahwa distribusi usia pasien cukup merata pada sebagian besar interval, dengan puncak distribusi terjadi pada rentang usia 85 hingga 90 tahun. Rentang ini memiliki jumlah pasien tertinggi, yaitu lebih dari 300 orang. Sementara itu, interval usia lainnya, terutama antara 60 hingga 85 tahun, memiliki jumlah pasien yang relatif lebih rendah, meskipun distribusinya tetap stabil dengan fluktuasi kecil di antara interval. Distribusi ini menunjukkan bahwa dataset lebih didominasi oleh pasien lanjut usia, khususnya yang berusia mendekati 90 tahun. Hasil diagram histogram dapat dilihat pada gambar 10.



Gambar 10. Diagram Histogram Umur Pasien

5. SELEKSI FITUR

Pada proyek ini, pemilihan fitur saya lakukan menggunakan Random Forest karena algoritma ini efektif dalam menangani dataset dengan berbagai ukuran, baik kecil maupun besar. Random Forest merupakan algoritma yang sangat fleksibel dan andal untuk menangani dataset dengan tipe data numerik, seperti pada proyek ini, di mana seluruh fitur dalam dataset terdiri dari data numerik. Algoritma ini bekerja dengan membangun banyak pohon keputusan yang digabungkan untuk menghasilkan prediksi yang lebih akurat. Selain itu, Random Forest tidak memerlukan normalisasi atau transformasi data khusus, sehingga dapat langsung diterapkan pada dataset dengan skala variabel yang beragam. Random Forest juga mampu menangani hubungan yang kompleks dan non-linear antar fitur dengan baik, serta dapat mengidentifikasi fitur yang paling relevan meskipun terdapat banyak fitur dalam dataset. Pemilihan fitur ini bertujuan untuk meningkatkan efisiensi dan akurasi model prediksi dengan hanya mempertimbangkan variabel-variabel yang paling signifikan. Hal ini juga membantu mengurangi kompleksitas model sekaligus mempertahankan interpretabilitas dan relevansi terhadap permasalahan yang ingin diselesaikan, yaitu klasifikasi penyakit Alzheimer. Dari analisis, diperoleh 18 fitur, dengan nilai kepentingan (importance) tertinggi. Fitur dengan nilai kepentingan tertinggi adalah functional_assessment dengan nilai importance sebesar 0,187101, diikuti oleh ADL sebesar 0,164764 dan MMSE sebesar 0,121234. Fitur lainnya termasuk memory_complaints, behavioral_problems, physical_activity, HDL_cholesterol, sleep_quality, Triglycerides, total_cholesterol, diet_quality, BMI, alcohol_consumption, LDL_cholesterol, systolic_BP, diastolic_BP, age, dan education_level. Seleksi fitur dilakukan dengan menerapkan ambang batas (threshold) sebesar 0,01. Berdasarkan kriteria ini, semua fitur yang diambil memiliki nilai importance lebih dari 0,01. Selain 18 fitur utama ini, saya melakukan drop terhadap fitur yang tidak terdapat pada hasil pemilihan ini. Hasil dari pemilihan fitur dapat dilihat pada Gambar 11.

```
Feature Importance:
 functional_assessment        0.187101
ADL                           0.164764
MMSE                          0.121234
memory_complaints             0.075928
behavioral_problems           0.043380
physical_activity             0.032228
HDl_cholesterol               0.030657
sleep_quality                 0.030560
Triglycerides                 0.029869
total_cholesterol             0.029589
diet_quality                  0.029418
BMI                           0.029128
alcohol_consumption           0.028697
LDL_cholesterol               0.028244
systolic_BP                   0.027566
diastolic_BP                  0.024985
age                           0.023360
education_level               0.010767
ethnicity                     0.007266
gender                        0.004575
depression                    0.004527
cardiovascular_disease        0.004056
forgetfulness                 0.003940
family_history_AD             0.003543
smoking                       0.003514
confusion                     0.003376
hypertension                  0.003291
task_completion_difficulty    0.003193
diabetes                      0.003149
personality_changes           0.002823
head_injury                   0.002724
disorientation                0.002549
dtype: float64

Fitur terpilih berdasarkan feature importance: ['functional_assessment',
```

Gambar 11. Hasil Pemilihan Fitur

Fitur-fitur yang tidak termasuk dalam hasil seleksi fitur berdasarkan nilai importance di-drop dari dataset. Langkah ini bertujuan untuk memastikan bahwa hanya fitur yang paling relevan digunakan dalam pembangunan model, sehingga mempermudah proses analisis dan pemodelan. Dengan mengurangi jumlah fitur yang diproses, kompleksitas data dapat diminimalkan, sehingga proses komputasi menjadi lebih efisien. Selain itu, penghapusan fitur yang tidak signifikan juga dapat mengurangi risiko overfitting, di mana model menjadi terlalu bergantung pada fitur-fitur yang tidak relevan dan kehilangan kemampuan generalisasi terhadap data baru. Proses ini membantu menjaga fokus model pada variabel-variabel yang benar-benar memiliki kontribusi terhadap klasifikasi penyakit Alzheimer. Dengan demikian, model yang dihasilkan tidak hanya lebih akurat tetapi juga lebih sederhana dan mudah diinterpretasikan. Kode untuk mengganti nama kolom dapat dilihat pada Gambar 12.

```
#Drop fitur yang tidak terdapat pada feature selection
df = df.drop(['disorientation', 'head_injury', 'personality_changes', 'diabetes', 'task_completion_difficulty',
    'hypertension', 'confusion', 'smoking', 'family_history_AD', 'forgetfulness', 'cardiovascular_disease',
    'depression', 'gender', 'ethnicity'], axis=1)
df.head()
```

Gambar 12. Kode Untuk Menghapus Fitur Yang Tidak Penting

Setelah dilakukan penghapusan fitur yang tidak relevan berdasarkan seleksi fitur menggunakan Random Forest, jumlah fitur dalam dataset berkurang dari sebelumnya 35 fitur menjadi 19 fitur. Pengurangan ini mencakup fitur-fitur yang tidak memenuhi ambang batas nilai importance serta fitur target (diagnosis) yang tetap dipertahankan. Dengan dataset yang lebih sederhana dan fokus, diharapkan model dapat menghasilkan prediksi yang lebih akurat dan mudah diinterpretasikan. Jumlah sampel data dan fitur dari dataset dapat dilihat pada gambar 13.

```
# Melihat ukuran data
df.shape

(2149, 19)
```

Gambar 13. Jumlah Data Pada Dataset

6. MODELING

Sebelum melakukan proses pemodelan, saya terlebih dahulu melakukan pembagian dataset (split data) yang telah di-oversampling menjadi data training dan data testing dengan proporsi 80:20. Pembagian ini dilakukan untuk memastikan bahwa model dapat dilatih menggunakan data training sebanyak mungkin dan diuji performanya pada data testing secara adil. Dengan proporsi ini, data training berjumlah 2.222, sedangkan data testing berjumlah 556. Secara umum, model machine learning cenderung memberikan hasil akurasi yang lebih baik ketika memiliki data training yang lebih banyak dibandingkan data testing, karena data training yang lebih besar memberikan lebih banyak informasi bagi model untuk belajar pola yang relevan. Pada Tabel 2 berikut, dapat dilihat rincian pembagian data yang dilakukan dalam proyek ini.

Tabel 2. Train/Test Split

Keterangan	Data Training	Data Testing	Total
Proporsi	80%	20%	100%
Jumlah	2222	556	2778

Selanjutnya, pada proses pemodelan ini, saya menggunakan algoritma XGBoost (Extreme Gradient Boosting), yang merupakan salah satu algoritma machine learning berbasis tree dengan performa tinggi. Algoritma ini saya pilih karena mampu menangani dataset dengan dimensi tinggi, ketidakseimbangan kelas, serta menghasilkan prediksi yang akurat melalui mekanisme boosting secara iteratif. Selain itu, XGBoost memiliki keunggulan dalam hal efisiensi waktu komputasi dan kemampuan generalisasi yang baik. Pada pemodelan ini, saya menggunakan parameter default sebagai baseline untuk mengevaluasi performa awal model. Parameter seperti learning rate, maksimum depth, dan jumlah estimators akan dioptimasi lebih lanjut menggunakan teknik tuning hyperparameter untuk meningkatkan performa model. Kode untuk pelatihan model dan prediksi dapat dilihat pada Gambar 14.

```
# Melatih model dengan XGBoost
model_xgb = XGBClassifier(random_state=42)
model_xgb.fit(X_train, y_train)
y_pred_xgb = model_xgb.predict(X_test)
```

Gambar 14. Proses Pelatihan Model Dengan XGBoost

Setelah melakukan pemodelan awal, langkah selanjutnya saya melakukan proses tuning hyperparameter untuk meningkatkan performa model XGBoost. Proses ini bertujuan untuk menemukan kombinasi parameter terbaik yang dapat memaksimalkan akurasi prediksi model pada data testing. Beberapa parameter utama yang dioptimasi meliputi n_estimators, learning_rate, max_depth, subsample, dan colsample_bytree. Kode untuk hyperparameter tuning dapat dilihat pada Gambar 15.

```
# Tentukan parameter grid
param_grid = {
    'n_estimators': [50, 100, 200, 300, 500],
    'learning_rate': [0.001, 0.01, 0.1, 0.2],
    'max_depth': [3, 5, 7, 9, 11],
    'subsample': [0.6, 0.7, 0.8, 0.9, 1.0],
    'colsample_bytree': [0.6, 0.7, 0.8, 0.9, 1.0],
}

# Inisialisasi model XGBoost
best_accuracy = 0
best_params = {}

# Melakukan pencarian grid manual
for n_estimators in param_grid['n_estimators']:
    for learning_rate in param_grid['learning_rate']:
        for max_depth in param_grid['max_depth']:
            for subsample in param_grid['subsample']:
                for colsample_bytree in param_grid['colsample_bytree']:
                    # Inisialisasi model dengan parameter yang sedang diuji
                    model = XGBClassifier(n_estimators=n_estimators,
                                          learning_rate=learning_rate,
                                          max_depth=max_depth,
                                          subsample=subsample,
                                          colsample_bytree=colsample_bytree,
                                          random_state=42)

                    # Latih model
                    model.fit(X_train, y_train)

                    # Evaluasi model
                    y_pred = model.predict(X_test)
                    accuracy = model.score(X_test, y_test)

                    # Simpan hasil terbaik
                    if accuracy > best_accuracy:
                        best_accuracy = accuracy
                        best_params = {
                            'n_estimators': n_estimators,
                            'learning_rate': learning_rate,
                            'max_depth': max_depth,
                            'subsample': subsample,
                            'colsample_bytree': colsample_bytree
                        }

# Menampilkan parameter terbaik dan hasil akurasi terbaik
print("Best Parameters:", best_params)
print("Best Accuracy on Test Data:", best_accuracy)

# Evaluasi model dengan parameter terbaik
model_best = XGBClassifier(**best_params, random_state=42)
model_best.fit(X_train, y_train)
y_pred_best = model_best.predict(X_test)
print(classification_report(y_test, y_pred_best))
```

Gambar 15. Kode Melakukan Hyperparameter Tuning

Link GitHub: https://github.com/MIKHSAN-22115260/KLASIFIKASI-PENYAKIT-ALZHEIMER-MENGGUNAKAN-ALGORITMA-EXTREME-GRADIENT-BOOSTING

Link google colab:
https://colab.research.google.com/drive/1MIW5D8-Ihw1Wd3Fhl2Uo1hQ4Jz8jdImo

Link Launchinpad:
https://launchinpad.com/project/alzheimers-disease-classification-a9dcaaf

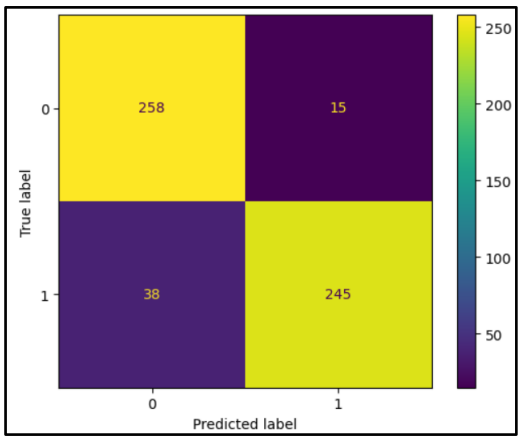
7. EVALUASI MODEL

Hasil evaluasi model XGBoost menunjukkan performa yang sangat baik pada data testing, meskipun parameter model masih menggunakan nilai default (belum dituning). Model mencapai akurasi sebesar 90%, yang mencerminkan bahwa sebagian besar prediksi model sesuai dengan label sebenarnya. Untuk kelas 0, precision sebesar 87% dan recall sebesar 95% menunjukkan bahwa model sangat baik dalam mengidentifikasi data yang benar-benar termasuk dalam kelas ini, meskipun terdapat beberapa data lain yang salah diklasifikasikan ke kelas ini. Sementara itu, untuk kelas 1, precision sebesar 94% dan recall sebesar 87% menunjukkan bahwa model sedikit lebih baik dalam mengidentifikasi data yang benar-benar termasuk kelas ini dibandingkan menghindari kesalahan prediksi ke kelas ini. Nilai rata-rata makro (macro avg) untuk precision, recall, dan F1-score adalah 91%, yang menunjukkan performa yang seimbang antara kedua kelas. Selain itu, rata-rata berbobot (weighted avg) juga menghasilkan nilai 90%, mengindikasikan bahwa model mempertimbangkan distribusi kelas dengan baik dalam performanya. Rincian laporan klasifikasi dapat dilihat pada Gambar 16.

	precision	recall	f1-score	support
0	0.87	0.95	0.91	273
1	0.94	0.87	0.90	283
accuracy			0.90	556
macro avg	0.91	0.91	0.90	556
weighted avg	0.91	0.90	0.90	556

Gambar 16. Laporan Klasifikasi Model XGBoost

Selanjutnya hasil evaluasi dianalisis menggunakan confusion matrix. Confusion matrix menunjukkan bahwa model menghasilkan 258 True Negative (prediksi benar untuk kelas 0: Non-Alzheimer), 15 False Positive (prediksi salah di mana kelas sebenarnya 0, tetapi diprediksi 1: Alzheimer), 38 False Negative (prediksi salah di mana kelas sebenarnya 1: Alzheimer, tetapi diprediksi 0: Non-Alzheimer), dan 245 True Positive (prediksi benar untuk kelas 1: Alzheimer). Hasil ini menunjukkan bahwa model memiliki kemampuan yang baik dalam membedakan antara kelas Non-Alzheimer dan Alzheimer, meskipun terdapat beberapa kesalahan prediksi, tetapi hasil ini sudah cukup baik. Rincian confusion matrix dapat dilihat pada Gambar 17.



Gambar 17. Confusion Matrix untuk Model XGBoost

Kemudian dari hasil hyperparameter tuning menunjukkan bahwa kombinasi parameter terbaik untuk model XGBoost adalah sebagai berikut: n_estimators = 100, learning_rate = 0.2, max_depth = 11, subsample = 0.9, dan colsample_bytree = 0.6. Dengan parameter ini, model XGBoost mencapai akurasi terbaik sebesar 91.55%, yang menunjukkan peningkatan sebesar 2% dibandingkan dengan model baseline. Model akhir dengan parameter optimal diuji kembali menggunakan data testing, dan hasil evaluasinya menunjukkan peningkatan performa yang signifikan. Metrik klasifikasi seperti precision, recall, F1-score, dan accuracy menunjukkan bahwa model mampu menangkap pola data secara lebih baik. Untuk kelas 0 (Non-Alzheimer), precision sebesar 90% dan recall sebesar 94% menunjukkan identifikasi yang sangat baik, sementara untuk kelas 1 (Alzheimer), precision sebesar 94% dan recall sebesar 89% menunjukkan model mampu mengklasifikasikan data dengan akurat meskipun ada beberapa kesalahan prediksi. Nilai rata-rata makro (macro avg) untuk precision, recall, dan F1-score adalah 92%, yang menunjukkan performa yang seimbang antara kedua kelas. Rata-rata berbobot (weighted avg) juga menghasilkan nilai 92%, mengindikasikan bahwa model mempertimbangkan distribusi kelas dengan baik dalam performanya. Secara keseluruhan, hasil ini menunjukkan bahwa model XGBoost setelah tuning memiliki kemampuan klasifikasi yang sangat baik untuk membedakan antara kelas Non-Alzheimer dan Alzheimer, dengan peningkatan performa sebesar 2% pada akurasi dibandingkan model baseline. Rincian laporan klasifikasi dari hasil hyperparameter tuning dapat dilihat pada Gambar 18.

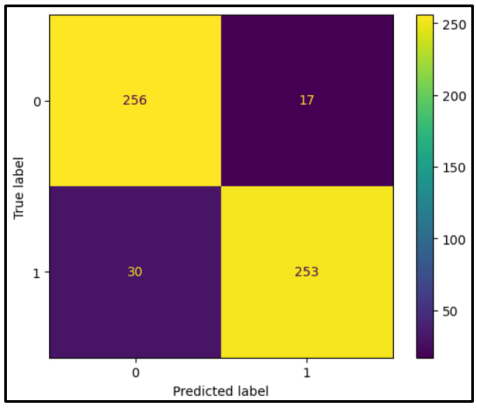
```
Best Parameters: {'n_estimators': 100, 'learning_rate': 0.2, 'max_depth': 11, 'subsample': 0.9, 'colsample_bytree': 0.6}
Best Accuracy on Test Data: 0.9154676258992805
              precision    recall  f1-score   support

           0       0.90      0.94      0.92       273
           1       0.94      0.89      0.92       283

    accuracy                           0.92       556
   macro avg       0.92      0.92      0.92       556
weighted avg       0.92      0.92      0.92       556
```

Gambar 18. Laporan Klasifikasi Hasil Tuning Model XGBoost

Berikutnya, hasil evaluasi model XGBoost setelah tuning juga dianalisis menggunakan confusion matrix. Berdasarkan confusion matrix, model menghasilkan 256 True Negative (prediksi benar untuk kelas 0: Non-Alzheimer), 17 False Positive (prediksi salah di mana kelas sebenarnya 0, tetapi diprediksi 1: Alzheimer), 30 False Negative (prediksi salah di mana kelas sebenarnya 1: Alzheimer, tetapi diprediksi 0: Non-Alzheimer), dan 253 True Positive (prediksi benar untuk kelas 1: Alzheimer). Hasil ini menunjukkan bahwa model setelah tuning dapat mengklasifikasikan data dengan baik, meskipun masih ada kesalahan prediksi baik untuk kelas Non-Alzheimer maupun Alzheimer. Meskipun demikian, model menunjukkan performa yang lebih baik setelah tuning dibandingkan dengan model sebelumnya, dengan kesalahan yang lebih sedikit. Rincian confusion matrix hasil dari tuning dapat dilihat pada Gambar 17.



Gambar 17. Confusion Matrix Hasil Tuning Model XGBoost

8. ANALISA DAN PEMBAHASAN

Dari model XGBoost yang telah saya buat dan dilakukan hyperparameter tuning menunjukkan performa yang sangat baik dalam memprediksi klasifikasi negative Alzheimer dan positif Alzheimer. Hal ini terlihat dari metrik evaluasi yang konsisten tinggi, dengan akurasi sebesar 91.55%**,** meningkat sebesar 2% dibandingkan dengan model sebelum dilakukan hyperparameter tuning.

Kemampuan model dalam membedakan kedua kelas dapat dijelaskan oleh kekuatan algoritma XGBoost yang secara intrinsik menggunakan boosting untuk meningkatkan performa model secara bertahap melalui iterasi. Parameter optimal yang diperoleh dari tuning, seperti learning_rate = 0.2 dan max_depth = 11, memungkinkan model untuk menangkap pola-pola penting dalam data tanpa kehilangan kemampuan generalisasi. Selain itu, kombinasi subsample = 0.9 dan colsample_bytree = 0.6 membantu model untuk bekerja secara efisien dengan memilih subset data dan fitur yang relevan pada setiap iterasi.

Hasil confusion matrix juga sangat baik, di mana model mampu mengklasifikasikan sebagian besar sampel dengan benar. Sebanyak 256 True Negative untuk kelas Non-Alzheimer dan 253 True Positive untuk kelas positif Alzheimer, ini menunjukkan bahwa model memiliki keakuratan yang tinggi dalam memprediksi kelas sebenarnya. Kesalahan prediksi relatif kecil, yaitu 17 False Positive dan 30 False Negative, menunjukkan bahwa meskipun ada beberapa kesalahan klasifikasi, model yang saya buat ini secara keseluruhan mampu menangkap pola dari data dengan sangat baik.

Evaluasi pada metrik macro average dan weighted average yang keduanya sebesar 92% menunjukkan bahwa model memiliki performa yang seimbang untuk kedua kelas, tanpa ada bias yang signifikan terhadap salah satu kelas. Hal ini penting mengingat klasifikasi yang seimbang sangat dibutuhkan untuk memastikan prediksi yang akurat dan adil.

Secara keseluruhan, performa yang dicapai oleh model ini mencerminkan bahwa XGBoost merupakan algoritma yang sangat efektif untuk tugas klasifikasi ini. Penggunaan parameter optimal dari hasil tuning berkontribusi besar terhadap peningkatan akurasi model dan kemampuan untuk mengenali pola kompleks dalam data.

9. KESIMPULAN

Berdasarkan eksperimen yang telah saya lakukan, dapat disimpulkan bahwa:

1. Algoritma XGBoost berhasil diimplementasikan untuk melakukan klasifikasi Non-Alzheimer dan positif Alzheimer dengan tingkat akurasi yang tinggi.
2. Setelah dilakukan tuning hyperparameter, model mencapai akurasi terbaik sebesar 91.55% atau digenap jadi 92%, yang menunjukkan peningkatan sebesar 2% dibandingkan dengan model default atau sebelum dilakukan tuning.
3. Evaluasi menggunakan metrik precision, recall, dan F1-score menunjukkan hasil yang konsisten dan seimbang untuk kedua kelas, dengan rata-rata nilai sebesar 92%.
4. Confusion matrix menunjukkan bahwa model mampu mengklasifikasikan data dengan baik, dengan jumlah prediksi benar yang signifikan pada kedua kelas, yaitu 256 True Negative untuk kelas Non-Alzheimer dan 253 True Positive untuk kelas positif Alzheimer.
5. Hasil eksperimen ini menunjukkan bahwa algoritma XGBoost dengan parameter optimal merupakan solusi yang efektif dan andal untuk menangani masalah klasifikasi seperti yang terdapat dalam dataset ini.

10. REFERENSI

[1] L. Chen, H. Qiao, and F. Zhu, "Alzheimer's Disease Diagnosis With Brain Structural MRI Using Multiview-Slice Attention and 3D Convolution Neural Network," *Front. Aging Neurosci.*, vol. 14, no. April, 2022, doi: 10.3389/fnagi.2022.871706.

[2] J. Tian *et al.*, "Modular machine learning for Alzheimer's disease classification from retinal vasculature," *Sci. Rep.*, vol. 11, no. 1, pp. 1–11, 2021, doi: 10.1038/s41598-020-80312-2.

[3] M. Liu *et al.*, "A multi-model deep convolutional neural network for automatic hippocampus segmentation and classification in Alzheimer's disease," *Neuroimage*, vol. 208, no. December 2019, 2020, doi: 10.1016/j.neuroimage.2019.116459.

[4] P. Gayathri *et al.*, "Deep Learning Augmented with SMOTE for Timely Alzheimer's Disease Detection in MRI Images," *Int. J. Adv. Comput. Sci. Appl.*, vol. 15, no. 2, pp. 499–508, 2024, doi: 10.14569/IJACSA.2024.0150253.

[5] S. T. Ahmed and S. M. Kadhem, "Alzheimer's disease prediction using three machine learning methods," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 27, no. 3, pp. 1689–1697, 2022, doi: 10.11591/ijeecs.v27.i3.pp1689-1697.

[6] A. Lubis, S. Sibagariang, and N. Ardi, "Classification of Alzheimer Disease from MRI Image Using Combination Naïve Bayes and Invariant Moment," 2023, doi: 10.4108/eai.5-10-2022.2327750.

[7] H. S. SURESHA and S. S. PARTHASARATHI, "Relieff Feature Selection Based Alzheimer Disease Classification Using Hybrid Features and Support Vector Machine in Magnetic Resonance Imaging," *Int. J. Comput. Eng. Technol.*, vol. 10, no. 1, pp. 124–137, 2019, doi: 10.34218/ijcet.10.1.2019.015.

[8] S. Shahzadi *et al.*, "Voxel Extraction and Multiclass Classification of Identified Brain Regions across Various Stages of Alzheimer's Disease Using Machine Learning Approaches," *Diagnostics*, vol. 13, no. 18, 2023, doi: 10.3390/diagnostics13182871.

[9] A. Jenber Belay, Y. M. Walle, and M. B. Haile, "Deep Ensemble learning and quantum machine learning approach for Alzheimer's disease detection," *Sci. Rep.*, vol. 14, no. 1, pp. 1–10, 2024, doi: 10.1038/s41598-024-61452-1.

[10] S. Qiu *et al.*, "Development and validation of an interpretable deep learning framework for Alzheimer's disease classification," *Brain*, vol. 143, no. 6, pp. 1920–1933, 2020, doi: 10.1093/brain/awaa137.