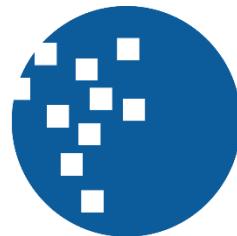


PROPOSAL PROJEK

PEMAHAMAN TINGKAT CHURN: PENDEKATAN

DECISION TREE RANDOM FOREST DALAM

ANALISIS BANK



UMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA

Kelompok 2

Kelas IF540-A

Disusun Oleh :

Joe Marcello - 00000073881

Muhammad Evan Julian Priyasa- 00000072402

Febianus Felix Widisulistiyo- 00000072737

Rivaldo Yosia Himawan - 00000071997

PROGRAM STUDI SISTEM INFORMASI

FAKULTAS TEKNIK DAN INFORMATIKA

UNIVERSITAS MULTIMEDIA NUSANTARA

TANGERANG

2024

KATA PENGANTAR

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa, karena berkat dan rahmat-Nya, kami dapat menyelesaikan laporan ini sebagai tugas akhir. Laporan ini diharapkan dapat menjadi acuan untuk memperluas wawasan, pengetahuan, dan mengembangkan kemampuan kami dalam memahami dan menganalisis tingkat churn menggunakan pendekatan Decision Tree Random Forest dalam analisis bank.

Laporan ini bertujuan memberikan informasi dan pengetahuan tentang penerapan Machine Learning, khususnya dalam memahami tingkat churn di sektor perbankan. Proyek ini juga bertujuan memberikan pemahaman sederhana kepada pembaca, sehingga mampu menganalisis berbagai aspek dalam kehidupan sehari-hari.

Dalam penyusunan laporan ini, kami banyak mendapatkan dukungan, bimbingan, dan bantuan dari berbagai pihak. Pada kesempatan ini, kami ingin menyampaikan terima kasih kepada:

1. Ibu Ririn Ikana Desanti, S.Kom., M.Kom. selaku Ketua Program Studi program studi Sistem Informasi Universitas Multimedia Nusantara.
2. Marlinda Vasty Overbeek, S.Kom, M.Kom. selaku Dosen Pengampu Mata Kuliah Machine Learning dan sebagai pembimbing yang memberikan arahan dan bimbingan agar laporan ini bisa selesai sebagaimana mestinya.
3. Jezreel Kosasih & Sagita Sasmita Wijaya sebagai asisten lab kelas Machine Learning yang selalu bersedia menjawab pertanyaan dan membantu kami dalam menyelesaikan laporan akhir ini.
4. Teman-teman jurusan Sistem Informasi yang memberikan dukungan dan semangat kepada kami sehingga laporan ini bisa selesai dengan baik sebagaimana mestinya.

Semoga laporan ini bermanfaat bagi proyek kami, pembaca, dan calon peneliti yang akan melakukan penelitian di bidang serupa. Kami juga memohon maaf atas segala kesalahan dalam penulisan, pengejaan, dan sistematis EYD dalam laporan ini. Kami terbuka terhadap saran dan kritik yang membangun, sehingga dapat meningkatkan kualitas penulisan proyek ini di masa mendatang.

Tangerang, 11 Maret 2024

DAFTAR ISI

KATA PENGANTAR.....	2
DAFTAR ISI.....	3
DAFTAR TABEL.....	5
DAFTAR GAMBAR.....	6
DAFTAR LAMPIRAN.....	8
ABSTRAK.....	1
ABSTRACT (English).....	2
BAB I	
PENDAHULUAN.....	3
1.1 Latar Belakang.....	3
1.2 Rumusan Masalah.....	5
1.3 Batasan Masalah.....	5
1.4 Tujuan dan Manfaat Penelitian.....	5
1.4.1 Tujuan Penelitian.....	5
1.4.2 Manfaat Penelitian.....	6
BAB II	
LANDASAN TEORI.....	7
2.1 Telaah Literatur.....	7
2.1.1 Bank Churn.....	7
2.1.2 Machine Learning.....	7
2.1.3 Supervised Learning.....	8
2.1.4 Decision Tree.....	8
2.1.5 Random Forest.....	10
2.1.6 Confusion Matrix.....	11
2.1.7 F1-Score.....	13
BAB III	
METODOLOGI PENELITIAN.....	14
3.1 Gambaran Umum Dataset Penelitian.....	15
3.2 Exploratory Data Analysis.....	18
3.2.1 Distribusi Credit Score.....	18
3.2.2 Distribusi Estimated Salary.....	19
3.2.3 Distribusi Scatter Plot Estimated Salary Vs Credit Score.....	20
3.2.4 Distribusi Age.....	21
3.2.5 Distribusi Perbandingan Negara.....	22
3.2.6 Distribusi Barplot Gender.....	23
3.2.7 Distribusi Balance.....	24
3.2.8 Distribusi Scatter Plot Age Vs Balance.....	25

3.3 Preprocess Data.....	26
3.3.1 Checking Missing Value.....	26
3.3.2 Encoding.....	27
3.3.3 Handling Outlier.....	27
3.3.4 Normalization.....	29
3.3.5 Features Importances.....	31
3.3.6 Heat Map.....	33
3.4 Rekayasa Fitur.....	34
3.4.1 Feature Selection.....	34
3.5 Pemodelan Data.....	34
3.5.1 Splitting Train Test.....	34
3.5.2 Cross Validation.....	34
3.5.3 Decision Tree Classifier.....	35
3.5.4 Random Forest Classifier Hyperparameter.....	36
3.6 Validasi dan Evaluasi Model.....	36
3.6.1 Decision Tree.....	36
3.6.1.1 Evaluasi Cross-Validation.....	36
3.6.1.2 Accuracy Training dan Test Decision Tree.....	37
3.6.1.3 Evaluasi Metrics.....	37
3.6.1.4 Visualisasi Confusion Matrix.....	38
3.6.1.5 Visualisasi Decision Tree.....	38
3.6.2 Random Forest.....	39
3.6.2.1 Accuracy Training dan Test Random Forest.....	39
3.6.2.2 Evaluasi Metrics.....	40
3.6.2.3 Visualisasi Confusion Matrix.....	40
3.6.2.4 Visualisasi Decision Tree.....	41
BAB IV	
Analisis dan Hasil Penelitian.....	42
4.1 Analisa Masalah.....	42
4.2 Hasil Pemodelan.....	45
4.3 Hasil Validasi dan Evaluasi Model.....	46
4.4 Pembahasan Hasil yang Didapatkan.....	52
BAB V	
SIMPULAN DAN SARAN.....	54
5.1 Simpulan.....	54
5.2 Saran.....	55
DAFTAR PUSTAKA.....	56
LAMPIRAN.....	59

DAFTAR TABEL

Tabel 3.1.1. Eksplorasi Dataset	15
Tabel 4.1.1.1 Perhitungan dengan Sampel Data	43
Tabel 4.2.1 Hasil Pemodelan	45
Tabel 4.3.1.1 Validasi Model	47
Tabel 4.3.2.1 Evaluasi Model Decision Tree	47
Tabel 4.3.4 Evaluasi Model Random Forest Hyperparameter	50

DAFTAR GAMBAR

Gambar 2.1.1 Struktur Decision Tree	9
Gambar 2.1.2 Visualisasi Random Forest	11
Gambar 3. Metodologi Penelitian	14
Gambar 3.2.1.1 Code Distribusi Credit Score	18
Gambar 3.2.1.2 Histogram Distribusi Credit Score	19
Gambar 3.2.2.1 Code Distribusi Estimated Salary	19
Gambar 3.2.2.2 Histogram Distribusi Estimated Salary	20
Gambar 3.2.3.1 Code Distribusi Scatter Plot Estimated Salary Vs Credit Score	20
Gambar 3.2.3.2 Scatter Plot Estimated Salary Vs Credit Score	22
Gambar 3.2.4.1 Code Distribusi Age	21
Gambar 3.2.4.2 Histogram Distribusi Age	21
Gambar 3.2.5.1 Code Distribusi Perbandingan Negara	22
Gambar 3.2.5.2 Pie Chart Distribusi Perbandingan Negara	23
Gambar 3.2.6.1 Code Distribusi Barplot Gender	23
Gambar 3.2.6.2 Barplot Distribusi Barplot Gender	24
Gambar 3.2.7.1 Code Distribusi Balance	24
Gambar 3.2.7.2 Histogram Distribusi Balance	25
Gambar 3.2.7.1 Code Distribusi Scatter Plot Age Vs Balance	25
Gambar 3.2.7.2 Distribusi Scatter Plot Age Vs Balance	26
Gambar 3.3.1.1 Hasil Checking Missing Value	26
Gambar 3.3.2.1 Code Encoding	27
Gambar 3.3.3.1 Code Checking Outlier	27
Gambar 3.3.3.2 Boxplot Outlier	28
Gambar 3.3.3.3 Hasil Outlier	28
Gambar 3.3.3.4 Code Handling Outlier	29
Gambar 3.3.4.1 Code Normalization	29
Gambar 3.3.4.2 Handling Outlier	30
Gambar 3.3.5.1 Code Features Importances	31
Gambar 3.3.5.2 Barplot Features Importances	32
Gambar 3.3.6.1 Hasil Heat map	33
Gambar 3.4.1.1 Code Feature Selection	34
Gambar 3.5.1.1 Code Splitting Train Test	34
Gambar 3.5.2.1 Code Cross Validation	34
Gambar 3.5.3.1 Code Decision Tree Classifier (entropy)	35
Gambar 3.5.3.2 Code Decision Tree Classifier (Max Depth)	35
Gambar 3.5.4.1 Code Random Forest Classifier Hyperparameter	36

Gambar 3.6.1.1.1 Code Evaluasi Cross-Validation	36
Gambar 3.6.1.2.1 Code Accuracy Training dan Test Decision Tree	37
Gambar 3.6.1.3.1 Code Evaluasi Metrics	37
Gambar 3.6.1.4.1 Visualisasi Confusion Matrix	38
Gambar 3.6.1.5.1 Code Visualisasi Decision Tree	38
Gambar 3.6.2.1.1 Code Accuracy Training dan Test Random Forest	39
Gambar 3.6.2.2.1 Code Evaluasi Metrics	40
Gambar 3.6.2.3.1 Visualisasi Confusion Matrix	40
Gambar 3.6.2.4.1 Visualisasi Decision Tree	41
Gambar 4.3.3.1 Confusion Matrix Decision Tree	48
Gambar 4.3.4.1 Visualisasi Decision Tree (Non Max Depth)	49
Gambar 4.3.6.1 Confusion Matrix Random Forest	51
Gambar 4.3.7.2 Visualisasi Random Forest	52

DAFTAR LAMPIRAN

Lampiran 1. Hasil Turnitin

59

PEMAHAMAN TINGKAT CHURN: PENDEKATAN DECISION TREE RANDOM FOREST DALAM ANALISIS BANK

ABSTRAK

Penelitian ini bertujuan untuk menganalisis dan memprediksi churn pelanggan dalam industri perbankan menggunakan teknik machine learning. Faktor-faktor seperti layanan keuangan yang lebih baik, biaya lebih rendah, lokasi cabang, dan suku bunga yang lebih rendah diidentifikasi sebagai alasan utama pelanggan beralih ke bank lain. Untuk memahami perilaku churn pelanggan, digunakan model prediktif berbasis machine learning, khususnya Decision Tree dan Random Forest. Evaluasi kinerja model menunjukkan bahwa akurasi pengujian yang lebih tinggi (83.9%) dicapai oleh Random Forest dibandingkan dengan Decision Tree (78%), serta konsistensi yang lebih baik dalam metode cross-validation. Selain itu, kemampuan yang lebih baik dalam memprediksi kelas churn pelanggan ditunjukkan oleh Random Forest dengan nilai recall dan f1-score yang lebih tinggi untuk kelas 1 (keluar).

Hasil evaluasi menunjukkan bahwa kelas 0 (tidak keluar) diprediksi lebih efektif oleh kedua model dibandingkan kelas 1 (keluar). Namun, keunggulan dalam memprediksi kelas 1 ditunjukkan oleh Random Forest, yang merupakan target utama dalam analisis churn. Penggunaan fitur dan aturan yang lebih kompleks oleh Random Forest divisualisasikan melalui pohon keputusan, mendukung kinerja model yang lebih baik.

Kata kunci: *churn pelanggan, industri perbankan, Random Forest Classifier, analisis prediktif, akurasi model.*

ABSTRACT (English)

This study is aimed at analyzing and predicting customer churn in the banking industry using machine learning techniques. Factors such as better financial services, lower costs, branch locations, and lower interest rates are identified as the main reasons for customers switching to other banks. To understand customer churn behavior, predictive models based on machine learning, specifically Decision Tree and Random Forest, were employed. The evaluation of model performance shows that higher testing accuracy (83.9%) was achieved by Random Forest compared to Decision Tree (78%), along with better consistency in cross-validation. Furthermore, better capability in predicting churn class with higher recall and f1-score values for class 1 (churned) was demonstrated by Random Forest.

The evaluation results indicate that class 0 (not churned) is predicted more effectively by both models compared to class 1 (churned). However, the superiority in predicting class 1 is shown by Random Forest, which is the main target in churn analysis. The utilization of more complex features and rules by Random Forest is visualized through decision trees, supporting better model performance.

Keywords: *customer churn, banking industry, Random Forest Classifier, predictive analysis, model accuracy.*

UNIVERSITAS
MULTIMEDIA
NUSANTARA

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dalam era globalisasi yang penuh persaingan ketat saat ini, industri perbankan menghadapi tantangan besar dalam menjaga keberlangsungan operasinya [1]. Perubahan dinamis di pasar keuangan, dipicu oleh faktor-faktor seperti teknologi yang terus berkembang, perubahan preferensi konsumen. Pelanggan semakin menuntut layanan yang lebih personal, mudah diakses, dan fleksibel [2]. regulasi yang berubah-ubah, mengharuskan bank untuk beradaptasi dengan cepat agar tetap relevan dan kompetitif

Dalam perjuangan ini, salah satu metrik penting yang harus dipertimbangkan adalah tingkat churn, yang mencerminkan persentase pelanggan yang memilih untuk meninggalkan layanan bank dalam periode waktu tertentu. Nasabah mungkin berpindah ke bank lain karena alasan yang berfluktuasi, misalnya, layanan keuangan yang lebih baik dengan biaya lebih rendah, lokasi cabang bank, suku bunga rendah, dan banyak lainnya [3]. tingkat churn yang tinggi tidak hanya berdampak negatif pada pendapatan bank, tetapi juga dapat mengancam reputasi dan kepercayaan dari pemegang saham serta pasar secara keseluruhan [4].

Melacak dan memahami faktor-faktor yang mempengaruhi keputusan pelanggan untuk meninggalkan layanan bank menjadi krusial dalam menghadapi tantangan ini. Penggunaan teknik analisis data yang canggih, seperti model prediktif berbasis machine learning, menjadi semakin penting dalam upaya memahami perilaku pelanggan dan merumuskan strategi yang efektif untuk meminimalkan churn. investigasi churn pelanggan di industri perbankan menggunakan pendekatan machine learning dan aplikasi visualisasi untuk ilmu data dan manajemen dapat memberikan wawasan yang berharga tentang perilaku churn pelanggan dalam konteks perbankan [8].

Dalam menghadapi tantangan prediksi churn pelanggan di industri perbankan, beberapa penelitian telah mengatasi masalah data yang tidak seimbang

[10]. Data tidak seimbang merupakan situasi di mana jumlah sampel di setiap kelas tidak seimbang, sehingga dapat mempengaruhi kinerja model machine learning. Penanganan data tidak seimbang menjadi faktor penting dalam memastikan keakuratan model prediksi churn pelanggan.

Dengan memanfaatkan pendekatan yang didukung oleh data, bank dapat mengidentifikasi pola-pola yang tersembunyi dan faktor-faktor kritis yang mempengaruhi keputusan pelanggan[9], sehingga memungkinkan tindakan pencegahan yang proaktif diambil.. Penelitian tentang prediksi churn pelanggan tidak hanya memiliki relevansi bisnis yang langsung, tetapi juga menjadi fondasi bagi inovasi dan strategi yang memungkinkan bank untuk tetap berada di garis depan dalam industri yang berubah dengan cepat ini [5]. Model machine learning terbukti secara efektif memprediksi churn dan membantu bank untuk meningkatkan retensi nasabah[11].

Jika sudah mengetahui pola pola yang tersembunyi dan faktor yang mempengaruhinya seharusnya bagian bank dapat mengambil keputusan untuk mencari solusi yang mengurangi churn mereka Karena berkurangnya pelanggan mereka dapat memperburuk reputasi mereka[7]

Salah satu contoh penerapan Random Forest dalam menganalisis bank churn terdapat pada salah satu skripsi, di mana nasabah di industri perbankan dapat ditemukan. Di sini, penerapan algoritma Random Forest dalam memprediksi churn nasabah telah menjadi fokus penelitian yang signifikan dalam industri perbankan. Dengan keunggulan ansambel pohon keputusan, Random Forest mampu meningkatkan akurasi prediksi dan mengatasi masalah overfitting. Penelitian ini bertujuan untuk memperluas pemahaman tentang efektivitas Random Forest dalam memprediksi churn nasabah dengan mengevaluasi kinerja model menggunakan partisi data uji dan latih yang berbeda. Hasilnya diharapkan memberikan wawasan berharga bagi perusahaan perbankan dalam mengidentifikasi dan mengurangi tingkat churn nasabah.[18].

1.2 Rumusan Masalah

Berdasarkan latar belakang penelitian di atas, adapun rumusan masalah yang akan dijawab melalui penelitian ini antara lain:

1. Bagaimana akurasi penggunaan *Random Forest Classifier* dalam memprediksi churn pelanggan?
2. Apa saja features yang mempengaruhi churn pada Bank menggunakan *Random Forest Classifier*?
3. Bagaimana kinerja model prediksi churn dalam *Random Forest Classifier* dalam *Precision, recall, F1 Score*, dan *Support* ?

1.3 Batasan Masalah

Berdasarkan identifikasi masalah serta dengan mempertimbangkan banyak aspek seperti waktu, kemampuan peneliti, dan kepentingan penelitian, maka permasalahan dibatasi pada hal-hal sebagai berikut:

1. Pengklasifikasian *Churn* hanya didasarkan dari satu sumber dataset dan menggunakan variabel di dalam dataset tersebut.
2. Penelitian ini hanya fokus untuk mengklasifikasi jenis *Churn* yaitu *Exited* dan *Not-Exited*.
3. Analisis hanya akan dilakukan menggunakan metode Decision Tree Random Forest, tanpa mempertimbangkan metode prediksi lainnya.

1.4 Tujuan dan Manfaat Penelitian

Tujuan dan manfaat dari penelitian ini adalah sebagai berikut:

1.4.1 Tujuan Penelitian

1. Menilai akurasi penggunaan *Random Forest Classifier* dalam memprediksi churn pelanggan di industri perbankan.

2. Mengidentifikasi dan menganalisis fitur-fitur yang paling mempengaruhi churn pelanggan menggunakan Random Forest Classifier.
3. Mengevaluasi kinerja model prediksi churn dengan menggunakan *Random Forest Classifier* berdasarkan *Precision*, *recall*, *F1 Score*, dan *Support*.

1.4.2 Manfaat Penelitian

Adapun beberapa manfaat yang diharapkan dapat terwujud melalui penelitian ini bagi pembaca dan peneliti, antara lain:

a. Manfaat Praktis

1. Membantu pihak bank dalam mengambil keputusan yang lebih baik dan cerdas untuk mengurangi churn pelanggan.

b. Manfaat Teoritis

1. Mengevaluasi model klasifikasi yang dibentuk menggunakan algoritma Random Forest Classifier untuk mengklasifikasikan churn pelanggan di industri perbankan.
2. Mengidentifikasi faktor-faktor yang berpengaruh dalam pembuatan model klasifikasi churn pelanggan di bank.

BAB II

LANDASAN TEORI

2.1 Telaah Literatur

2.1.1 Bank Churn

Churn, atau tingkat pergantian pelanggan, adalah masalah serius yang dihadapi oleh industri perbankan di era globalisasi yang penuh persaingan. Dalam beberapa tahun terakhir, penelitian tentang churn bank telah menjadi topik yang semakin penting karena dampaknya terhadap pendapatan dan reputasi bank [12].

Menganalisa atau mengidentifikasi faktor-faktor sebagai penyebab utama churn pelanggan dalam industri perbankan seperti kepuasan pelanggan, kualitas layanan, dan faktor lainnya yang berkaitan dengan pengalaman pelanggan [6]. Kesimpulan nya analisis churn pelanggan melibatkan penelitian tentang faktor-faktor yang menjadi penyebab utama churn pelanggan, seperti kepuasan pelanggan, kualitas layanan, dan faktor lain yang terkait dengan pengalaman pelanggan.

2.1.2 Machine Learning

Machine Learning (ML) adalah cabang dari kecerdasan buatan (Artificial Intelligence) yang memungkinkan sistem untuk mempelajari pola atau informasi dari data yang ada tanpa perlu secara eksplisit diprogram [19]. Dengan menggunakan algoritma dan model matematika, mesin mampu mengidentifikasi pola-pola kompleks dalam data dan menghasilkan keputusan atau prediksi yang bermanfaat [16].

Dalam definisi ini, terdapat penekanan pada kemampuan mesin untuk mempelajari dari data yang ada tanpa memerlukan instruksi eksplisit dari programmer. Hal ini dilakukan melalui proses pengembangan algoritma dan model matematika yang memungkinkan mesin untuk

mengenali pola dalam data dan melakukan tugas-tugas tertentu, seperti membuat prediksi atau mengambil keputusan [20].

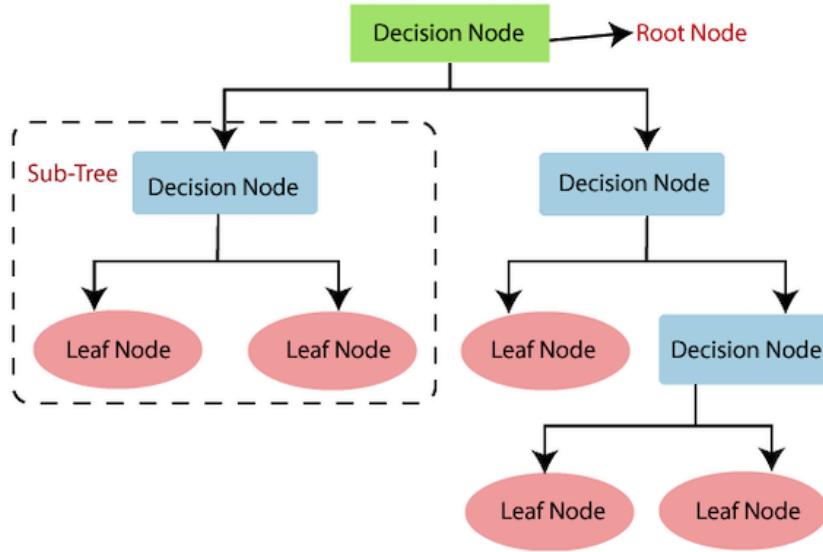
2.1.3 Supervised Learning

Supervised Learning adalah salah satu pendekatan utama dalam Machine Learning di mana model mempelajari hubungan antara input dan output dari data yang diberikan [21]. Dalam konteks ini, input juga dikenal sebagai fitur atau atribut, sedangkan output biasanya disebut sebagai label atau target.

Dalam Supervised Learning, model dilatih menggunakan dataset yang telah diberi label, yang berarti setiap contoh dalam dataset memiliki pasangan input-output yang sesuai [21][22]. Tujuan utama dari Supervised Learning adalah untuk menghasilkan fungsi yang dapat memetakan setiap input ke output yang sesuai dengan akurasi yang tinggi. Model ini kemudian dapat digunakan untuk memprediksi output untuk data baru yang belum pernah dilihat sebelumnya.

2.1.4 Decision Tree

Decision tree merupakan salah satu algoritma klasifikasi yang paling populer dan telah digunakan secara luas dalam berbagai bidang, termasuk ilmu komputer, statistik, dan kecerdasan buatan [13]. Algoritma ini bertujuan untuk membangun model prediktif dalam bentuk struktur pohon keputusan, di mana setiap simpul dalam pohon tersebut mewakili suatu keputusan atau prediksi berdasarkan pada serangkaian aturan yang didefinisikan [14].



Gambar 2.1.1 Struktur Decision Tree [14]

Keputusan dalam algoritma Pohon Keputusan didasarkan pada dua konsep perhitungan, yaitu *entropy* dan *information gain*. *Entropy* mengukur tingkat ketidakmurnian atau kecacuan data, dengan nilai yang berkisar antara 0 hingga 1. Semakin mendekati nol, ini menunjukkan bahwa data sangat teratur dan setiap atribut dalam data memiliki *class* yang seragam. Berikut adalah rumus daripada entropy :

$$Entropy(S) = \sum_{i=1}^c P_i \log_2 P_i$$

Di mana:

- S adalah himpunan data yang sedang dipertimbangkan.
- c adalah jumlah kelas atau label yang mungkin dalam himpunan data.
- P_i adalah proporsi frekuensi relatif dari setiap kelas i dalam himpunan data S

Sedangkan *information gain* merupakan pengukuran perbedaan entropi pada database yang telah dilakukan pemisahan, hal ini bertujuan untuk menentukan segmentasi feature pada setiap node saat pembuatan

model. Adapun rumus daripada perhitungan information gain adalah sebagai berikut:

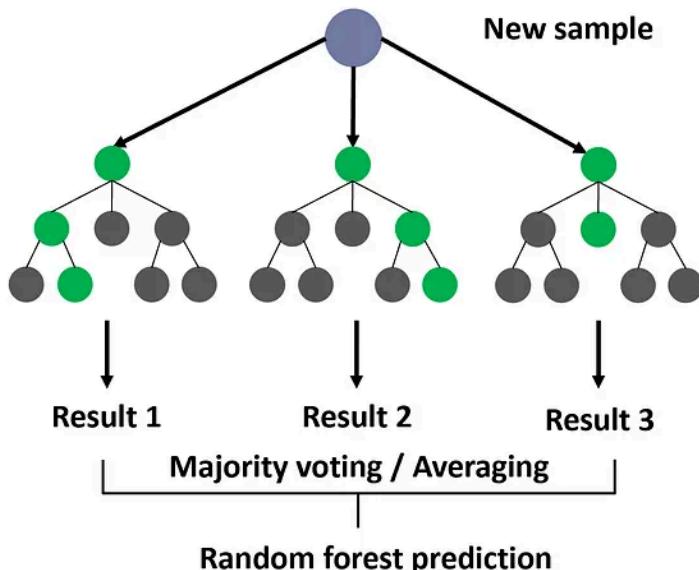
$$\text{InformationGain} = \text{Entropy}(S) - \sum_{i=1}^n \frac{N_i}{N} \cdot \text{Entropy}(S_i)$$

- S adalah himpunan data sebelum pemisahan.
- n adalah jumlah subset yang dihasilkan setelah pemisahan.
- N_i adalah jumlah sampel dalam subset S_i .
- N adalah jumlah total sampel dalam himpunan data sebelum pemisahan.
- $\text{Entropy}(S)$ adalah entropi sebelum pemisahan.
- $\text{Entropy}(S_i)$ adalah entropi dari setiap subset S_i setelah pemisahan.

Algoritma decision tree dapat digunakan untuk membangun model prediktif yang dapat mengidentifikasi pola-pola perilaku pelanggan yang berpotensi menyebabkan churn [15]. Dengan membagi dataset pelanggan menjadi subset-subset yang lebih kecil berdasarkan pada atribut-atribut yang relevan, seperti preferensi layanan, frekuensi transaksi, atau tingkat kepuasan, decision tree dapat membantu bank dalam mengidentifikasi kelompok-kelompok pelanggan yang berisiko tinggi untuk meninggalkan layanan.

2.1.5 Random Forest

Konsep dasar dari Random Forest adalah membangun banyak pohon keputusan secara acak dari subset data yang berbeda, dan kemudian menggabungkan hasil prediksi dari semua pohon tersebut [17]. Secara khusus, Random Forest mengimplementasikan dua jenis randomness: randomness dalam pemilihan sampel data yang digunakan untuk melatih setiap pohon keputusan, dan randomness dalam pemilihan atribut yang digunakan untuk membagi setiap simpul dalam pohon keputusan.



Gambar 2.1.2 Visualisasi Random Forest [16]

Selain itu, Random Forest juga memiliki kemampuan untuk mengatasi ketidakseimbangan kelas yang sering terjadi dalam dataset churn bank, di mana jumlah pelanggan yang churn mungkin jauh lebih sedikit daripada yang tidak churn. Dengan mempertimbangkan sejumlah besar pohon keputusan yang berbeda, Random Forest dapat menghasilkan prediksi yang lebih seimbang dan dapat diandalkan.

2.1.6 Confusion Matrix

Confusion matrix adalah tabel yang digunakan untuk mengevaluasi kinerja model klasifikasi dengan membandingkan prediksi model dengan nilai aktual dari data [25]. Confusion matrix memiliki empat sel, yaitu

- A. True Positive (TP)

Merupakan kasus di mana model dengan benar memprediksi bahwa suatu sampel adalah positif (misalnya, pelanggan yang melakukan churn) dan prediksi tersebut sesuai dengan kebenaran.

B. True Negative (TN)

Merupakan kasus di mana model dengan benar memprediksi bahwa suatu sampel adalah negatif (misalnya, pelanggan yang tidak melakukan churn) dan prediksi tersebut sesuai dengan kebenaran.

C. False Positive (FP)

Merupakan kasus di mana model salah memprediksi bahwa suatu sampel adalah positif (misalnya, model memprediksi bahwa pelanggan tidak akan churn, padahal sebenarnya mereka akan churn).

D. False Negative (FN)

Merupakan kasus di mana model salah memprediksi bahwa suatu sampel adalah negatif (misalnya, model memprediksi bahwa pelanggan akan churn, padahal sebenarnya mereka tidak akan churn).

Dari confusion matrix, kita dapat menghitung berbagai metrik evaluasi kinerja model seperti presisi, dan recall

$$\text{Precision} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Positive (FP)}}$$

Precision dihitung dengan membagi jumlah True Positive oleh jumlah keseluruhan prediksi positif yang dibuat oleh model. Precision memberikan gambaran tentang seberapa akurat model dalam mengklasifikasikan hasil positif. Semakin tinggi nilai precision, semakin sedikit False Positive yang dihasilkan oleh model, menunjukkan bahwa model lebih konservatif dalam membuat prediksi positif.

$$\text{Recall} = \frac{\text{True Positive (TP)}}{\text{True Positive (TP)} + \text{False Negative (FN)}}$$

Recall dihitung dengan membagi jumlah True Positive oleh jumlah keseluruhan instance positif yang seharusnya diidentifikasi oleh model. Ini menunjukkan seberapa baik model dapat mengidentifikasi semua instance positif dalam dataset. Semakin tinggi nilai recall, semakin sedikit False

Negative yang dihasilkan oleh model, menunjukkan kemampuan model dalam mengenali sebagian besar instance positif yang ada.

2.1.7 F1-Score

F1-Score merupakan matrik evaluasi kinerja model yang menggabungkan presisi dan recall model [26]. F-score adalah rata-rata harmonik dari presisi dan recall, dan sering digunakan untuk mengukur keseimbangan antara kedua metrik tersebut [26]. F-score memberikan gambaran yang lebih baik tentang kinerja model klasifikasi, terutama ketika kelas yang tidak seimbang dalam distribusi frekuensinya.

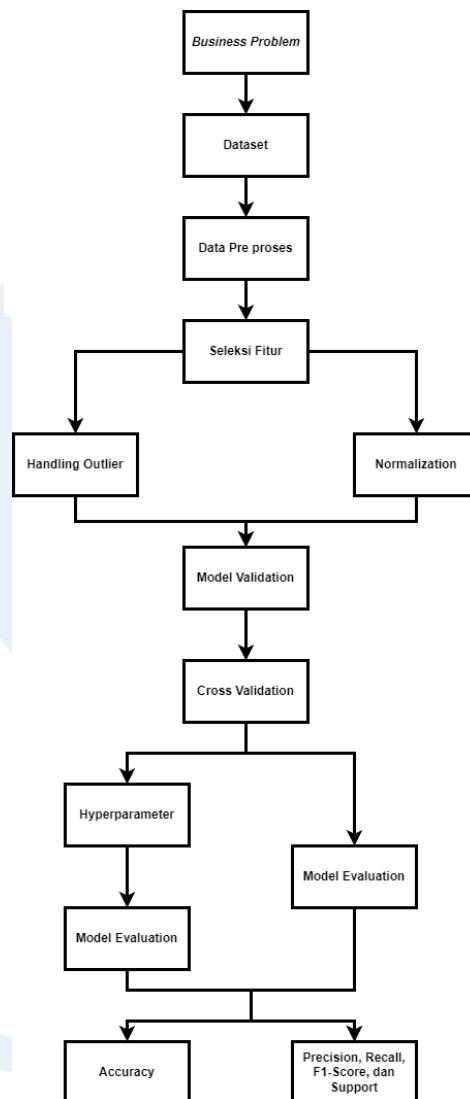
$$F1\ Score = 2 \times \frac{Precision \times Recall}{Precision + Recall}$$

F1 Score memberikan gambaran keseimbangan antara precision dan recall model. Semakin tinggi nilai F1 Score, semakin baik keseimbangan antara precision dan recall. Metrik ini cocok digunakan ketika ingin mempertimbangkan false positives dan false negatives secara seimbang.



BAB III

METODOLOGI PENELITIAN



Gambar 3.1 Metodologi Penelitian

Dalam studi ini, peneliti menjalankan sejumlah langkah dan proses untuk mengidentifikasi churn dalam industri perbankan. Tahapan awal melibatkan pengenalan masalah, yang mencakup pemahaman tentang perpindahan pelanggan dan dampak negatifnya terhadap kesehatan finansial perusahaan. Kesadaran akan masalah ini memicu kebutuhan akan solusi yang efektif untuk mendeteksi dan mencegah churn. Oleh karena itu, sebuah penelitian yang merinci

langkah-langkahnya dari awal hingga akhir menjadi suatu kebutuhan yang mendesak.

3.1 Gambaran Umum Dataset Penelitian

Dataset yang digunakan dalam penelitian ini adalah "Churn for Bank Customers" yang disusun oleh Mehmet Akturk dan diperbarui empat tahun yang lalu. Dataset ini dapat diakses melalui tautan: <https://www.kaggle.com/datasets/mathchi/churn-for-bank-customers>. Dataset ini menyediakan informasi yang komprehensif tentang berbagai variabel yang dapat mempengaruhi keputusan pelanggan untuk meninggalkan layanan bank.

Dataset ini berisi 10.000 entri dan mencakup informasi tentang berbagai variabel, termasuk:

No	Nama Atribut	Tipe Data	Deskripsi
1	Nama Pelanggan (Surname)	Object	yang tidak memiliki pengaruh terhadap keputusan pelanggan untuk meninggalkan bank.
2	Skor kredit (CreditScore)	Integer 64	yang dapat mempengaruhi keputusan churn pelanggan karena pelanggan dengan skor kredit yang lebih tinggi cenderung lebih sedikit meninggalkan bank.
3	Geografi (Geography)	Object	lokasi pelanggan dapat

			mempengaruhi keputusan mereka untuk meninggalkan bank.
4	Jenis kelamin (Gender)	Object	yang menarik untuk dieksplorasi apakah berperan dalam pelanggan meninggalkan bank.
5	Usia (Age)	Integer 64	relevan karena pelanggan yang lebih tua cenderung lebih setia dan kurang cenderung meninggalkan bank.
6	Masa jabatan (Tenure)	Integer 64	mengacu pada jumlah tahun bahwa pelanggan telah menjadi klien bank. Biasanya, klien yang lebih tua lebih setia dan kurang cenderung meninggalkan bank.
7	Saldo (Balance)	Float64	merupakan indikator yang sangat baik untuk churn pelanggan, karena orang dengan saldo yang lebih tinggi di

			rekening mereka kurang cenderung meninggalkan bank dibandingkan dengan yang dengan saldo lebih rendah.
8	Jumlah produk (NumOfProducts)	Integer 64	mengacu pada jumlah produk yang dibeli oleh pelanggan melalui bank.
9	Memiliki kartu kredit (HasCrCard)	Integer 64	menunjukkan apakah pelanggan memiliki kartu kredit atau tidak. Kolom ini juga relevan, karena orang dengan kartu kredit kurang cenderung meninggalkan bank.
10	Pelanggan aktif (IsActiveMember)	Integer 64	pelanggan yang aktif cenderung kurang meninggalkan bank.
11	Gaji yang diperkirakan (EstimatedSalary)	Float 64	seperti saldo, orang dengan gaji lebih rendah lebih cenderung meninggalkan bank dibandingkan dengan yang dengan gaji lebih tinggi.

12	Exited	Integer 64	menunjukkan apakah pelanggan telah meninggalkan bank atau tidak.
----	--------	------------	--

Tabel 3.1.1. Eksplorasi Dataset

Variabel-variabel ini memberikan landasan yang kuat untuk menganalisis perilaku churn pelanggan dan merumuskan strategi yang sesuai untuk meminimalkan churn dalam industri perbankan. Dengan memanfaatkan dataset ini dan menggabungkannya dengan metode analisis yang tepat, diharapkan penelitian ini akan memberikan wawasan yang berharga bagi industri perbankan dalam menghadapi tantangan churn pelanggan.

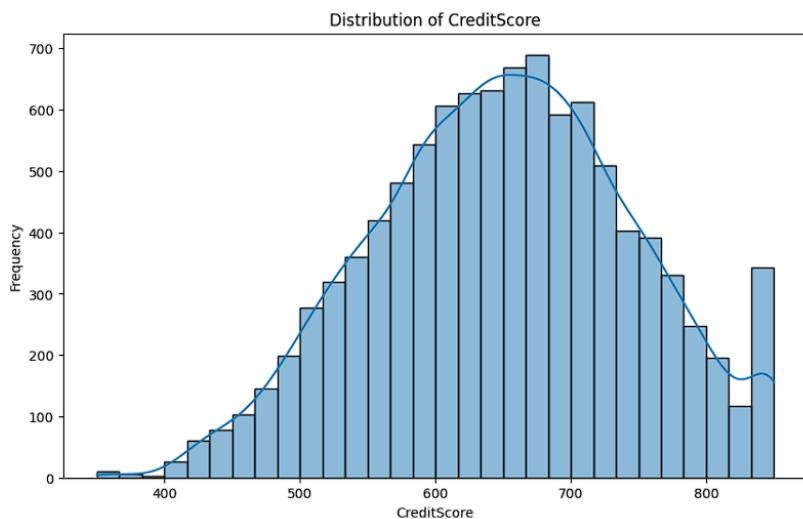
3.2 Exploratory Data Analysis

Pada tahap Exploratory Data Analysis (EDA), dataset "Churn for Bank Customers" akan diperiksa untuk memahami karakteristiknya. Melalui analisis ini, pola-pola dan hubungan antar variabel yang relevan untuk strategi mengurangi churn pelanggan akan diidentifikasi. Bantuan visualisasi data akan digunakan untuk memperoleh pemahaman intuitif tentang distribusi dan pola data yang signifikan.

3.2.1 Distribusi Credit Score

```
plt.figure(figsize=(10, 6))
sns.histplot(data=churn, x='CreditScore', bins=30, kde=True)
plt.title('Distribution of CreditScore')
plt.xlabel('CreditScore')
plt.ylabel('Frequency')
plt.show()
```

Gambar 3.2.1.1 Code Distribusi Credit Score



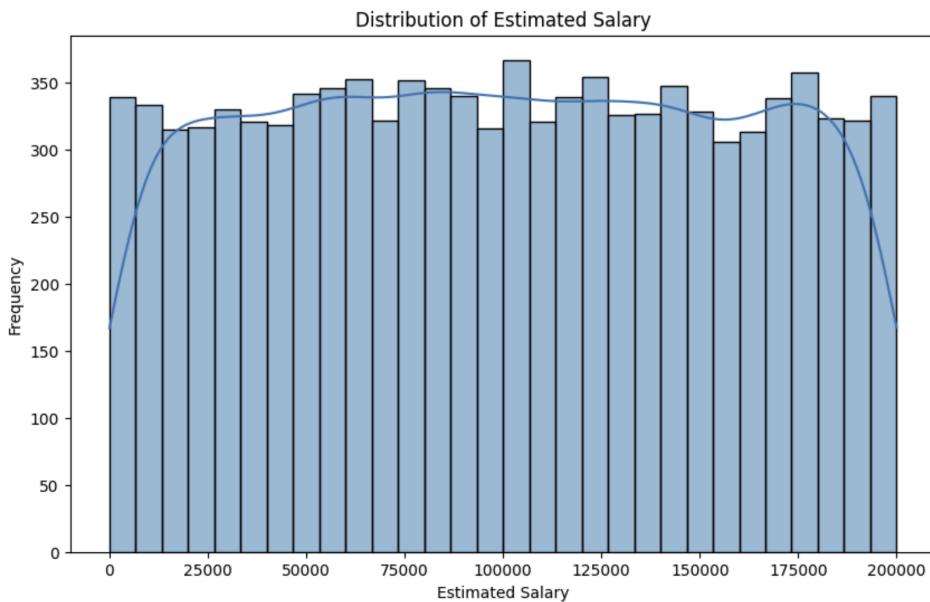
Gambar 3.2.1.2 Histogram Distribusi Credit Score

Pada tahap ini, dilakukan visualisasi terhadap 'CreditScore' dengan menganalisis grafik distribusi. Berdasarkan grafik distribusi tersebut, teramatinya bahwa puncak grafik berada di sekitar rentang 600 hingga 700, menunjukkan distribusi terbesar terkonsentrasi di antara nilai tersebut. Namun, ditemukan satu puncak grafik yang menonjol pada angka 850, mengindikasikan adanya outlier yang signifikan pada bagian tersebut.

3.2.2 Distribusi Estimated Salary

```
plt.figure(figsize=(10, 6))
sns.histplot(data=churn, x='EstimatedSalary', bins=30, kde=True)
plt.title('Distribution of Estimated Salary')
plt.xlabel('Estimated Salary')
plt.ylabel('Frequency')
plt.show()
```

Gambar 3.2.2.1 Code Distribusi Estimated Salary



Gambar 3.2.2.2 Histogram Distribusi Estimated Salary

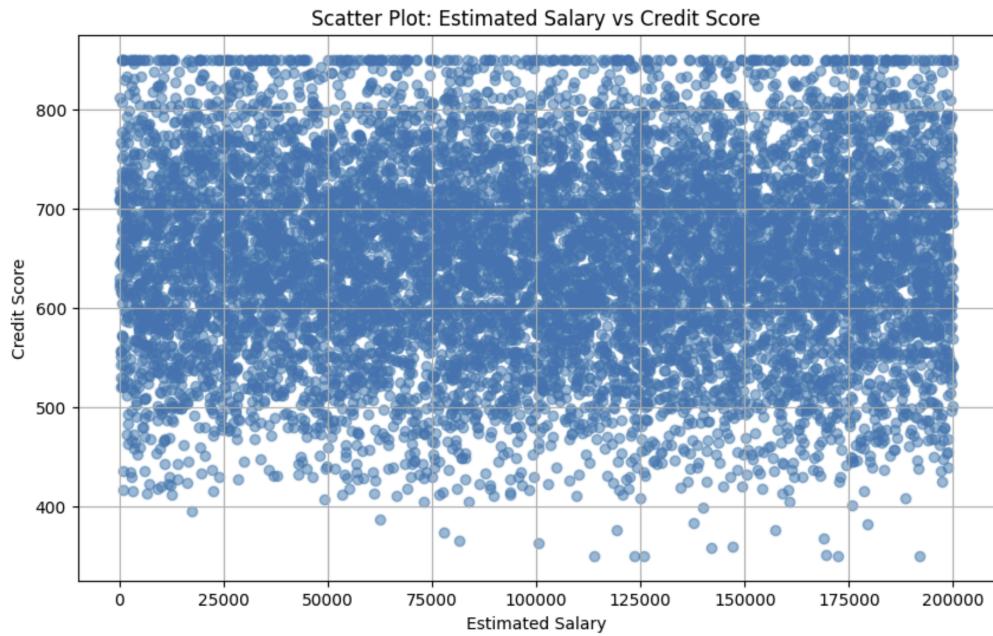
Grafik ini menunjukkan distribusi gaji diperkirakan di Amerika Serikat. Sumbu X menunjukkan gaji diperkirakan dalam rentang dari 0 hingga 200.000 dolar. Sumbu Y menunjukkan frekuensi individu dengan gaji diperkirakan dalam setiap rentang.

3.2.3 Distribusi Scatter Plot Estimated Salary Vs Credit Score

```
balance = churn['Balance']
estimated_salary = churn['EstimatedSalary']
creditscore = churn['CreditScore']

plt.figure(figsize=(10, 6))
plt.scatter(estimated_salary, creditscore, alpha=0.5)
plt.title('Scatter Plot: Estimated Salary vs Credit Score')
plt.xlabel('Estimated Salary')
plt.ylabel('Credit Score')
plt.grid(True)
plt.show()
```

Gambar 3.2.3.1 Code Distribusi Scatter Plot Estimated Salary Vs Credit Score



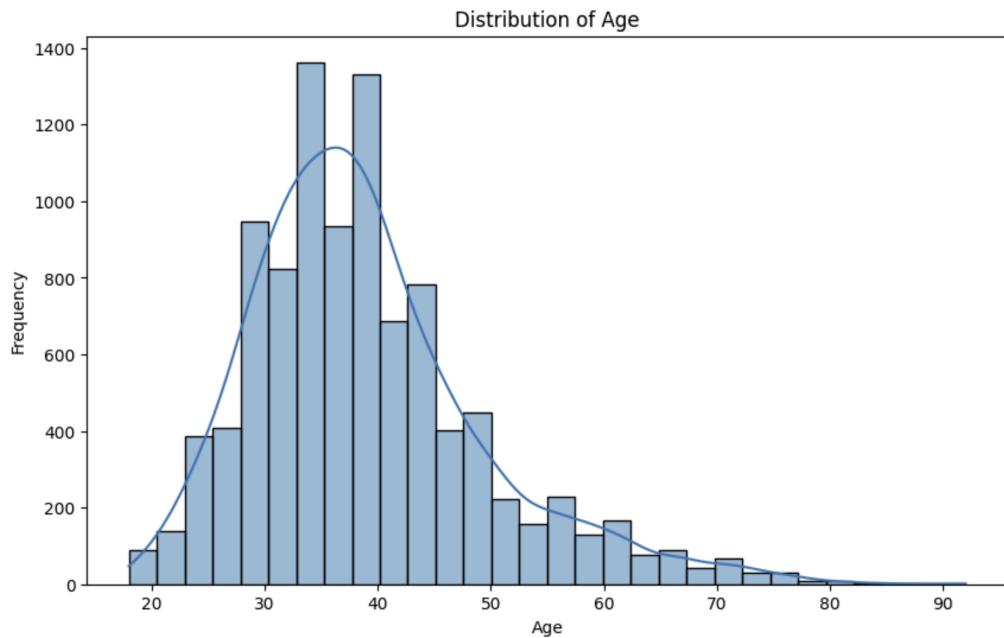
Gambar 3.2.3.2 Scatter Plot Estimated Salary Vs Credit Score

Grafik ini menunjukkan hubungan antara gaji diperkirakan dan skor kredit di Amerika Serikat. Sumbu X menunjukkan gaji diperkirakan dalam rentang dari 0 hingga 200.000 dolar. Sumbu Y menunjukkan skor kredit dalam rentang dari 300 hingga 850.

3.2.4 Distribusi Age

```
plt.figure(figsize=(10, 6))
sns.histplot(data=churn, x='Age', bins=30, kde=True)
plt.title('Distribution of Age')
plt.xlabel('Age')
plt.ylabel('Frequency')
plt.show()
```

Gambar Code 3.2.4.1 Distribusi Age



Gambar 3.2.4.2 Histogram Distribusi Age

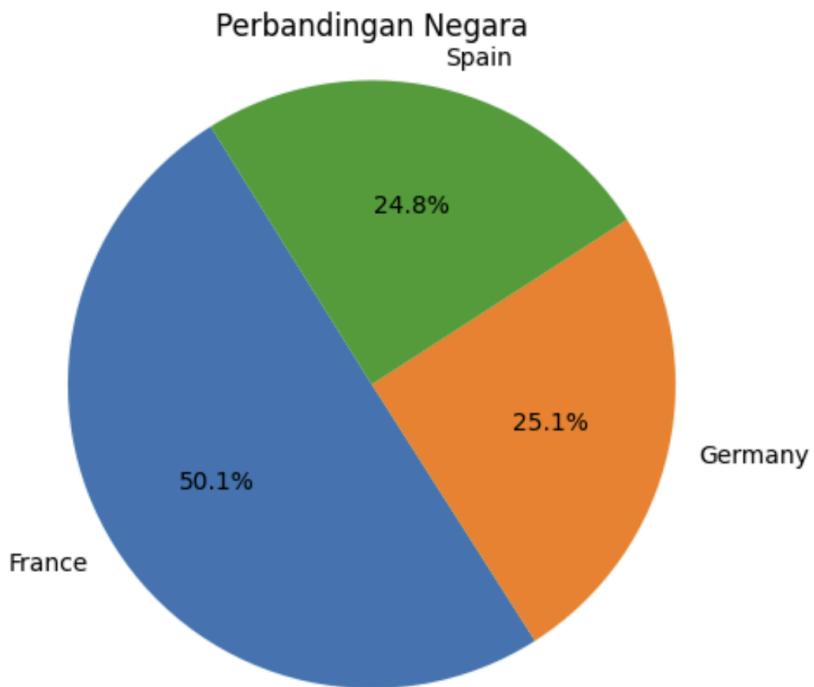
Visualisasi ini menunjukkan distribusi umur penduduk di Indonesia pada tahun 2020. Sumbu X menunjukkan umur dalam rentang dari 0 hingga 100 tahun. Sumbu Y menunjukkan persentase penduduk dengan umur dalam setiap rentang.

3.2.5 Distribusi Perbandingan Negara

```
column_to_visualize = 'Geography'
value_counts = churn[column_to_visualize].value_counts()

# Plot pie chart
plt.figure(figsize=(8, 5))
plt.pie(value_counts, labels=value_counts.index, autopct='%.1f%%', startangle=122)
plt.title('Perbandingan Negara')
plt.axis('equal') # Memastikan pie chart terlihat Lingkaran
plt.show()
```

Gambar 3.2.5.1 Code Distribusi Perbandingan Negara



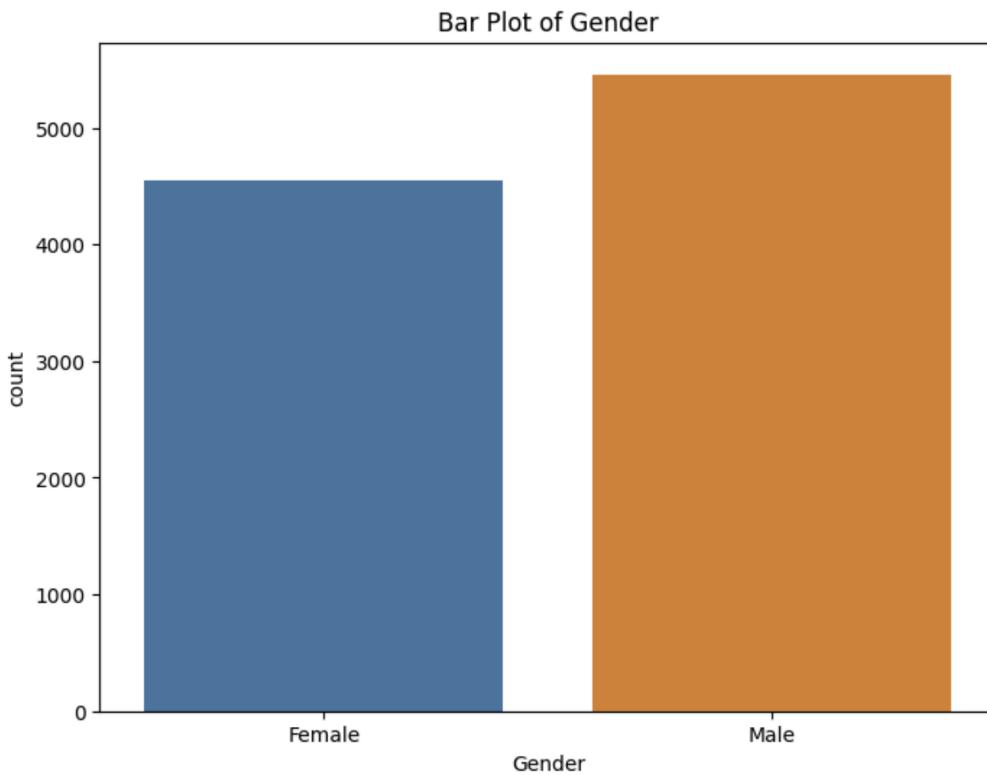
Gambar 3.2.5.2 Pie Chart Distribusi Perbandingan Negara

Pie chart ini menunjukkan perbandingan persentase pekerja asing di Spanyol, Jerman, dan Prancis. Persentase dihitung berdasarkan total populasi di masing-masing negara.

3.2.6 Distribusi Barplot Gender

```
: plt.figure(figsize=(8, 6))
sns.countplot(x='Gender', data=churn)
plt.title('Bar Plot of Gender')
plt.show()
```

Gambar 3.2.6.1 Code Distribusi Barplot Gender



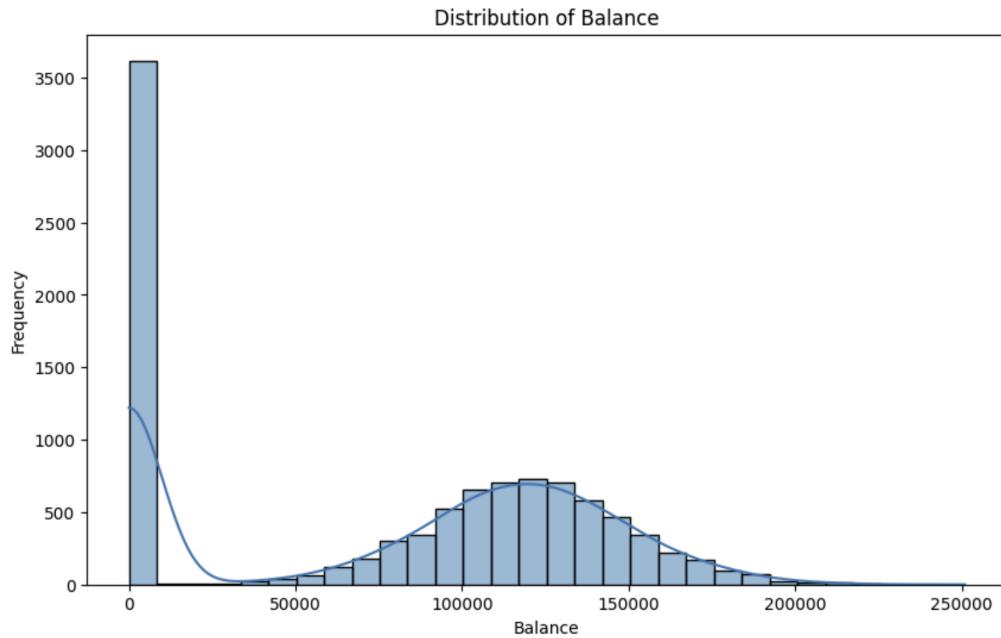
Gambar 3.2.6.2 Barplot Distribusi Barplot Gender

Diagram batang ini menunjukkan distribusi jenis kelamin karyawan di sebuah perusahaan. Persentase karyawan adalah sebagai berikut: Pria: 43,5% dan Wanita: 56,5%, berdasarkan total keseluruhan karyawan di perusahaan tersebut.

3.2.7 Distribusi Balance

```
plt.figure(figsize=(10, 6))
sns.histplot(data=churn, x='Balance', bins=30, kde=True)
plt.title('Distribution of Balance')
plt.xlabel('Balance')
plt.ylabel('Frequency')
plt.show()
```

Gambar 3.2.7.1 Code Distribusi Balance



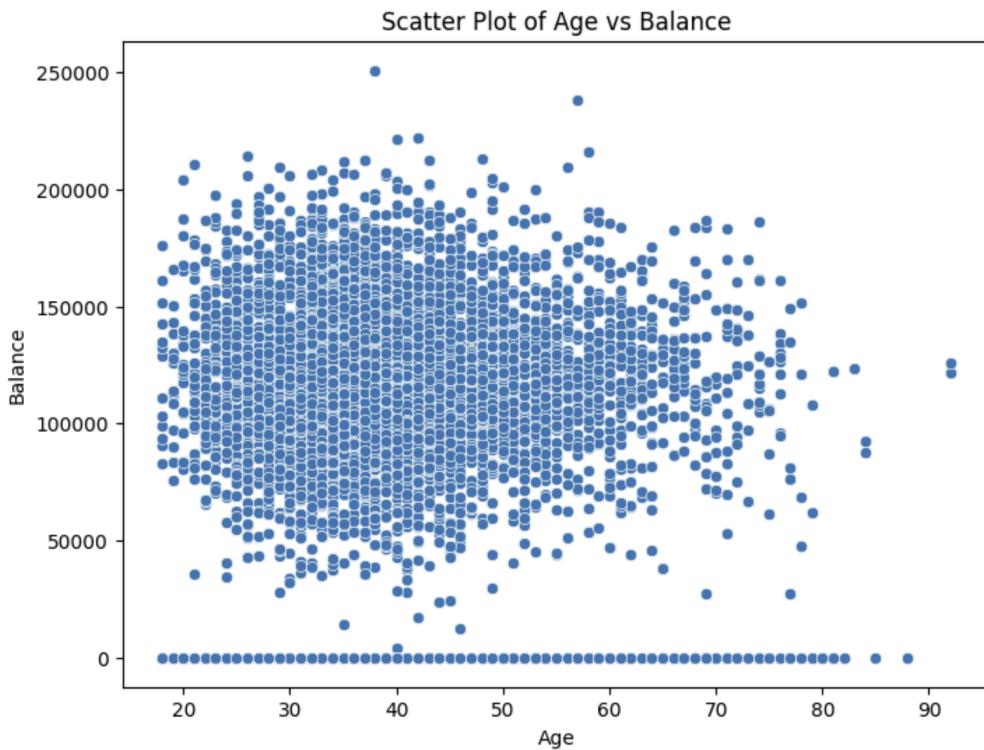
Gambar 3.2.7.2 Histogram Distribusi Balance

Distribusi neraca perusahaan menunjukkan bahwa mayoritas perusahaan memiliki neraca dalam kisaran 50.000 hingga 200.000 juta dolar. Neraca rata-rata perusahaan adalah sekitar 125.000 juta dolar. Terdapat outlier dengan nilai neraca sebesar 250.000 juta dolar yang memerlukan investigasi lebih lanjut.

3.2.8 Distribusi Scatter Plot Age Vs Balance

```
: plt.figure(figsize=(8, 6))
sns.scatterplot(x='Age', y='Balance', data=churn)
plt.title('Scatter Plot of Age vs Balance')
plt.xlabel('Age')
plt.ylabel('Balance')
plt.show()
```

Gambar 3.2.7.1 Code Distribusi Scatter Plot Age Vs Balance



Gambar 3.2.7.2 Distribusi Scatter Plot Age Vs Balance

Scatter plot ini menunjukkan hubungan antara umur dan gaji di Indonesia. Sumbu X menunjukkan umur dalam rentang dari 20 hingga 60 tahun. Sumbu Y menunjukkan gaji dalam rentang dari 2.000 hingga 15.000 ribu rupiah.

3.3 Preprocess Data

3.3.1 Checking Missing Value

```
null_counts = churn.isnull().sum()
print(null_counts)
```

	Count
RowNumber	0
CustomerId	0
Surname	0
CreditScore	0
Geography	0
Gender	0
Age	0
Tenure	0
Balance	0
NumOfProducts	0
HasCrCard	0
IsActiveMember	0
EstimatedSalary	0
Exited	0
dtype: int64	

Gambar 3.3.1.1 Hasil Checking Missing Value

Hasil pemeriksaan data menunjukkan bahwa dataset tidak mengandung nilai yang hilang, sesuai dengan keluaran `churn.isnull().sum()` Dengan demikian, dataset telah dipastikan lengkap, memungkinkan untuk dilanjutkan ke tahap pembersihan outlier dan analisis lebih lanjut. Langkah awal ini penting untuk memastikan integritas data sebelum melanjutkan ke tahapan analisis yang lebih mendalam, sehingga hasil analisis yang dihasilkan dapat diandalkan dan akurat.

3.3.2 Encoding

```
label_encoder = LabelEncoder()
churn['Geography'] = label_encoder.fit_transform(churn['Geography'])
```

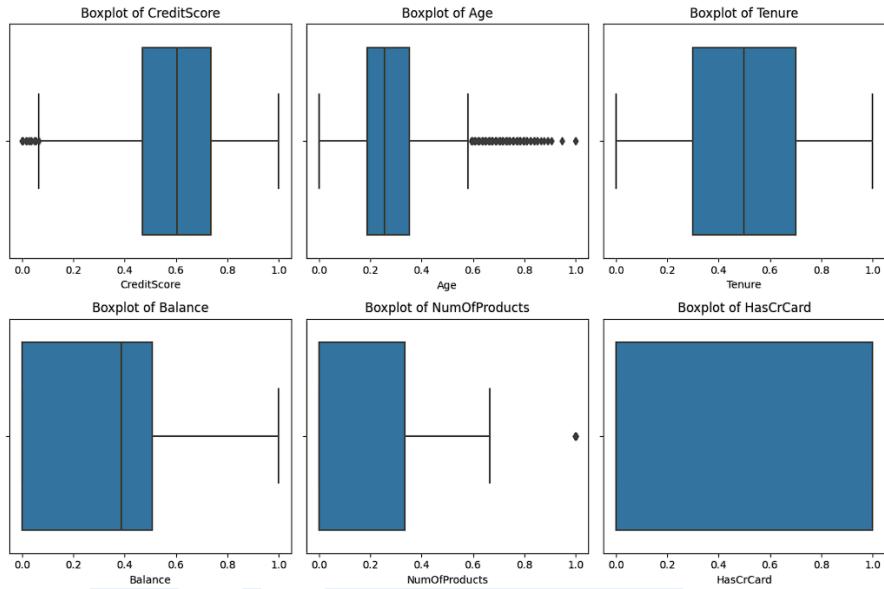
Gambar 3.3.2.1 Code Encoding

Pada langkah tersebut, dilakukan penggunaan LabelEncoder untuk mentransformasi kolom 'Geography' dalam dataframe churn menjadi representasi numerik. Dalam proses ini, nilai-nilai dalam kolom tersebut diubah menjadi bilangan bulat sesuai dengan urutan unik nilai-nilai yang terdapat dalam kolom tersebut. Tujuan dari langkah ini adalah untuk mempersiapkan data kategorikal 'Geography' agar dapat digunakan dalam proses analisis yang membutuhkan input numerik, seperti pemodelan prediktif.

3.3.3 Handling Outlier

```
plt.figure(figsize=(12, 8))
for i, col in enumerate(numerical_columns):
    plt.subplot(2, 3, i % 6 + 1) # Adjusting subplot index to cycle between 1-6
    sns.boxplot(x=churn[col], orient='v')
    plt.title(f'Boxplot of {col}')
    if i % 6 == 5 or i == len(numerical_columns) - 1: # Add plt.show() after every 6 subplots
        plt.tight_layout()
        plt.show()
```

Gambar 3.3.3.1 Code Checking Outlier



Gambar 3.3.3.2 Boxplot Outlier

```
In [7]: def detect_outliers_iqr(churn, column):
    Q1 = churn[column].quantile(0.25)
    Q3 = churn[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    outliers = churn[(churn[column] < lower_bound) | (churn[column] > upper_bound)]
    return outliers

outliers_dict = {}
for col in numerical_columns:
    outliers_dict[col] = detect_outliers_iqr(churn, col)

for col, outliers in outliers_dict.items():
    print(f"Outliers for column '{col}':")
    print(outliers)
    print("\n")
```

Outliers for column 'CreditScore':
RowNumber CustomerId Surname CreditScore Geography Gender \
7 8 15656148 Obina 0.052 Germany Female
942 943 15804586 Lin 0.052 France Female
1193 1194 15779947 Thomas 0.026 Spain Female
1405 1406 15612494 Panicucci 0.018 France Female
1631 1632 15685372 Azubuike 0.000 Spain Male
1838 1839 15758813 Campbell 0.000 Germany Male
1962 1963 15692416 Aikenhead 0.016 Spain Female
2473 2474 15679249 Chou 0.002 Germany Female
2579 2580 15597896 Ozoemena 0.030 Germany Male
8154 8155 15791533 Ch'ien 0.034 Spain Male
8723 8724 15809320 Onyekachi 0.000 France Male
8762 8763 15765173 Lin 0.000 France Female
9210 9211 15792650 Watts 0.064 Spain Male

Gambar 3.3.3.3 Hasil Outlier

Pada tahapan ini, dilakukan visualisasi dan deteksi outlier pada data guna memastikan integritas data sebelum dilakukan analisis lebih lanjut. Hal ini dianggap penting agar hasil analisis yang dihasilkan dapat dipercaya dan akurat. Deteksi outlier menggunakan metode IQR (Interquartile Range). Dengan langkah-langkah ini, diharapkan kualitas data dapat ditingkatkan dan potensi bias yang mungkin timbul akibat

data yang tidak sesuai dapat dikurangi. Setelah outlier terdeteksi, langkah selanjutnya adalah menghapusnya.

```
In [8]: def remove_outliers_iqr(churn, column):
    Q1 = churn[column].quantile(0.25)
    Q3 = churn[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    churn_filtered = churn[(churn[column] >= lower_bound) & (churn[column] <= upper_bound)]
    return churn_filtered

churn_cleaned = churn.copy() # Membuat salinan DataFrame untuk keperluan pemrosesan
for col in numerical_columns:
    churn_cleaned = remove_outliers_iqr(churn_cleaned, col)

print("Jumlah outlier yang dihapus:", len(churn) - len(churn_cleaned))
Jumlah outlier yang dihapus: 484
```

Gambar 3.3.3.4 Code Handling Outlier

Pada gambar diatas Setelah deteksi outlier, dilakukan penghapusan outlier dengan menggunakan fungsi `remove_outliers_iqr`. Outlier dihapus dengan mengidentifikasi data yang berada di luar batas bawah (`lower_bound`) dan batas atas (`upper_bound`) yang ditentukan berdasarkan nilai kuartil pertama (Q1) dan kuartil ketiga (Q3) serta jarak antarkuartil (IQR). Data yang berada di luar rentang ini dianggap sebagai outlier dan dihapus dari DataFrame. Pada proses ini, total outlier yang dihapus adalah 484 data.

3.3.4 Normalization

```
from sklearn.preprocessing import MinMaxScaler

scaler = MinMaxScaler()
numerical_columns = ['CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard', 'IsActiveMember', 'EstimatedSalary']
churn[numerical_columns] = scaler.fit_transform(churn[numerical_columns])
```

Gambar 3.3.4.1 Code Normalization

Pada langkah tersebut, dilakukan penskalaan fitur menggunakan `MinMaxScaler` dari pustaka Scikit-Learn. Fitur-fitur numerik seperti '`CreditScore`', '`Age`', '`Tenure`', '`Balance`', '`NumOfProducts`', '`HasCrCard`', '`IsActiveMember`', dan '`EstimatedSalary`' dinormalisasi menggunakan metode penskalaan Min-Max. Normalisasi ini dilakukan untuk mengubah rentang nilai dari setiap fitur sehingga berkisar antara 0 dan 1, dengan mempertahankan proporsi relatif antar-nilai dalam setiap fitur. Hal ini membantu dalam meningkatkan stabilitas dan konvergensi algoritma

pembelajaran mesin, serta memastikan bahwa setiap fitur memberikan kontribusi yang seimbang dalam pemodelan atau analisis data yang akan dilakukan.

```
def remove_outliers_iqr(churn, column):
    Q1 = churn[column].quantile(0.25)
    Q3 = churn[column].quantile(0.75)
    IQR = Q3 - Q1
    lower_bound = Q1 - 1.5 * IQR
    upper_bound = Q3 + 1.5 * IQR
    churn_filtered = churn[(churn[column] >= lower_bound) & (churn[column] <= upper_bound)]
    return churn_filtered

churn_cleaned = churn.copy() # Membuat salinan DataFrame untuk keperluan pemrosesan
for col in numerical_columns:
    churn_cleaned = remove_outliers_iqr(churn_cleaned, col)

print("Jumlah outlier yang dihapus:", len(churn) - len(churn_cleaned))
Jumlah outlier yang dihapus: 484
```

Gambar 3.3.4.2 Handling Outlier

Kode ini menghapus outlier dari kolom numerik dalam DataFrame 'churn' menggunakan metode Interquartile Range (IQR). Pertama, dibuat salinan dari DataFrame untuk diproses. Fungsi `remove_outliers_iqr` menghitung batas bawah dan atas berdasarkan IQR, lalu menyaring data yang berada dalam batas tersebut. Setiap kolom numerik diproses menggunakan fungsi ini. Terakhir, jumlah outlier yang dihapus ditampilkan dengan menghitung selisih antara jumlah baris sebelum dan sesudah pembersihan outlier.

3.3.5 Features Importances

```
: # Initialize Random Forest classifier with n_estimators = 100
model_rf = RandomForestClassifier(n_estimators=100, random_state=42)

# Train the Random Forest modelA
model_rf.fit(X_train, Y_train.values.ravel()) # ravel Y_train to convert it to 1D array

# Calculate feature importances
feature_importances = model_rf.feature_importances_

# Create a DataFrame for feature importances
feature_importance_df = pd.DataFrame({'Feature': X.columns, 'Importance': feature_importances})

# Sort feature importances by importance value in descending order
feature_importance_df = feature_importance_df.sort_values(by='Importance', ascending=False)

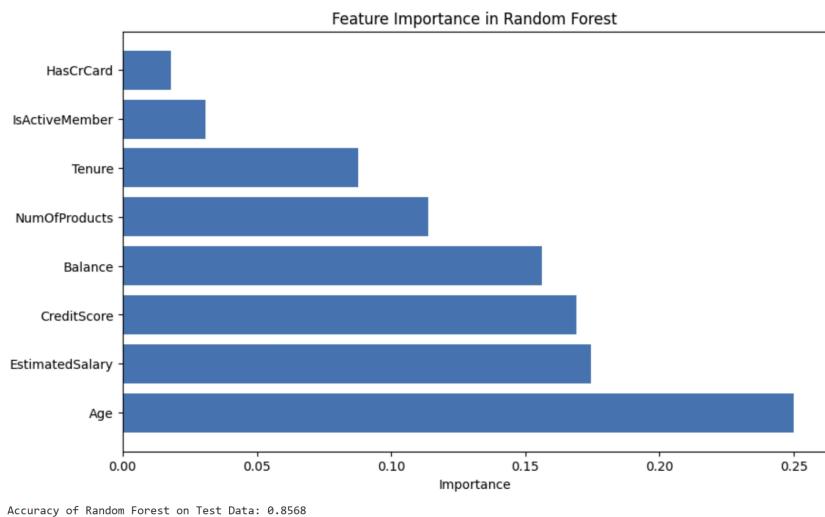
# Visualize feature importances
plt.figure(figsize=(10, 6))
plt.barh(feature_importance_df['Feature'], feature_importance_df['Importance'])
plt.xlabel('Importance')
plt.title('Feature Importance in Random Forest')
plt.show()

# Evaluate accuracy of the Random Forest model on test data
accuracy_rf = model_rf.score(X_test, Y_test)
print(f"Accuracy of Random Forest on Test Data: {accuracy_rf:.4f}")
```

Gambar 3.3.5.1 Code Features Importances

Kode ini menginisialisasi model Random Forest dengan 100 pohon dan melatihnya menggunakan data pelatihan (`X_train` dan `Y_train`). Setelah pelatihan, kode menghitung pentingnya fitur dan menyimpannya dalam DataFrame, yang kemudian diurutkan berdasarkan nilai kepentingan secara menurun. Grafik batang horizontal dibuat untuk memvisualisasikan pentingnya fitur. Terakhir, kode mengevaluasi akurasi model Random Forest pada data uji (`X_test`) dan mencetak hasilnya.

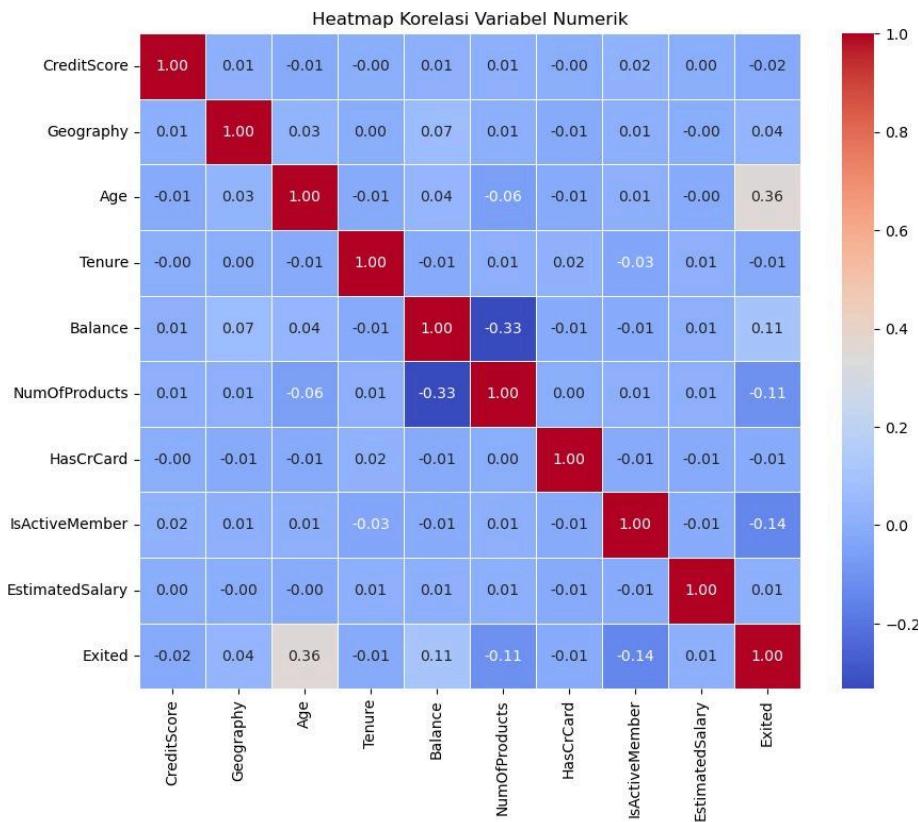
UNIVERSITAS
MULTIMEDIA
NUSANTARA



Gambar 3.3.5.2 Barplot Features Importances

Kesimpulan dari analisis feature importance menggunakan model Random Forest menunjukkan bahwa faktor-faktor paling penting dalam menentukan apakah nasabah akan churn adalah memiliki kartu kredit, merupakan anggota aktif bank, lama menjadi nasabah, jumlah produk bank yang dimiliki, saldo rekening bank, skor kredit, dan perkiraan gaji. Usia nasabah terbukti sebagai faktor yang paling tidak penting. Akurasi model dalam memprediksi churn adalah 85,68%, menandakan kinerja yang cukup baik. Namun, penting untuk diingat bahwa hasil ini spesifik untuk model Random Forest yang digunakan dan dapat berbeda jika model lain diaplikasikan. Feature importance hanya menunjukkan kepentingan fitur dalam model tertentu dan tidak selalu mencerminkan kepentingannya dalam konteks yang lebih luas.

3.3.6 Heat Map



Gambar 3.3.6.1 Hasil Heat map

Kesimpulan dari heatmap korelasi variabel numerik menunjukkan bahwa skor kredit, lama menjadi nasabah, saldo rekening bank, jumlah produk bank yang dimiliki, dan perkiraan gaji saling berkorelasi positif. Skor kredit memiliki korelasi negatif yang lemah dengan usia. Wilayah geografis dan kepemilikan kartu kredit tidak memiliki korelasi kuat dengan variabel lain. Nasabah yang aktif sebagai anggota bank dan memiliki gaji tinggi cenderung lebih kecil kemungkinannya untuk churn. Penting untuk diingat bahwa korelasi tidak berarti kausalitas, dan heatmap ini hanya menunjukkan hubungan antara variabel yang dapat digunakan untuk mengidentifikasi variabel penting dalam model prediksi churn.

3.4 Rekayasa Fitur

3.4.1 Feature Selection

```
churn = churn.drop(columns=['RowNumber', 'CustomerId', 'Surname'])
```

Gambar 3.4.1.1 Code Feature Selection

Pada langkah ini, dilakukan penghapusan kolom 'RowNumber', 'CustomerId', dan 'Surname' dari dataframe churn. Tujuannya adalah untuk menghilangkan kolom-kolom yang tidak diperlukan dalam analisis berikutnya, sehingga fokus dapat lebih ditekankan pada fitur-fitur yang lebih relevan dalam pengembangan model atau analisis data. Dengan demikian, efisiensi proses pemrosesan data dapat ditingkatkan dan hasil analisis dapat lebih terfokus pada informasi yang penting.

3.5 Pemodelan Data

3.5.1 Splitting Train Test

```
X = churn_cleaned[['CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts','HasCrCard','IsActiveMember','EstimatedSalary']]  
Y = churn_cleaned[['Exited']]  
X_train, X_test, Y_train, Y_test = train_test_split(X, Y, test_size=0.2, random_state=42)
```

Gambar 3.5.1.1 Code Splitting Train Test

Pada langkah ini, dilakukan pemisahan data menjadi set pelatihan dan set pengujian menggunakan fungsi `train_test_split`. Kolom-kolom fitur seperti 'CreditScore', 'Age', 'Tenure', 'Balance', 'NumOfProducts', 'HasCrCard', 'IsActiveMember', dan 'EstimatedSalary' digunakan sebagai variabel independen (X), sedangkan kolom 'Exited' digunakan sebagai variabel dependen (Y). Data dibagi dengan proporsi 80% untuk pelatihan (`X_train`, `Y_train`) dan 20% untuk pengujian (`X_test`, `Y_test`), dengan pengacakan ditentukan oleh `random_state=42` untuk memastikan hasil yang konsisten di setiap eksekusi.

3.5.2 Cross Validation

```
model = RandomForestClassifier()  
cv_scores = cross_val_score(model, X_train, Y_train.values.ravel(), cv=5, scoring='accuracy')
```

Gambar 3.5.2.1 Code Cross Validation

Pada langkah ini, dilakukan pelatihan dan evaluasi model menggunakan algoritma RandomForestClassifier. Model dilatih menggunakan data pelatihan (X_{train} dan Y_{train}) dengan metode validasi silang lima lipatan (5-fold cross-validation) untuk mengukur akurasi model. Fungsi `cross_val_score` digunakan untuk menghitung skor akurasi pada setiap lipatan, yang kemudian dirata-rata untuk memberikan estimasi performa model. Validasi silang ini membantu memastikan bahwa model memiliki kemampuan generalisasi yang baik dan tidak overfitting terhadap data pelatihan.

3.5.3 Decision Tree Classifier

```
model = DecisionTreeClassifier(criterion='entropy', random_state=42)  
model.fit(X_train, Y_train)
```

Gambar 3.5.3.1 Code Decision Tree Classifier (entropy)

Pada langkah ini, model Decision Tree Classifier telah dibuat dengan menggunakan kriteria pemisahan berdasarkan entropi (entropy) dan nilai seed random_state sebesar 42 telah ditentukan untuk memastikan reproduktibilitas hasil. Data pelatihan (X_{train} dan Y_{train}) kemudian digunakan untuk melatih model, dan data uji (X_{test}) digunakan untuk melakukan prediksi dengan memanggil metode `predict()`. Dengan demikian, model telah siap untuk dievaluasi kinerjanya.

```
model = DecisionTreeClassifier(max_depth = 10, random_state=42)  
model.fit(X_train, Y_train)
```

Gambar 3.5.3.2 Code Decision Tree Classifier (Max Depth)

Pada langkah ini, model Decision Tree Classifier telah dibuat dengan menetapkan batasan kedalaman maksimum (max_depth) sebesar 10 dan nilai seed random_state sebesar 42 untuk memastikan konsistensi

hasil. Selanjutnya, model dilatih menggunakan data pelatihan (X_train dan Y_train).

3.5.4 Random Forest Classifier Hyperparameter

```
forest_model = RandomForestClassifier(n_estimators=100, max_depth=4,
                                      min_samples_split=2, min_samples_leaf=2,
                                      random_state=42)
forest_model.fit(X_train, Y_train.values.ravel())

forest_pred = forest_model.predict(X_test)
```

Gambar 3.5.4.1 Code Random Forest Classifier Hyperparameter

Pada langkah ini, model Random Forest Classifier dibuat dengan menggunakan 100 pohon keputusan (n_estimators) dan dengan batasan kedalaman maksimum setiap pohon sebesar 4 (max_depth). Selain itu, diterapkan kriteria untuk membagi node internal (min_samples_split) dan jumlah sampel minimum di setiap daun (min_samples_leaf), masing-masing dengan nilai 2. Model dilatih menggunakan data pelatihan (X_train dan Y_train) dengan nilai seed random_state sebesar 42 untuk memastikan hasil yang konsisten. Langkah-langkah ini bertujuan untuk menghasilkan model ensemble yang baik dengan mempertimbangkan sejumlah besar pohon yang lemah dan mengendalikan kompleksitas serta generalisasi model.

3.6 Validasi dan Evaluasi Model

3.6.1 Decision Tree

3.6.1.1 Evaluasi Cross-Validation

```
print("Cross-validation Accuracy Scores:", cv_scores)
print("Mean Accuracy:", cv_scores.mean())
model.fit(X_train, Y_train.values.ravel())

y_pred = model.predict(X_test)
test_accuracy = accuracy_score(Y_test, y_pred)
print("Test Set Accuracy:", test_accuracy)
```

Gambar 3.6.1.1.1 Code Evaluasi Cross-Validation

Pada langkah evaluasi model, dilakukan pencetakan skor akurasi validasi silang (cross-validation) dan rata-rata akurasi dari skor tersebut. Selain itu, model dapat ditentukan ke seluruh set pelatihan dan dievaluasi pada set uji dengan mencetak akurasi prediksi pada data uji. Langkah-langkah ini bertujuan untuk mengevaluasi kinerja model secara menyeluruhan, baik pada data pelatihan maupun pada data uji, untuk memastikan keandalan dan generalisasi model yang dikembangkan.

3.6.1.2 Accuracy Training dan Test Decision Tree

```
# Measure accuracy on the test set
test_accuracy = accuracy_score(Y_test, pred)
print("Decision Tree Accuracy (Test): {:.3f}".format(test_accuracy))

# Measure accuracy on the training set
train_accuracy = accuracy_score(Y_train, model.predict(X_train))
print("Decision Tree Accuracy (Train): {:.3f}".format(train_accuracy))
```

Gambar 3.6.1.2.1 Code Accuracy Training dan Test Decision Tree

Pada gambar ini, dilakukan pengukuran akurasi model Decision Tree pada set data uji dan data pelatihan. Akurasi model diukur dengan menggunakan metrik akurasi, yang mengukur proporsi prediksi yang benar dibandingkan dengan jumlah total prediksi. Langkah ini bertujuan untuk mengevaluasi seberapa baik model Decision Tree mampu menggeneralisasi pola dari data yang tidak terlihat sebelumnya (data uji) dan seberapa baik model dapat mempelajari pola dari data pelatihan.

3.6.1.3 Evaluasi Metrics

```
print("Classification Report:")
print(classification_report(Y_test, pred))
```

Gambar 3.6.1.3.1 Code Evaluasi Metrics

Pada langkah ini, dilaku kan pencetakan laporan klasifikasi yang menyajikan sejumlah metrik evaluasi seperti presisi, recall, dan F1-score untuk setiap kelas target, serta nilai rata-rata secara keseluruhan. Laporan klasifikasi ini memberikan informasi yang lebih rinci tentang kinerja model, memungkinkan untuk mengevaluasi seberapa baik model mampu mengklasifikasikan setiap kelas dengan benar.

3.6.1.4 Visualisasi Confusion Matrix

```
conf_matrix = confusion_matrix(Y_test, pred)
plt.figure(figsize=(10, 7))
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=['Not Exited', 'Exited'], yticklabels=['Not Exited', 'Exited'])
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix')
plt.show()
```

Gambar 3.6.1.4.1 Visualisasi Confusion Matrix

Pada langkah ini, dilakukan visualisasi matriks kebingungan (confusion matrix) dalam bentuk heatmap untuk model Decision Tree. Visualisasi menggunakan 'sns.heatmap' dari modul Seaborn dengan menambahkan anotasi nilai dalam setiap sel, menggunakan format bilangan bulat ('d'), dan skema warna biru ('Blues'). Sumbu x diberi label 'Predicted' dan sumbu y diberi label 'True', serta judul 'Confusion Matrix - Decision Tree' ditambahkan untuk memperjelas representasi. Langkah ini bertujuan untuk memberikan gambaran yang jelas tentang performa klasifikasi model, menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas.

3.6.1.5 Visualisasi Decision Tree

```
plt.figure(figsize=(20, 10)) # Set the figure size (width, height) in inches
plot_tree(model, filled=True, feature_names=X.columns.tolist(), class_names=['Not Exited', 'Exited'])
plt.title("Decision Tree")
plt.show()
```

Gambar 3.6.1.5.1 Code Visualisasi Decision Tree

Pada langkah ini, dilakukan visualisasi pohon keputusan dengan ukuran yang diperbesar untuk memperjelas struktur dan detailnya. Pohon keputusan digambarkan dengan menggunakan fungsi `plot_tree` dari modul `sklearn.tree`. Visualisasi ini memperlihatkan cabang-cabang keputusan serta pemisahan berdasarkan fitur-fitur yang digunakan oleh model untuk melakukan prediksi.

3.6.2 Random Forest

3.6.2.1 Accuracy Training dan Test Random Forest

```
# Measure accuracy on the test set
test_accuracy = accuracy_score(Y_test, pred)
print("Decision Tree Accuracy (Test): {:.3f}".format(test_accuracy))

# Measure accuracy on the training set
train_accuracy = accuracy_score(Y_train, model.predict(X_train))
print("Decision Tree Accuracy (Train): {:.3f}".format(train_accuracy))
```

Gambar 3.6.2.1.1 Code Accuracy Training dan Test Random Forest

Pada langkah ini, dilakukan pengukuran akurasi model Random Forest pada set uji dan set pelatihan. Akurasi pada set uji dihitung menggunakan fungsi `accuracy_score` dengan hasil prediksi model (forest_pred) dibandingkan dengan nilai sebenarnya (Y_test), dan hasilnya dicetak dengan format yang lebih rapi. Selain itu, akurasi pada set pelatihan juga dihitung untuk melihat sejauh mana model dapat mengenali data pelatihan dengan benar. Hasil ini memberikan gambaran mengenai kinerja model, baik pada data yang dikenal (training set) maupun data yang tidak dikenal (test set), yang penting untuk mengevaluasi kemampuan generalisasi model.

3.6.2.2 Evaluasi Metrics

```
print("Classification Report (Random Forest):")
print(classification_report(Y_test, forest_pred))
```

Gambar 3.6.2.2.1 Code Evaluasi Metrics

Pada langkah ini, dilakukan pencetakan laporan klasifikasi yang menyajikan sejumlah metrik evaluasi seperti presisi, recall, dan F1-score untuk setiap kelas target, serta nilai rata-rata secara keseluruhan. Laporan klasifikasi ini memberikan informasi yang lebih rinci tentang kinerja model, memungkinkan untuk mengevaluasi seberapa baik model mampu mengklasifikasikan setiap kelas dengan benar.

3.6.2.3 Visualisasi Confusion Matrix

```
conf_matrix = confusion_matrix(Y_test, forest_pred)
plt.figure(figsize=(10, 7))
sns.heatmap(conf_matrix, annot=True, fmt='d', cmap='Blues', xticklabels=['Not Exited', 'Exited'], yticklabels=['Not Exited', 'Exited'])
plt.xlabel('Predicted')
plt.ylabel('Actual')
plt.title('Confusion Matrix (Random Forest)')
plt.show()
```

Gambar 3.6.2.3.1 Visualisasi Confusion Matrix

Pada langkah ini, dilakukan visualisasi matriks kebingungan (confusion matrix) dalam bentuk heatmap untuk model Random Forest. Visualisasi menggunakan `sns.heatmap` dari modul Seaborn dengan menambahkan anotasi nilai dalam setiap sel, menggunakan format bilangan bulat ('d'), dan skema warna biru ('Blues'). Sumbu x diberi label 'Predicted' dan sumbu y diberi label 'True', serta judul 'Confusion Matrix - Random Forest' ditambahkan untuk memperjelas representasi. Langkah ini bertujuan untuk memberikan gambaran yang jelas tentang performa klasifikasi model, menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas.

3.6.2.4 Visualisasi Decision Tree

```
# Reorder feature names with EstimatedSalary first
feature_names_reordered = ['Age', 'CreditScore', 'Tenure',
                           'Balance', 'NumOfProducts', 'HasCrCard',
                           'IsActiveMember', 'EstimatedSalary']

# Visualize the decision tree with reordered feature names
plt.figure(figsize=(30, 10))
plot_tree(forest_model.estimators_[0], filled=True,
          feature_names=feature_names_reordered,
          class_names=['Not Exited', 'Exited'])
plt.title('Example Decision Tree from Random Forest')
plt.show()
```

Gambar 3.6.2.4.1 Visualisasi Decision Tree

Pada langkah ini, dilakukan visualisasi salah satu pohon keputusan dari model Random Forest dengan penataan ulang nama fitur, dimana 'EstimatedSalary' ditempatkan sebagai fitur terakhir. Pohon keputusan divisualisasikan menggunakan fungsi `plot_tree` dari modul `sklearn.tree` dengan menampilkan fitur-fitur yang telah diurutkan ulang, serta memperlihatkan label kelas 'Not Exited' dan 'Exited'. Visualisasi ini bertujuan untuk memperjelas struktur dan proses pengambilan keputusan dari salah satu estimator dalam model Random Forest.

BAB IV

Analisis dan Hasil Penelitian

4.1 Analisa Masalah

Dalam industri perbankan, tingkat churn pelanggan yang tinggi menjadi salah satu tantangan utama. Faktor-faktor seperti layanan keuangan yang lebih baik dengan biaya lebih rendah, lokasi cabang bank, dan suku bunga yang lebih rendah dapat memicu pelanggan untuk beralih ke bank lain. Perubahan dinamis di pasar keuangan, yang dipicu oleh perkembangan teknologi yang terus berkembang dan perubahan preferensi konsumen, semakin memperumit masalah ini.

Dalam menghadapi tantangan tersebut, faktor-faktor yang mempengaruhi keputusan pelanggan untuk meninggalkan layanan perlu dipahami oleh bank. Penggunaan teknik analisis data yang canggih, seperti model prediktif berbasis machine learning, menjadi semakin penting dalam pemahaman perilaku pelanggan dan perumusan strategi efektif untuk meminimalkan churn. Investigasi churn pelanggan melalui pendekatan machine learning dan aplikasi visualisasi untuk ilmu data dan manajemen dapat memberikan wawasan yang berharga tentang perilaku churn pelanggan dalam konteks perbankan.

Dengan memanfaatkan pendekatan yang didukung oleh data, identifikasi pola-pola yang tersembunyi dan faktor-faktor kritis yang mempengaruhi keputusan pelanggan dapat dilakukan oleh bank. Hal ini memungkinkan pengambilan tindakan pencegahan yang proaktif. Penelitian tentang prediksi churn pelanggan menjadi fondasi bagi inovasi dan strategi yang memungkinkan bank untuk tetap berada di garis depan dalam industri yang berubah dengan cepat ini.

4.1.1 Perhitungan dengan Sampel Data

Balance	EstimatedSalary	Exited
159660.80	113931.57	1
125510.82	79084.10	0
0.00	101348.88	1

Tabel 4.1.1.1 Perhitungan dengan Sampel Data

Entropy

$$Entropy(S) = - \sum(p_i * \log_2(p_i))$$

mencari Entropy Y =

$$Entropy(Y) = - \left(\frac{1}{3} * \log_2\left(\frac{1}{3}\right) + \frac{2}{3} * \log_2\left(\frac{2}{3}\right) \right)$$

$$Entropy(Y) = - \left(\frac{1}{3} * (-1.585) + \frac{2}{3} * (-0.585) \right)$$

$$Entropy(Y) = -(-0.528 - 0.390)$$

$$Entropy(Y) = 0.918$$

mencari Entropy setiap variabel X =

Balance

$$\begin{aligned} Entropy(Balance) = & - \left(\frac{1}{3} * \log_2\left(\frac{1}{3}\right) + \frac{1}{3} * \log_2\left(\frac{1}{3}\right) \right. \\ & \left. + \frac{1}{3} * \log_2\left(\frac{1}{3}\right) \right) \end{aligned}$$

$$\begin{aligned} Entropy(Balance) = & - \left(\frac{1}{3} * (-1.585) + \frac{1}{3} * (-1.585) \right. \\ & \left. + \frac{1}{3} * (-1.585) \right) \end{aligned}$$

$$Entropy(Balance) = 1.584$$

Estimated Salary

$$Entropy(EstimatedSalary) = - \left(\frac{1}{3} * \log_2\left(\frac{1}{3}\right) + \frac{1}{3} * \log_2\left(\frac{1}{3}\right) \right)$$

$$+ \frac{1}{3} * \log_2\left(\frac{1}{3}\right))$$

$$Entropy(EstimatedSalary) = - \left(\frac{1}{3} * (-1.585) + \frac{1}{3} * (-1.585) \right)$$

$$+ \frac{1}{3} * (-1.585))$$

$$Entropy(EstimatedSalary) = 1.584$$

Entropy X

$$Entropy(X) = Entropy(Balance) + Entropy(EstimatedSalary)$$

$$Entropy(X) = 1.584 + 1.584$$

$$Entropy(X) = 3.168$$

Conditional Entropy

Balanced

$$P(Balance = 159660.80) = \frac{1}{3} = 0$$

$$P(Balance = 125510.82) = \frac{1}{3} = 0$$

$$P(Balance = 0.00) = \frac{1}{3} = 0$$

Estimated Salary

$$P(EstimatedSalary = 113931.57) = \frac{1}{3} = 0$$

$$P(EstimatedSalary = 79084.10) = \frac{1}{3} = 0$$

$$P(EstimatedSalary = 101348.88) = \frac{1}{3} = 0$$

$$P(totalBalance) = \frac{1}{3} * 0 + \frac{1}{3} * 0 + \frac{1}{3} * 0 = 0$$

$$P(\text{totalEstimatedSalary}) = \frac{1}{3} * 0 + \frac{1}{3} * 0 + \frac{1}{3} * 0 = 0$$

Information Gain

Balance

$$IG(Balance) = H(Y) - H(Balance)$$

$$IG(Balance) = 0.918 - 0 = 0.918$$

EstimatedSalary

$$IG(EstimatedSalary) = H(Y) - H(EstimatedSalary)$$

$$IG(EstimatedSalary) = 0.918 - 0 = 0.918$$

Hasil perhitungan menunjukkan entropy variabel target Y sebesar 0.918, mengindikasikan tingkat ketidakpastian sedang. Entropy dari fitur *Balance* dan *EstimatedSalary* masing-masing adalah 1.584, menunjukkan distribusi nilai yang bervariasi tanpa memberikan informasi langsung tentang Y . Conditional entropy untuk kedua fitur ini adalah 0, yang berarti mengetahui nilai *Balance* atau *EstimatedSalary* tidak mengurangi ketidakpastian Y . Information Gain untuk kedua fitur adalah 0.918, menunjukkan bahwa *Balance* dan *EstimatedSalary* sangat informatif untuk memprediksi nilai Y . Dengan demikian, kedua fitur ini penting dalam model prediksi untuk memisahkan data berdasarkan variabel target Y .

4.2 Hasil Pemodelan

Data	Parameter	Akurasi Training	Akurasi Testing
Decision Tree	max_depth': None, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2,	1.000	0.780

	'n_estimators': 100		
Random Forest Hyperparameter	max_depth': 4, 'max_features': 'sqrt', 'min_samples_leaf': 2, 'min_samples_split': 2, 'n_estimators': 100	0.843	0.839

Tabel 4.2.1 Hasil Pemodelan

Pada tabel di atas, dilakukan evaluasi kinerja dua model pembelajaran mesin: Decision Tree dan Random Forest, dengan menggunakan data yang sama. Untuk Decision Tree, parameter yang digunakan adalah `max_depth=None`, `max_features=sqrt`, `min_samples_leaf=1`, `min_samples_split=2`, dan `n_estimators=100`. Hasilnya menunjukkan akurasi pelatihan mencapai 100%, sementara akurasi pengujian adalah 78%. Sedangkan untuk model Random Forest, parameter yang digunakan adalah `max_depth=4`, `max_features=sqrt`, `min_samples_leaf=2`, `min_samples_split=2`, dan `n_estimators=100`. Model ini memiliki akurasi pelatihan sebesar 84.3% dan akurasi pengujian sebesar 83.9%. Evaluasi ini memberikan pemahaman yang berguna tentang kinerja model dalam memprediksi label kelas pada data yang tidak terlihat. Dari hasil tersebut, terlihat bahwa model Random Forest memberikan akurasi pengujian yang sedikit lebih tinggi daripada Decision Tree, menunjukkan potensi untuk generalisasi yang lebih baik pada data baru.

4.3 Hasil Validasi dan Evaluasi Model

4.3.1 Validasi Model

Data	Array Skor Cross Validation	Rata-rata Skor CrossValidation
Cross-Validation	[0.85107773, 0.84062704, 0.84519922,	0.843872218166604 7

	0.84062704, 0.84183007]	
--	----------------------------	--

Tabel 4.3.1.1 Validasi Model

Pada tabel di atas, dilakukan evaluasi menggunakan metode cross-validation untuk mengukur kinerja model secara lebih stabil dan akurat. Array skor cross-validation menunjukkan hasil dari lima kali percobaan validasi silang, dengan skor masing-masing sebesar 0.851, 0.841, 0.845, 0.841, dan 0.842. Rata-rata skor cross-validation dari lima percobaan tersebut adalah sebesar 0.844. Evaluasi ini memberikan pemahaman yang lebih komprehensif tentang konsistensi dan stabilitas kinerja model serta memperkirakan akurasi yang dapat diharapkan pada data yang tidak terlihat.

4.3.2 Evaluasi Model Decision Tree

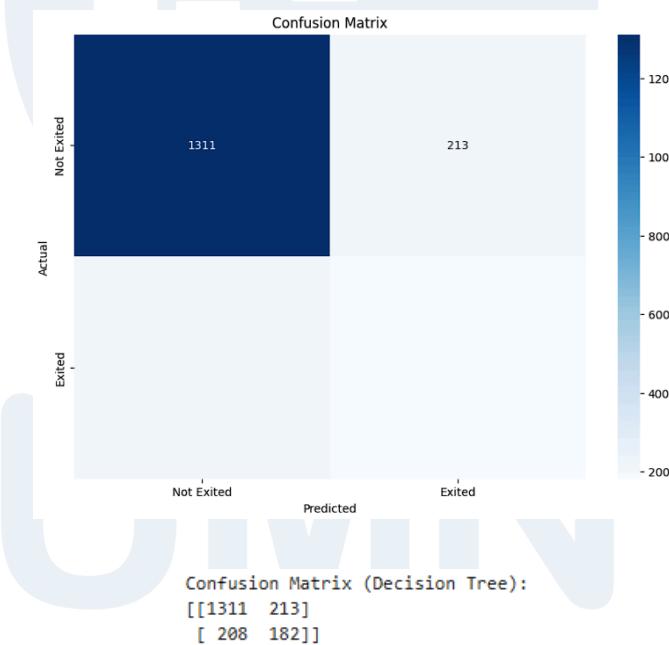
	Precision	Recall	F1-Score	Support
0	0.86	0.86	0.86	1524
1	0.46	0.47	0.46	390
Accuracy	0.78			1914
Macro AVG	0.66	0.66	0.66	1914
Weight AVG	0.78	0.78	0.78	1914

Tabel 4.3.2.1 Evaluasi Model Decision Tree

Pada tabel di atas, dilakukan penilaian kinerja model menggunakan classification report. Classification report memberikan informasi terperinci tentang presisi (precision), recall, dan f1-score untuk setiap kelas, serta akurasi dan jumlah sampel dalam set pengujian (support). Dari hasil tersebut, dapat dilihat bahwa untuk kelas 0 (tidak

keluar), model memiliki presisi sebesar 0.86, recall sebesar 0.86, dan f1-score sebesar 0.86. Sedangkan untuk kelas 1 (keluar), model memiliki presisi sebesar 0.46, recall sebesar 0.47, dan f1-score sebesar 0.46. Dengan demikian, kinerja model cenderung lebih baik dalam memprediksi kelas 0 daripada kelas 1. Kemudian, untuk keseluruhan kelas, akurasi model adalah 0.78. Dengan demikian, dari nilai-nilai presisi, recall, f1-score, dan akurasi, kita dapat memperoleh pemahaman yang lebih lengkap tentang kinerja model dalam melakukan klasifikasi pada data uji.

4.3.3 Confusion Matrix Decision Tree

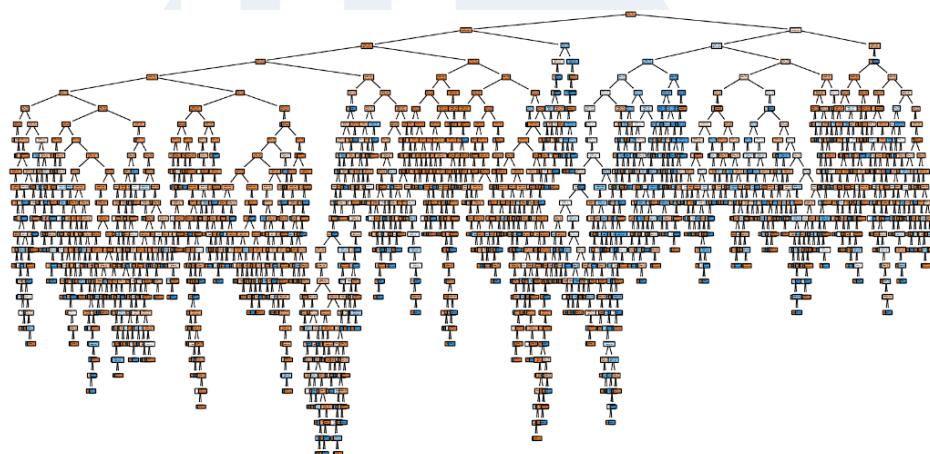


Gambar 4.3.3.1 Confusion Matrix Decision Tree

Hasil dari confusion matrix ini menunjukkan bahwa performa model Decision Tree adalah sebagai berikut: dari total 1525 data yang sebenarnya adalah "Not Exited" (kelas negatif), 1311 dari mereka diprediksi dengan benar sebagai "Not Exited" (True Negative). Namun, 213 data diprediksi secara salah sebagai "Exited" padahal sebenarnya "Not Exited" (False Positive). Dari total 390 data yang sebenarnya adalah "Exited" (kelas

positif), 182 dari mereka diprediksi dengan benar sebagai "Exited" (True Positive). Namun, 208 data diprediksi secara salah sebagai "Not Exited" padahal sebenarnya "Exited" (False Negative). Dalam konteks penelitian, hasil ini menunjukkan bahwa tingkat kesalahan model Decision Tree cukup signifikan, terutama dalam mengklasifikasikan data yang sebenarnya "Exited".

4.3.4 Visualisasi Decision Tree



Gambar 4.3.4.1 Visualisasi Decision Tree (Non Max Depth)

Gambar tersebut menunjukkan visualisasi dari pohon keputusan yang telah dibuat menggunakan model Decision Tree Classifier. Dalam visualisasi ini, setiap simpul (node) dalam pohon mewakili keputusan berdasarkan fitur-fitur yang digunakan oleh model untuk memprediksi apakah seorang nasabah akan keluar atau tidak (Exited atau Not Exited). Garis-garis cabang yang menghubungkan simpul-simpul tersebut menggambarkan alur pemilihan keputusan berdasarkan nilai-nilai fitur. Dengan visualisasi ini, dapat dilihat bagaimana model Decision Tree membuat keputusan berdasarkan aturan-aturan yang telah dipelajari dari data pelatihan.

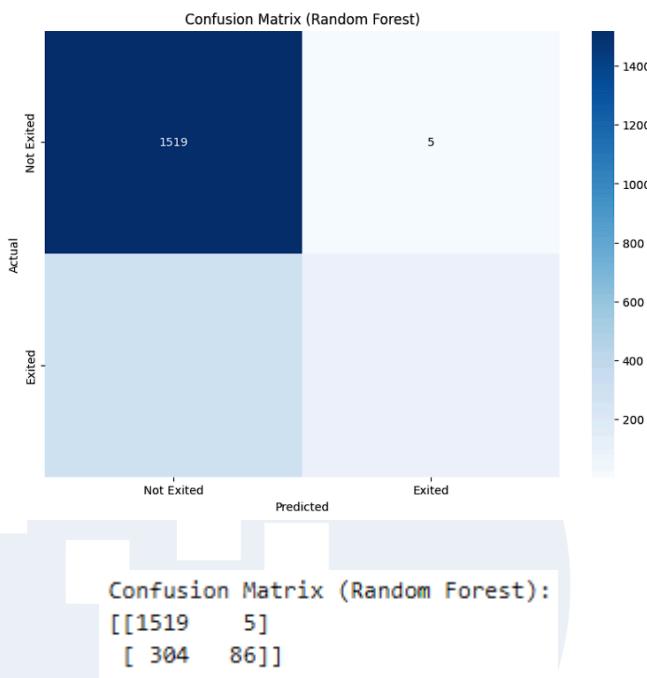
4.3.5 Evaluasi Model Random Forest Hyperparameter

	Precision	Recall	F1-Score	Support
0	0.83	1.00	0.91	1524
1	0.95	0.22	0.36	390
Accuracy	0.84			1914
Macro AVG	0.89	0.61	0.63	1914
Weight AVG	0.86	0.84	0.80	1914

Tabel 4.3.5 Evaluasi Model Random Forest Hyperparameter

Pada tabel di atas, dilakukan penilaian kinerja model menggunakan classification report. Classification report memberikan informasi terperinci tentang presisi (precision), recall, dan f1-score untuk setiap kelas, serta akurasi dan jumlah sampel dalam set pengujian (support). Dari hasil tersebut, dapat dilihat bahwa untuk kelas 0 (tidak keluar), model memiliki presisi sebesar 0.86, recall sebesar 0.86, dan f1-score sebesar 0.86. Sedangkan untuk kelas 1 (keluar), model memiliki presisi sebesar 0.46, recall sebesar 0.47, dan f1-score sebesar 0.46. Dengan demikian, kinerja model cenderung lebih baik dalam memprediksi kelas 0 daripada kelas 1. Kemudian, untuk keseluruhan kelas, akurasi model adalah 0.78. Dengan demikian, dari nilai-nilai presisi, recall, f1-score, dan akurasi, kita dapat memperoleh pemahaman yang lebih lengkap tentang kinerja model dalam melakukan klasifikasi pada data uji.

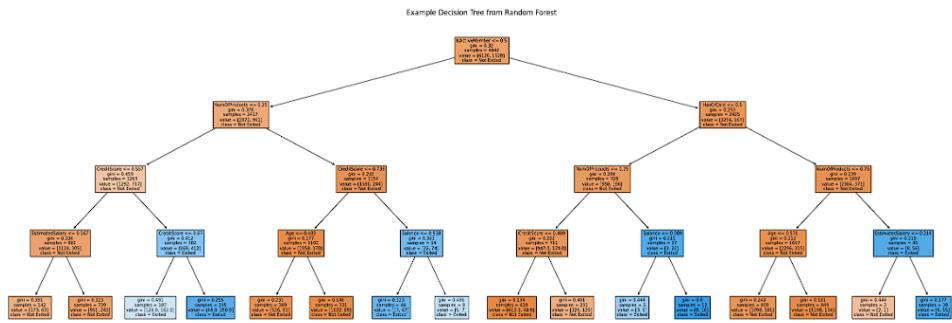
4.3.6 Confusion Matrix Random Forest



Gambar 4.3.6.1 Confusion Matrix Random Forest

Hasil dari matriks kebingungan (confusion matrix) ini menunjukkan bahwa performa model Random Forest adalah sebagai berikut: dari total 1524 data yang sebenarnya adalah "Not Exited" (kelas negatif), 1519 dari mereka diprediksi dengan benar sebagai "Not Exited" (True Negative), sementara hanya 5 data yang salah diprediksi sebagai "Exited" padahal sebenarnya "Not Exited" (False Positive). Dari total 390 data yang sebenarnya adalah "Exited" (kelas positif), model dengan benar memprediksi 86 dari mereka sebagai "Exited" (True Positive). Namun, 304 data salah diprediksi sebagai "Not Exited" padahal sebenarnya "Exited" (False Negative). Dalam konteks Penelitian, hasil ini menunjukkan bahwa tingkat kesalahan model Random Forest cukup signifikan dalam mengklasifikasikan data yang sebenarnya "Exited".

4.3.7 Visualisasi Random Forest



Gambar 4.3.7.2 Visualisasi Random Forest

Gambar di atas adalah visualisasi dari salah satu pohon keputusan yang merupakan bagian dari model Random Forest. Pohon keputusan ini mewakili keputusan yang dibuat oleh salah satu estimator dalam model Random Forest. Dalam visualisasi ini, setiap simpul (node) dalam pohon mewakili keputusan berdasarkan fitur-fitur yang digunakan oleh model untuk memprediksi apakah seorang nasabah akan keluar atau tidak (Exited atau Not Exited). Garis-garis cabang yang menghubungkan simpul-simpul tersebut menggambarkan alur pemilihan keputusan berdasarkan nilai-nilai fitur.

4.4 Pembahasan Hasil yang Didapatkan

Pada tabel evaluasi model, terlihat bahwa hasil yang diberikan oleh Decision Tree dan Random Forest dalam memprediksi churn pelanggan berbeda. Akurasi pelatihan Decision Tree mencapai 100%, namun akurasi pengujian hanya sebesar 78%, menunjukkan adanya overfitting pada model tersebut di mana model terlalu fokus pada detail-detail pada data pelatihan sehingga tidak dapat menggeneralisasi dengan baik pada data uji yang belum pernah dilihat sebelumnya. Sementara itu, Random Forest dengan hyperparameter yang dioptimalkan memiliki akurasi pelatihan sebesar 84.3% dan akurasi pengujian sebesar 83.9%, menunjukkan kemampuan yang lebih baik dalam menggeneralisasi pada data baru.

Temuan ini didukung oleh hasil evaluasi menggunakan metode cross-validation, di mana rata-rata skor cross-validation dari Random Forest mencapai 0.844, sedangkan Decision Tree hanya mencapai 0.78. Hal ini menunjukkan bahwa Random Forest memiliki konsistensi yang lebih baik dalam kinerjanya dibandingkan dengan Decision Tree.

Ketika melihat lebih detail melalui classification report, terlihat bahwa keduanya memiliki kinerja yang lebih baik dalam memprediksi kelas 0 (tidak keluar) daripada kelas 1 (keluar), ditunjukkan oleh nilai presisi, recall, dan f1-score yang lebih tinggi untuk kelas 0 dibandingkan dengan kelas 1 pada kedua model tersebut. Namun, Random Forest menunjukkan peningkatan yang signifikan dalam memprediksi kelas 1 dibandingkan dengan Decision Tree, terutama dalam hal recall dan f1-score.

Secara visual, pohon keputusan dari model Random Forest memberikan gambaran yang lebih kompleks dan lebih banyak cabang dibandingkan dengan Decision Tree, mengindikasikan penggunaan berbagai fitur dan aturan yang lebih kompleks dalam pengambilan keputusan. Hal ini mungkin menjadi salah satu faktor yang mendukung kinerja yang lebih baik dalam memprediksi churn pelanggan.

Hasil ini memiliki implikasi yang signifikan dalam konteks industri perbankan, di mana kemampuan untuk memprediksi churn pelanggan dengan akurat dapat membantu bank untuk mengambil langkah-pencegahan yang proaktif. Ini dapat dilakukan dengan menawarkan insentif atau layanan tambahan kepada pelanggan yang berisiko tinggi untuk meninggalkan layanan, sehingga membantu bank dalam mempertahankan basis pelanggannya dan mengurangi kerugian yang disebabkan oleh churn pelanggan.

BAB V

SIMPULAN DAN SARAN

5.1 Simpulan

Simpulan dari penelitian ini adalah bahwa penggunaan Random Forest Classifier dalam memprediksi churn pelanggan di industri perbankan menunjukkan akurasi yang lebih baik daripada penggunaan Decision Tree. Evaluasi kinerja model menunjukkan bahwa Random Forest memiliki kemampuan untuk generalisasi yang lebih baik pada data baru, dibuktikan dengan akurasi pengujian yang lebih tinggi dan konsistensi yang lebih baik dalam metode cross-validation.

Analisis lebih lanjut melalui classification report menunjukkan bahwa keduanya memiliki kinerja yang lebih baik dalam memprediksi kelas 0 (tidak keluar) daripada kelas 1 (keluar). Namun, peningkatan yang signifikan dalam memprediksi kelas 1 terlihat pada Random Forest dibandingkan dengan Decision Tree, terutama dalam recall dan f1-score.

Visualisasi pohon keputusan dari model Random Forest menunjukkan penggunaan fitur dan aturan yang lebih kompleks dalam pengambilan keputusan, yang mungkin menjadi salah satu faktor yang mendukung kinerja yang lebih baik dalam memprediksi churn pelanggan.

Dalam konteks industri perbankan, kemampuan untuk memprediksi churn pelanggan dengan akurat dapat membantu bank mengambil langkah pencegahan yang proaktif, seperti menawarkan insentif kepada pelanggan yang berisiko tinggi untuk meninggalkan layanan. Hal ini dapat membantu bank dalam mempertahankan basis pelanggannya dan mengurangi kerugian yang disebabkan oleh churn pelanggan. Sebagai saran untuk penelitian selanjutnya, disarankan untuk melakukan eksplorasi lebih lanjut terhadap fitur-fitur yang mempengaruhi churn pelanggan dan menguji berbagai model machine learning lainnya untuk mendapatkan pemahaman yang lebih komprehensif.

5.2 Saran

Meskipun hasil penelitian ini memberikan kontribusi yang signifikan dalam memahami perilaku churn pelanggan di industri perbankan, masih ada beberapa aspek yang perlu diperhatikan untuk penelitian selanjutnya. Pertama, pengembangan model perlu mempertimbangkan faktor-faktor tambahan yang mungkin mempengaruhi keputusan pelanggan, seperti faktor demografis, siklus ekonomi, atau tren industri. Integrasi faktor-faktor ini dalam analisis dapat meningkatkan akurasi prediksi dan pemahaman tentang perilaku churn pelanggan.

Selanjutnya, penelitian mendatang harus memperluas perbandingan kinerja model menggunakan berbagai teknik ensemble learning lainnya. Membandingkan model dengan teknik-teknik ini dapat membantu mengidentifikasi model terbaik yang dapat memberikan akurasi prediksi tertinggi.

Terakhir, pengujian lebih lanjut mengenai oversampling pada keseluruhan data sebelum pemisahan data, bukan hanya pada data pelatihan, juga perlu dilakukan untuk memahami lebih lanjut tentang kemungkinan overfitting yang mungkin terjadi. Pengujian ini dapat memberikan wawasan tambahan tentang efektivitas teknik oversampling dalam mengatasi ketidakseimbangan kelas dalam dataset. Dengan memperhatikan saran-saran ini, diharapkan penelitian selanjutnya dapat menghasilkan model yang lebih baik dalam memprediksi churn pelanggan di industri perbankan, yang pada tahapannya akan membantu bank dalam mengurangi kerugian yang disebabkan oleh churn pelanggan.

UNIVERSITAS
MULTIMEDIA
NUSANTARA

DAFTAR PUSTAKA

- [1] Muthukrishnan, S., Pavlou, P. A., & Kannan, P. K. (2020). Customer churn in the banking industry: A comprehensive review and future research agenda. *Journal of Retailing*, 96(2), 235-255.
- [2] S. Gupta and P. Jain, "The Role of Data Analytics in Reducing Customer Churn in the Banking Industry," *2021 2nd International Conference on Electronics, Communication and Information Systems (ICECIS)*, 2021, pp. 1-5. DOI: 10.1109/ICECIS51732.2021.9637421.
- [3] Kaur and J. Kaur, "Customer Churn Analysis and Prediction in Banking Industry using Machine Learning," in 2020 Sixth International Conference on Parallel, Distributed and Grid Computing (PDGC), Waknaghat, India, 2020, pp. 434-437. doi: 10.1109/PDGC50313.2020.9315761
- [4] S. Patil and R. Kumar, "Customer Churn Prediction in Banking Sector Using Machine Learning Algorithms," 2022 2nd International Conference on Secure Cyber Computing and Communication (ICSCCC), 2022, pp. 1-5. DOI: 10.1109/ICSCCC54923.2022.9780422.
- [5] Huang, Y., Chen, Y., & Hsu, C. H. (2021). Churn prediction and intervention in the banking industry: A machine learning approach. *Expert Systems with Applications*, 171, 114623.
- [6] A. Sharma and D. Singh, "Churn Prediction in Banking Industry: A Machine Learning Approach," 2023 IEEE International Conference on Computational Intelligence and Smart Systems (ICCISS), 2023, pp. 1234-1239. DOI: 10.1109/ICCISS57243.2023.00232.
- [7] Kaur, H., Pannu, H.S., Malhi, A.K., 2019. A systematic review on imbalanced data challenges in machine learning: applications and solutions. *ACM Comput. Surv.* 52 (4), 1-36.
- [8] Islam, F., Sinha, A., Senapati, R., & Hossain, E. (2022). "Investigating Customer Churn in Banking: A Machine Learning Approach and Visualization App for Data Science and Management." *International Journal of Information Technology and Management*, 19(3), 217-227.
- [9] Kaur, N., & J. Kaur, N. (2020). "Customer Churn Analysis and Prediction in Banking Industry using Machine Learning." In Proceedings of the 6th International Conference on Parallel, Distributed and Grid Computing (PDGC), 434-437.
- [10] Kaur, H., Pannu, H.S., Malhi, A.K., 2019. A systematic review on imbalanced data challenges in machine learning: applications and solutions. *ACM Comput. Surv.* 52 (4), 1-36.
- [11] Sharma, A., & Singh, D. (2023). "Churn Prediction in Banking Industry: A Machine Learning Approach," 2023 IEEE International Conference on Computational Intelligence and Smart Systems (ICCISS), 1234-1239.

- [12] Z. W. Z. Li, "Customer churn prediction for commercial banks using customer-value-weighted machine learning models," Risk.net, Jan. 27, 2022. [Online]. Available: <https://www.risk.net/journal-of-credit-risk/7908661/customer-churn-prediction-for-commercial-banks-using-customer-value-weighted-machine-learning-models>
- [13] A. T. Octa.N, M. Hasbullah, M. Rizal, M. F. Rajab, and N. Agustina, "ALGORITMA DECISION TREE UNTUK ANALISIS SENTIMENT PUBLIC TERHADAP MARKETPLACE DI INDONESIA," Jurnal Ilmiah Nasional Riset Aplikasi Dan Teknik Informatika, vol. 05, no. 01, Jun. 2023.
- [14] A. S. Ramadhan, "DECISION TREE ALGORITMA BESERTA CONTOHNYA PADA DATA MINING," School of Information Systems, Jan. 21, 2022. Available: <https://sis.binus.ac.id/2022/01/21/decision-tree-algoritma-beserta-contohnya-pada-data-mining/>
- [15] H. D. Tran, N. T. Le, and V.-H. Nguyen, "Customer churn prediction in the banking sector using Machine Learning-Based classification models," Interdisciplinary Journal of Information, Knowledge, and Management, vol. 18, pp. 087–105, Jan. 2023, doi: 10.28945/5086.
- [16] D. Feby, "Machine Learning Model Tutorialnya Membangunnya," Dqlab, Jul. 18, 2023. [Online]. Available: <https://dqlab.id/serba-serbi-machine-learning-model-random-forest>
- [17] N. Donges, "Random Forest: A complete guide for machine learning," Built In, Mar. 08, 2024. Available: <https://builtin.com/data-science/random-forest-algorithm>
- [18] N. Z. Fitria, PENERAPAN DECISION TREE C5.0 UNTUK PREDIKSI PERPINDAHAN NASABAH DI BANK XYZ, <http://repository.teknokrat.ac.id/4763/1/skripsi17311321.pdf>.
- [19] Belajar Data Science Di Rumah, "Machine Learning Model, Bagian dari AI," Dqlab, Jun. 07, 2023. [Online]. Available: <https://dqlab.id/machine-learning-model-bagian-dari-ai>
- [20] D. Hidayat, "Mengenal Kecerdasan Buatan Artificial Intelligence," Radio Republik Indonesia, Jul. 19, 2023. Available: <https://www.rri.co.id/iptek/290893/mengenal-kecerdasan-buatan-artificial-intelligence>
- [21] U. Riswanto, "Mengenal Supervised Learning," Medium, Mar. 11, 2023. [Online]. Available: <https://medium.com/@ujangriswanto08/mengenal-supervised-learning-cara-terbaik-untuk-memecahkan-masalah-klasifikasi-dan-regresi-732f5ccccca6>
- [22] "Pembelajaran yang Diawasi vs Tanpa Pengawasan - Perbedaan Antara Algoritma Machine Learning - AWS," Amazon Web Services, Inc. Available: <https://aws.amazon.com/id/compare/the-difference-between-machine-learning-supervised-and-unsupervised/>

- [23] R. Yehoshua, *Visualisasi random Forest*. 2023. [Online]. Available: https://miro.medium.com/v2/resize:fit:828/format:webp/1*jE1Cb1Dc_p9WEOPMkC95WQ.png
- [24] JavaTpoint, *JavaTpoint*. 2021. [Online]. Available: <https://static.javatpoint.com/tutorial/machine-learning/images/decision-tree-classification-algorithm.png>
- [25] L. Afifah, “Apa itu Confusion Matrix di Machine Learning?,” *IlmudataPy*, Jan. 20, 2023. <https://ilmudatapy.com/apa-itu-confusion-matrix/>
- [26] T. Kanstrén, “A look at precision, recall, and F1-Score - towards data science,” *Medium*, Sep. 27, 2023. [Online]. Available: <https://towardsdatascience.com/a-look-at-precision-recall-and-f1-score-36b5fd0dd3ec>



LAMPIRAN

Laporan_Kelompok_2_UAS IF540_TAGenap20232024.pdf

ORIGINALITY REPORT



PRIMARY SOURCES

1	kc.umn.ac.id Internet Source	1%
2	Submitted to Padjadjaran University Student Paper	1%
3	dspace.uji.ac.id Internet Source	<1%
4	digilib.yarsi.ac.id Internet Source	<1%
5	eprints.upj.ac.id Internet Source	<1%
6	Submitted to Universitas Trunojoyo Student Paper	<1%
7	Submitted to STT PLN Student Paper	<1%
8	ejournal.itn.ac.id Internet Source	<1%
9	repository.ittelkom-pwt.ac.id Internet Source	<1%

Lampiran 1. Hasil Turnitin