PREDIKSI CALON NASABAH DEPOSITO BERBASIS CRISP-DM DAN *SMOTE XGBOOST CLASSIFICATION*

TESIS

Diajukan sebagai salah satu syarat untuk memperoleh gelar

Magister Ilmu Komputer (M.Kom)

RIKI SUPRIYADI

14002371

Program Studi Ilmu Komputer (S2)

Fakultas Teknologi Informasi

Universitas Nusa Mandiri

2021

HALAMAN PERSETUJUAN DAN PENGESAHAN TESIS

Tesis ini diajukan oleh:

Nama	:	Riki Supriyadi
NIM	:	14002371
Program Studi	:	Ilmu Komputer
Jenjang	:	Strata Dua (S2)
Konsentrasi	:	*Data Mining*
Judul Tesis	:	Prediksi Calon Nasabah Deposito Berbasis CRISP-DM Dan SMOTE XGBoost Classification

Telah dipertahankan pada periode 2021-1 dihadapan Dewan Penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Magister Ilmu Komputer (M.Kom) pada Program Studi Ilmu Komputer (S2) Fakultas Teknologi Informasi Universitas Nusa Mandiri.

Jakarta, 19 Agustus 2021

PEMBIMBING TESIS

Pembimbing I	:	Dr. Didi Rosiyadi, M.Kom	
Pembimbing II	:	Eni Heni Hermaliani, M.M, M.Kom	

D E W A N P E N G U J I

Penguji I	:	Dr. Hilman Ferdinandus Pardede, S.T, M.EICT	
Penguji II	:	Dr. Agus Subekti, M.T	
Penguji III / Pembimbing I	:	Dr. Didi Rosiyadi, M.Kom	

DAFTAR ISI

Halaman

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

BAB I
PENDAHULUAN

Bank menjadi pilihan untuk menyimpan atau menabung uang, hal ini sudah menjadi kebiasaan di kalangan masyarakat dari dulu hingga sekarang. Pertumbuhan bank terutama di Indonesia sudah sangat cepat, banyak bank baru yang bermunculan dengan membawa beberapa inovasi dalam pelayanan yang diberikan, lebih mudah dalam proses pembukaan rekening ataupun saat bertransaksi. Hal ini tentu dipengaruhi dengan perkembangan teknologi yang semakin canggih.

Dalam perbankan salah satu produk yang sering ditawarkan kepada nasabah adalah deposito. Deposito adalah simpanan yang hanya dapat ditarik pada saat jatuh tempo [1]. Berdasarkan Undang-Undang Nomor 10 Tahun 1998, terdapat pengertian deposito itu sendiri, yaitu simpanan yang hanya dapat dibayarkan pada waktu tertentu melalui kesepakatan antara penyimpan dan bank [2]. Dan deposito memiliki tingkat bunga tahunan lebih tinggi dari cek atau tingkat bunga tabungan biasa. Namun jumlah nasabah yang menabung deposito masih tergolong minim karena meskipun tingkat bunga deposito lebih tinggi dari tabungan biasa, deposito hanya dapat dibayarkan pada waktu tertentu, hal ini membuat deposito dapat digunakan sebagai alternatif bentuk investasi [3]. Untuk itu, bank harus memanfaatkan peluang ini untuk menerapkan strategi pemasaran dan kegiatan promosi yang efektif.

Bank biasanya menggunakan pemasaran langsung untuk menargetkan basis pelanggannya dan menjangkau pelanggan untuk mencapai tujuan. Pelanggan yang dipilih oleh bank biasanya dihubungi secara pribadi melalui informasi kontak, seperti telepon seluler atau email, sehingga bank dapat menerima informasi langsung dari pelanggan, apakah pelanggan telah berlangganan produk yang disediakan oleh bank atau belum [4]. Dalam pemasaran langsung, dapat menggunakan salah satu metode untuk memprediksi pelanggan yang membuka simpanan di bank menggunakan prediksi berdasarkan data pelanggan yang ada, kemudian menggunakan teknik pengenalan pola (seperti statistik dan matematika) untuk memeriksa kumpulan data besar yang disimpan dan memprosesnya untuk

menemukan hubungan, pola, dan tren yang bermakna. proses tersebut dikenal dengan *data mining*.

Data mining adalah proses menemukan informasi dan pola yang berguna dari data yang sangat besar. *Data mining* juga dikenal sebagai *knowledge Discovery, knowledge extraction, data/ pattern analysis information, harvesting* dan lain lain [5]. *Data mining* menggunakan analisis matematis untuk mendapatkan atau menemukan pola dan tren dari data. Pada umumnya pola-pola tersebut sulit ditemukan dengan eksplorasi data biasa/tradisional, hal ini disebabkan hubungan antar data yang terlalu rumit, atau bisa juga disebabkan oleh data yang begitu besar.

Data mining bertujuan untuk menemukan pola yang sebelumnya tidak diketahui. Setelah diperoleh, pola-pola tersebut dapat digunakan untuk menyelesaikan berbagai macam masalah. *Data mining* juga telah menjadi teknologi baru yang kuat dengan potensi besar untuk membantu bisnis fokus pada hal yang paling penting [6]. Informasi dalam data yang dikumpulkan tentang perilaku pelanggan dan calon pelanggan. *Data mining* memungkinkan perusahaan untuk menemukan informasi dalam jumlah data yang begitu besar melalui pengolahan yang tepat dan efektif menggunakan berbagai metode yang ada pada *data mining* sehingga *data mining* dapat dengan mudah digambarkan. sebagai pola atau model atau aturan atau temuan dari *data mining*, Selain itu, *data mining* dapat membantu menjadi tren penjualan, mengembangkan kampanye pemasaran yang lebih cerdas, dan secara akurat memprediksi loyalitas pelanggan. [5]

Minat nasabah dalam pembukaan tabungan deposito pada perbankan tidak terlalu tinggi, hal ini membuat marketing pemasaran harus bisa mengetahui kondisi atau hal yang bisa menarik nasabah untuk menabung deposito. Berdasarkan hal tersebut perlu dilakukan penelitian untuk menentukan aspek kecenderungan dan juga algoritma apa yang paling sesuai dan efektif dalam melakukan prediksi terhadap nasabah yang melakukan pembukaan tabungan deposito di perbankan.

Penelitian dilakukan dengan data publik, sumber data yang digunakan bisa diperoleh dari berbagai sumber salah satunya UCI *Repository*, berdasarkan dataset *Bank Marketing* yang diperoleh dari UCI *Repository* [7] pada penelitian ini akan dilakukan beberapa eksperimen dalam memprediksi calon nasabah yang menabung

deposito di bank untuk mengetahui kecenderungan apa yang dominan terhadap nasabah yang menabung deposito dengan beberapa algoritma *data mining*.

Adapun manfaat dari penelitian ini adalah untuk mengetahui algoritma apa yang dapat memprediksi calon nasabah deposito dengan baik berdasarkan hasil akurasi yang diperoleh dan membantu manajemen perbankan mengenai minat nasabah terhadap pembukaan tabungan deposito berdasarkan data nasabah yang diperoleh dari pemasaran langsung, serta dapat memberikan bukti secara empiris untuk teori yang berkaitan sehingga dapat dijadikan sumbangan pemikiran untuk pengembangan teori berikutnya.

BAB II
LANDASAN/KERANGKA PEMIKIRAN

Tinjauan pustaka yang digunakan pada penelitian ini merupakan teori-teori yang menjadi landasan dalam penelitian ini yang didapat melalui referensi dari buku, website dan jurnal-jurnal penelitian baik yang internasional maupun nasional. Berikut tinjauan pustaka yang mendukung teoritis dari penelitian ini adalah sebagai berikut:

Data mining adalah proses penggalian data (sebelumnya tidak diketahui, implisit, dan dianggap tidak berguna) dari sejumlah besar data menjadi informasi, pengetahuan, atau pola [8]. *Data mining* merupakan proses yang menggunakan berbagai teknik dan alat analisis data untuk menemukan hubungan dan pola yang tersembunyi. Pendekatan dasar dalam *Data mining* adalah untuk meringkas data dan untuk mengekstrak informasi berguna yang masuk akal dan sebelumnya tidak diketahui [9].

Secara umum, *data mining* merupakan proses menemukan pola yang menarik dan tersembunyi dalam kumpulan data besar yang disimpan dalam basis data, gudang data, atau area penyimpanan data lainnya. Kunci *data mining* adalah data bisnis, informasi dan solusi. Mengubah data menjadi informasi dan tujuan akhir dari *data mining* adalah menggunakan informasi untuk membuat keputusan bisnis yang lebih baik dan keputusan yang lebih baik. [10]

Data mining dapat menemukan tren dan pola tersembunyi yang tidak muncul dalam analisis kueri sederhana sehingga dapat memiliki bagian penting dalam hal menemukan pengetahuan dan membuat keputusan. Tugas-tugas semacam itu dapat bersifat prediksi seperti klasifikasi dan regresi atau deskriptif seperti *clustering* dan asosiasi [11]

Extreme Gradient Boosting (*XGBoost)* diusulkan oleh Chen Tianqi pada tahun 2016, menghadirkan kompleksitas komputasi yang rendah, kecepatan *running* yang cepat dan akurasi yang tinggi [13]. Mirip dengan *gradient boosting*, *XGBoost* menggabungkan pengklasifikasi dasar yang lemah menjadi pengklasifikasi yang lebih kuat. Pada setiap iterasi dari proses pelatihan, sisa dari pengklasifikasi dasar digunakan di berikutnya classifier untuk mengoptimalkan fungsi tujuan [14].

XGBoost menerapkan ekspansi *second-order Taylor* ke *loss function* untuk menggantikan turunan pertama tidak seperti *Gradient boosting* konvensional, seperti persamaan berikut: [18].

$$L = \sum_i l(y, O(x_i) + \sum_k \Omega(G_k) \quad \text{.....(2.1)}$$

Di mana, l adalah *loss function* dari *training* dan L mendefinisikan *loss function* asli untuk algoritma *XGBoost.* Sisanya dari notasi konstan sama dengan metode boosting. G didefinisikan sebagai *weak estimator* untuk pohon keputusan. Selain itu, kompleksitas pohon keputusan, Ω (G_m) adalah diagregasikan dengan suku pertama untuk membentuk fungsi tujuan. Definisi istilah reguler, Ω (G_m), dihitung sebagai:

$$\Omega(G) = \omega T + \frac{1}{2} a \sum_{j=1}^{T} s_j^2 \quad \text{.....(2.2)}$$

di mana, T menunjukkan jumlah *leaf* pohon keputusan. Sementara, ω_j^2 menunjukkan norma skor L2 untuk masing-masing leaf. γ adalah ambang batas kontrol untuk membagi *node*, dan λ adalah koefisien untuk mengurangi masalah *overfitting*.

Persamaan akhir dapat dibentuk sebagai:

$$L^m = \sum_{i=1}^{N} l(y_i, O_i^{m-1} + G_m(x_i)) + \Omega(G_m) \quad \text{.....(2.3)}$$

$$\approx \sum_{i=1}^{N} [l(y_i, O_i^{m-1}) + g_i\, G_m(x_i) + \frac{1}{2} o_i G_m(x_i)] + \Omega(G_m) \quad \text{.....(2.4)}$$

Akhirnya, dalam persamaan sebelumnya, dua variabel mendefinisikan turunan pertama dan turunan kedua dari *loss function* adalah: *gi* = *∂Om*-1*l*(*yi, Om i* -1) and *oi* = *∂*^{2}*Om*-1*l*(*yi*, O_1^{m-1}), masing-masing.

Decision Tree merupakan teknik yang paling banyak digunakan untuk pembelajaran mesin, pengenalan pola, dan tugas klasifikasi [19]. *Decision Tree* adalah pohon keputusan dimana setiap node menunjukkan *fiture* (atribut), setiap tautan (cabang) menunjukkan keputusan (*rule*) dan setiap daun menunjukkan hasil (kategoris atau nilai). Karena pohon keputusan meniru pemikiran tingkat manusia jadi sangat mudah untuk mengambil data dan membuat interpretasi yang bagus [20].

Proses dalam *Decision Tree* yaitu mengubah bentuk data (tabel) menjadi model pohon (*tree*) kemudian mengubah model pohon tersebut menjadi aturan (*rule*) [21].

Algoritma yang digunakan adalah algoritma ID3. Algoritma ID3 menggunakan konsep dari *Entropy* dan *Information Gain* [21].
Untuk nilai *Entropy* dapat ditemukan dengan menggunakan rumus:

$$Entropy\ (\mathrm{S}) = \sum_{i=1}^{k} -p_j\, log_2 p_j \quad \text{.....(2.5)}$$

Keterangan:

S : ruang (data) sampel yang digunakan untuk *training*

K : banyaknya partisi pada S

pj : probabilitas yang didapat dari *Sum* (Ya) dibagi dengan total sampel

$$\mathrm{Gain}\ (\mathrm{A}) = Entropy\ (S) - \sum_{i=1}^{k} \frac{|Si|}{|S|} Entropy\ (S_{i)} \quad \text{.....(2.6)}$$

Keterangan:

S : ruang (data) sampel yang digunakan untuk *training*

A : atribut

$|Si|$: jumlah sampel untuk nilai V

$|S|$: jumlah seluruh sampel data

$Entropy\ (S_{i)}$: *entropy* untuk sampel – sampel yang memiliki nilai i

2.1.1. *K-Nearest Neighbor* (KNN)

K-Nearest Neighbor (KNN) merupakan salah satu algoritma yang digunakan dalam masalah klasifikasi. cara kerja KNN adalah mencari jarak terdekat antara data yang akan diestimasi dengan tetangga terdekat pada data latih [22].

Algoritma *K-Nearest Neighbor* (KNN) adalah salah satu algoritma paling sederhana yang digunakan untuk menyelesaikan masalah klasifikasi dan seringkali memberikan hasil yang kompetitif dan penting [24].

Sebagian besar peneliti telah menerapkan ukuran *Euclidean* untuk menghitung jarak minimum seperti yang ditunjukkan di bawah ini:

$$Eucd\ (X,Y)\ =\ \sqrt{(x_1\ -\ y_1)^2\ +\ (x_2\ -\ y_2)^2 + \cdots + (x_n\ -\ y_n)^2} \quad \text{..... (2.7)}$$

Keterangan:

X : Sampel data *training*

Y : Sampel data *testing*

n : *input parameter*

Logistic Regression adalah salah satu teknik statistik yang paling menguntungkan untuk pembentukan model probabilitas yang baik untuk mengklasifikasikan variabel dependen dikotomis dengan kategoris atau campuran dengan faktor pembeda non-kategoris. Model *Logit* mencoba membentuk model regresi dengan metode *maximum-likelihood* untuk memilih keanggotaan kelas yang terbaik [25].

Secara matematis, fungsi *Logistic Regression* didefinisikan sebagai:

$$O\ =\ log = \frac{prob}{1 - prob} = \ c_0\ +\ c_1 a_2\ +\ c_2 a_2\ + \cdots + c_n a_n \quad \text{.....(2.8)}$$

Keterangan:

O : nilai *respons* yang akan diprediksi,

Prob : mencerminkan probabilitas kehadiran sifat yang diinginkan

1-prob : kemungkinan tidak adanya sifat minat

c1cn : nilai koefisien yang akan ditentukan selama pelatihan

a1..an): *parameter input* dalam data

Synthetic Minority Over-sampling Technique (*SMOTE*) merupakan salah satu teknik dalam mengatasi data tidak seimbang atau *imbalance data* didasarkan pada pengambilan sampel data dari kelas minoritas dengan menghasilkan titik data pada segmen utama. Pendekatan ini sangat sederhana, dan sangat berhasil dalam praktek [28].

Misalkan diberikan data dengan jumlah variabel p maka jarak antara $x^T = [x_1, x_2, \ldots, x_p]$ dan $z^T = [z_1, z_2, \ldots, z_p]$ adalah $d(x, y) = \sqrt{(x_1 - z_1)^2 + (x_2 - z_2)^2 + \cdots + (x_p - z_p)^2}$. Untuk membangkitkan data dengan metode *SMOTE* maka digunakan persamaan sebagai berikut:

$$x_{syn} = x_i + (x_{knn} - x) \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots (2.9)$$

x_{syn} merupakan pengamatan baru hasil pembangkitan, x_{i} adalah pengamatan ke-i, x_{knn} merupakan x terdekat dari x_{i}, serta gamma adalah bilangan acak dari 0 hingga 1. Untuk data nominal, mengisi sebagian besar nilai tetangga terdekat. Untuk perhitungan jarak di *SMOTE*, jika ada variabel kategori maka akan diganti dengan kuadrat median standar deviasi variabel kontinyu kelas minoritas jika nilai kategorik pada pengamatan ke-i dan j berbeda.

Python merupakan bahasa pemrograman dengan lintas *platform*, gratis dan *open source*, serta dapat digunakan untuk mengembangkan aplikasi. *Python* memiliki standar *library* yang sangat luas (*Python Standard Library*) yang dapat digunakan untuk menyelesaikan berbagai masalah pemrograman di dunia nyata [30]. *Python* dikembangkan pada tahun 1991 oleh programmer kelahiran Belanda Guido van Rossum di CWI Amsterdam sebagai pepengembangan dari bahasa pemrograman ABC. Karena kecintaan Guido pada acara TV *Monty Python's Flying Circus*, Guido memilih *Python* sebagai nama bahasa yang ia ciptakan [31].

Deposito adalah simpanan berjangka yang hanya dapat ditarik hanya dapat dilakukan pada saat jatuh tempo [1]. Dalam Undang-Undang Nomor 10 Tahun 1998, terdapat pengertian deposito itu sendiri, yaitu simpanan yang hanya dapat dibayarkan pada waktu tertentu melalui kesepakatan antara penyimpan dan bank [2].

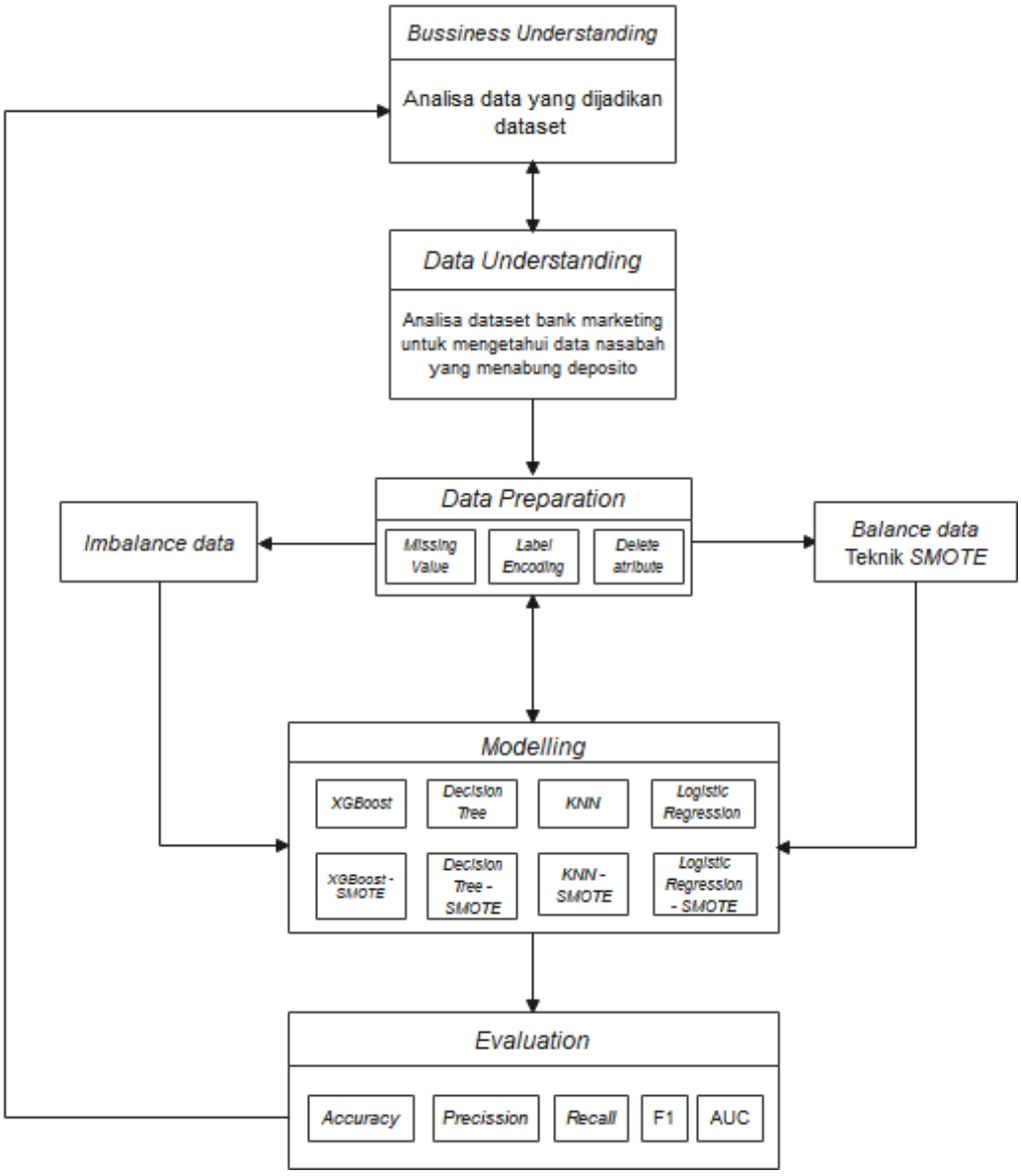
Deposito memiliki tingkat bunga tahunan lebih tinggi dari cek atau tingkat bunga tabungan biasa. Namun jumlah nasabah yang menabung deposito masih tergolong minim karena meskipun tingkat bunga deposito lebih tinggi dari tabungan biasa, deposito hanya dapat dibayarkan pada waktu tertentu, hal ini membuat deposito justru dapat digunakan sebagai alternatif bentuk investasi [3].

Cross-industry Standard Process for Data Mining (CRISP-DM) merupakan *platform* yang mengubah masalah bisnis menjadi tugas *data mining* dan melaksanakan proyek penambangan data secara independen dari area aplikasi dan teknologi yang digunakan. secara luas dianggap sebagai prinsip panduan yang paling relevan dan komprehensif untuk melaksanakan proyek analitik [36].

BAB III
METODOLOGI PENELITIAN

3.1. Tahapan Penelitian

Penelitian dilakukan menggunakan metode eksperimen. Eksperimen yang dilakukan dengan menerapkan CRISP-DM untuk mendapatkan informasi yang mendalam tentang nasabah yang akan menabung deposito, dan Model yang digunakan dalam penelitian ini yaitu:



Sumber: Hasil Penelitian (2021)

Gambar 3.1. Tahapan Model Penelitian

Gambar 3.1. menunjukan tahapan model penelitian dengan CRISP-DM, urutan tahapan tidak kaku, tahapan bergerak bolak balik terhadap tahapan yang berbeda. Tanda panah menunjukan alur kepentingan dan ketergantungan antar setiap tahapan. Penjelasan dari kelima tahapan metode dengan CRISP-DM yaitu:

1. Tahap *Business Understanding*

Tahap pertama CRISP-DM berfokus pada pemahaman tujuan penelitian, dan kemudian mengubah pemahaman ini ke dalam rumusan dan definisi masalah data mining. Pada tahap ini, penting untuk memahami bidang masalah dan menemukan solusi yang cocok untuk masalah yang ada, serta memahami faktor-faktor yang dapat mempengaruhi hasil penelitian. Penerapan data mining untuk mengklasifikasikan nasabah yang menabung deposito berdasarkan strategi pemasaran langsung. Diharapkan melalui penelitian ini bank dapat menerapkan strategi pemasaran yang tepat untuk meningkatkan loyalitas nasabah dan memperoleh nasabah baru.

2. Tahap *Data Understanding*

Tahap Data Understanding bertujuan untuk identifikasi dan pahami data yang dipunya. Selain itu, data harus diverifikasi. Data yang digunakan dalam penelitian ini adalah data *Bank marketing* yang diperoleh dari repositori UCI.

3. Tahap *Data Preparation*

Tahap *Data Preparation* mencakup semua aktivitas yang diperlukan untuk membuat kumpulan data akhir atau data yang akan dimasukan kedalam alat pemodelan dari data mentah awal. *Data Preparation* yang dilakukan dalam penelitian ini meliputi cek dan menghapus *missing value*, hapus atribut yang tidak digunakan, mengubah data kategori ke numerik, dan melakukan eksperimen berbeda yaitu data tidak seimbang dan data seimbang.

4. Tahap *Modelling*

Pemodelan yang diterapkan pada penelitian ini dengan membagi data *training* dan *testing* 80:20 dimana 80% dijadikan data *training* dan 20% data *testing*. Ada

dua eksperimen yang dilakukan yaitu menggunakan *imbalance* data dan *balance* data dimana dilakukan *oversampling* menggunakan teknik *Synthetic Minority Over-sampling Technique* (*SMOTE*) pada proses latih/ *training*.

5. Tahap *Evaluation*

Model yang dihasilkan diuji dengan menggunakan *confusion matrix*, yang menentukan tingkat akurasi. *confusion matrix* menggambarkan hasil akurasi, mulai dari prediksi positif benar, prediksi positif palsu, prediksi negatif benar, dan prediksi negatif palsu. Akurasi dihitung berdasarkan dari semua prediksi yang benar (positif dan negatif) dan dari semua data uji. Semakin tinggi nilai akurasi, semakin baik kinerja model. Pengujian juga diukur dengan kurva ROC. Kurva ROC menggambarkan derajat positif sebagai kurva. Uji dilakukan dengan menghitung luas daerah di bawah kurva (AUC). Semakin tinggi nilai ROC dan AUC maka semakin baik pembentukan model klasifikasi.

a. *Confusion Matrix*

Confusion Matrix merupakan visualisasi untuk mengevaluasi dari kinerja model klasifikasi. Untuk melakukan klasifikasi evaluasi komparatif, maka dalam penelitian ini menggunakan *Confusion Matrix*. *Confusion Matrix* ini meliputi informasi tentang kelas yang sebenarnya dan kelas prediksi. Hal ini akan ditemukan pada kolom matriks yang mewakili kelas yang diprediksi, sedangkan setiap baris mewakili kejadian pada kelas tersebut. *Confusion Matrix* adalah salah satu alat ukur berbentuk matrik 2x2 yang digunakan untuk mendapatkan jumlah ketepatan algoritma yang dipakai.

Tabel 3.1. *Confusion Matrix*

True *Condition*	***Predicted Condition***	
	Positive	***Negative***
Positive	*True Positive* (TP)	*False Negative* (FN)
Negative	*False Positive* (FP)	*True Negative* (TN)

Sumber: Meng, Yang, Qian dan Zhang (2020) [52]

Untuk menghitung performa dari model klasifikasi dapat dihitung dengan menggunakan beberapa cara yaitu:

1) *Accuracy*

Dengan menggunakan rumus di bawah ini akan didapatkan akurasi dari matriks yang mengukur tentang rasio kebenaran dari prediksi dari seluruh data yang dievaluasi.

$$\frac{TP+TN}{TP+FP+TN+FN} \quad \text{..... (3.1)}$$

2) *Precision*

Dengan menggunakan rumus di bawah akan mengukur pola positif yang diprediksi dengan benar dari total pola yang diprediksi di kelas positif.

$$\frac{TP}{TP+FP} \quad \text{.....(3.2)}$$

3) *Recall*

Dengan menggunakan rumus di bawah akan mengukur fraksi pola positif yang diklasifikasikan dengan benar.

$$\frac{TP}{TP+TN} \quad \text{.....(3.3)}$$

4) *F-measure*

Dengan menggunakan rumus di bawah ini merupakan nilai *comprefensif* dari *recall* dan *precision.*

$$\frac{2\ x\ precision\ x\ Recall}{precision+\ Recall} \quad \text{.....(3.4)}$$

Keterangan:

TP : Jumlah kasus positif yang tergolong positif

FP : Jumlah kasus negatif yang tergolong positif

TN : Jumlah kasus negatif yang tergolong negatif

FN : Jumlah kasus positif yang tergolong negatif

b. ROC Curve

Kurva ROC adalah metode untuk melihat, mengelola, dan memilih klasifikasi berdasarkan model yang diperlukan. ROC juga dapat digunakan untuk menjelaskan tingkat sensitivitas dan spesifisitas.

Performance akurasi *Area Under Curve* dapat diklasifikasikan menjadi lima kelompok yang terlihat pada tabel 3.2 sebagai berikut:

Tabel 3.2. *Performance* Keakurasian AUC

Performance	Klasifikasi
0.90 – 1.00	*Excellent Classification*
0.80 – 0.90	*Good Classification*
0.70 – 0.80	*Fair Classification*
0.60 – 0.70	*Poor Classification*
0.50 – 0.60	*Failure*

Sumber: Luque, Carrasco, Martín, Heras (2019) [53]

BAB IV
HASIL PENELITIAN DAN PEMBAHASAN

Berdasarkan metodologi penelitian pada bab 3, maka pada bab ini akan dijelaskan implementasi metodologi penelitian yang dilakukan adalah sebagai berikut:

4.1. *Business Understanding*

Pada tahap ini dilakukan pemahaman terhadap objek penelitian. Pemahaman objek dilakukan dengan menggali data dan informasi melalui dataset pada *Bank marketing*. Tujuan dari penelitian ini adalah melakukan klasifikasi terhadap nasabah bank yang menabung deposito. Klasifikasi dilakukan dengan algoritma *Extreme Gradient Boosting (XGBoost)* yang dikomparasi dengan algoritma lain seperti *Decision Tree, K-Nearest Neighbor* dan *Logistic Regression.* Diharapkan melalui penelitian ini bank dapat menerapkan strategi pemasaran yang tepat untuk meningkatkan loyalitas nasabah dan memperoleh nasabah baru.

4.2. *Data Understanding*

Tahap *data understanding* merupakan proses pemahaman pada dataset *Bank marketing*. Data ini mengacu pada kegiatan pemasaran langsung dari lembaga perbankan Portugis. Kegiatan pemasaran didasarkan melalui telepon. Untuk mengetahui nasabah menabung deposito "Ya" atau "Tidak". Dalam tahap ini peneliti memahami data dan menentukan model yang akan digunakan dalam penelitian. Berikut ini data yang digunakan sebagai penelitian.

A. Menganalisa dataset

Dalam menganalisa dataset perlu mengetahui data apa yang ada pada dataset. Sehingga dapat mengetahui proses selanjutnya yang harus dilakukan untuk mengolah data set tersebut dengan melihat atribut dan jenis datanya.

Pada Dataset *bank marketing* memiliki 41.188 data dengan 21 atribut. dengan satu target yaitu atribut ‘y’ dengan dua kelas yes dan no, berikut rincian dari dataset *bank marketing* pada Tabel 4.1.

Tabel 4.1 Dataset *Bank marketing*

No	Atribut	Keterangan	Type
1	*Age*	Usia Nasabah	Numerik
2	*Job*	jenis pekerjaan nasabah	Kategori
3	*Marital*	Status perkawinan	Kategori
4	*Education*	Menunjukkan tingkat pendidikan setiap pelanggan	Kategori
5	*Default*	Apakah pelanggan memiliki kredit secara *default*	Kategori
6	*Housing*	Apakah pelanggan memiliki pinjaman perumahan?	Kategori
7	*Loan*	Apakah pelanggan memiliki pinjaman pribadi	Kategori
8	*Contact*	Jenis komunikasi kontak seluler atau telepon	Kategori
9	*Month*	Kontak terakhir bulan	Kategori
10	*Day_of_week*	hari kontak terakhir dalam seminggu	Kategori
11	*Duration*	Durasi kontak terakhir dalam hitungan detik.	Numerik
12	*Campaign*	Jumlah kontak yang dilakukan untuk klien selama penawaran	Numerik
13	*Pdays*	jumlah hari yang berlalu setelah klien terakhir dihubungi dari kampanye	Numerik
14	*Previous*	jumlah kontak yang dilakukan sebelum kampanye ini dan untuk klien ini	Numerik
15	*Poutcome*	hasil dari penawaran sebelumnya	Kategori
16	*Emp.var.rate*	Tingkat variasi pekerjaan	Numerik
17	*Cons.price.idx*	hasil dari penawaran sebelumnya	Numerik
18	*Cons.conf.id*	Indeks kepercayaan konsumen	Numerik
19	*Euribor3m*	Tarif suku bunga euribor 3 bulan	Numerik
20	*Nr.employed*	Jumlah karyawan	Numerik
21	*y*	Apakah nasabah telah menabung deposito berjangka	*Binary*

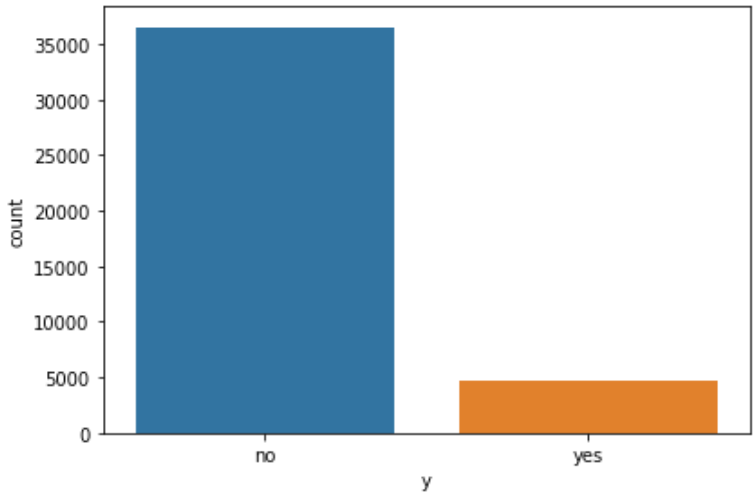
B. Memvisualisasikan dataset

Agar lebih mudah dalam melihat dataset yang ada pada *bank marketing*, maka divisualisasikan dalam bentuk diagram untuk melihat isi dataset dalam setiap *feature*.

Sumber: Hasil Penelitian (2021)

Gambar 4.2 Hubungan variabel kategori dengan target

Pada gambar 4.2 dapat dilihat isi data pada *feature* dengan tipe kategori yang ada pada dataset *Bank marketing*. Nasabah yang bekerja sebagai admin, teknisi, dan buruh lebih cenderung menabung deposito. Nasabah yang sudah menikah memiliki tingkat menabung deposito yang tinggi. Nasabah yang berpendidikan universitas cenderung menabung deposito. Nasabah yang tidak memiliki kredit default lebih cenderung menabung deposito. Selama bulan Mei hingga Agustus nasabah menunjukkan minat yang tinggi untuk menabung deposito. Nasabah yang memiliki pinjaman pribadi tampaknya kurang tertarik dengan deposito. Nasabah yang dihubungi melalui 'seluler' cenderung menabung deposito.



Sumber: Hasil Penelitian (2021)

Gambar 4.4 Data target

Pada gambar 4.4 dapat dilihat data yang menjadi target yaitu atribut "y" memiliki jumlah data yang tidak sama antara "*yes*" dan "*no*". dengan jumlah "*yes*" ada 36548 sedangkan "*no*" ada 4640, artinya dataset *bank marketing* memiliki data yang tidak seimbang atau *imbalance* data.

4.3. *Data Preparation*

Pada tahap *data preparation* ini akan melakukan *data preprocessing* terlebih dahulu terhadap *dataset* sebelum digunakan dan masuk ke tahap *modelling*, bertujuan untuk mendapatkan data yang bersih dan siap untuk digunakan dalam penelitian agar nantinya data yang digunakan dapat lebih maksimal saat diterapkan dengan algoritma data mining.

4.3.1. *Data Preprocessing*

Pada tahap *preprocessing* digunakan untuk membersihkan data. Dan berikut data *preprocessing* yang dilakukan pada penelitian ini.

1. Cek dan mengatasi *Missing Value*

Tahap ini digunakan untuk melihat apakah terdapat missing value pada dataset, missing value biasanya merupakan data yang hilang atau bernilai "0", jika ada maka perlu untuk diatasi.

Berdasarkan hasil cek *missing value* yang dilakukan tidak terdapat *missing value* atau data yang bernilai 0 pada dataset. artinya dataset sudah bersih.

2. *Label Encoding*

Tahap *Label Encoding* digunakan untuk merubah tipe data kategori menjadi numerik. Untuk mengonversi label kata menjadi angka, kita perlu menggunakan generator *label encoding*. *Label Encoding* mengacu pada proses mengubah label kata untuk dijadikan bentuk numerik. Jadi jika data sudah menjadi numerik, maka kita dapat menggunakannya secara langsung untuk memulai pelatihan agar data dapat dihitung untuk melakukan prediksi.

3. Menghapus atribut *Pdays*

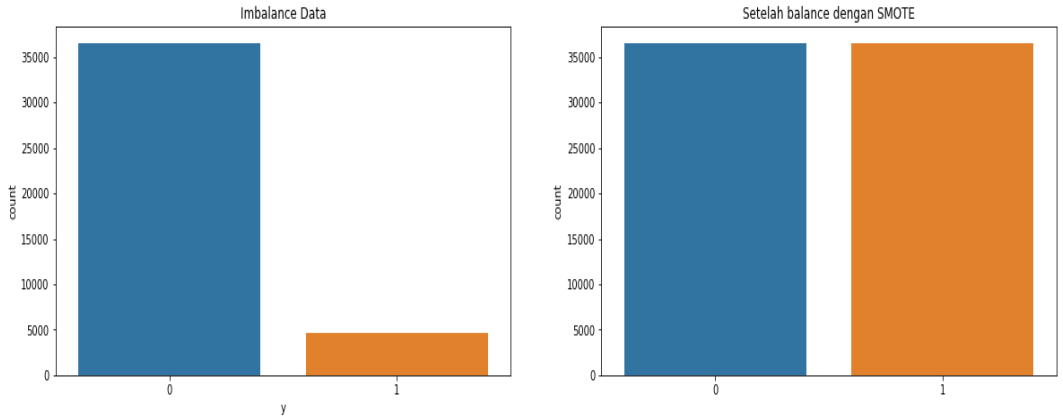
Atribut Pdays tidak digunakan dan dihapus karena memiliki nilai 999 yang artinya nasabah tidak dihubungi sekitar 90%.

4. Mengatasi *Imbalance* data

Berdasarkan Analisa yang dilakukan diketahui bahwa dataset yang digunakan merupakan data tidak seimbang atau *imbalance* data. Untuk mengatasi hal tersebut maka data yang tidak seimbang dilakukan *oversampling* dengan teknik *Synthetic Minority Over-sampling Technique* (*SMOTE*) pada tahap *training*.

4.4. *Modelling*

Tahap M*odelling* adalah pemilihan teknik data mining dengan menentukan algoritma yang digunakan. Dalam tahap *Modelling* ini dilakukan teknik pengklasifikasian data dengan algoritma *Extreme Gradient Boosting (XGBoost)* yang dikomparasi dengan algoritma lain seperti *Decision Tree, K-Nearest Neighbor* dan *Logistic Regression* dengan penerapan *split training testing* 80:20 dimana dari 41188 dibagi dua yaitu 80% (32950) data *training* dan 20% (8238) data *testing.* Pada data *training* dilakukan *oversampling* dengan teknik *Synthetic Minority Over-sampling Technique* (*SMOTE)*.



Sumber: Hasil Penelitian (2021)

Gambar 4.5 Mengatasi *Imbalance* data

Pada Gambar 4.5 menunjukkan hasil data asli atau yang tidak seimbang (*imbalance data*) dengan data yang seimbang (*balance*) yang sudah di *oversampling* dengan teknik *Synthetic Minority Over-sampling Technique* (*SMOTE*). Dimana terdapat 32950 data *training* yaitu:

Sebelum *oversampling*: jumlah label ‘1’ = 3721

Sebelum *oversampling*: jumlah label ‘0’ = 29229

Setelah *oversampling*: jumlah label ‘1’ = 29229

Setelah *oversampling*: jumlah label ‘0’ = 29229

Jadi setelah di *oversampling* dengan teknik *SMOTE* pada data *training*, jumlah data pada atribut “y” sudah seimbang. Kemudian dioptimalkan dengan *hyperparameter tuning* menggunakan GridsearchCV untuk menentukan parameter yang digunakan.

4.4.1. *Extreme Gradient Boosting (XGBoost)*

Pada model yang dibangun dengan algoritma klasifikasi *XGBoost* menerapkan *split training testing* 80:20 dimana 80 % data *training* dan 20% data *testing*, kemudian model yang dibangun dioptimalkan menggunakan GridsearchCV dengan cv=10, artinya data dibagi menjadi 10 bagian yaitu 9 data *training* dan 1 data *testing* dan dilakukan 10 kali dengan parameter objective='binary: logistic', gamma=0, learning_rate=0.1, max_depth=6, n_estimators=200, min_child_weight=5 dan memvariasikannya seperti min_child_weight: [1, 3, 5, 10]

dan max_depth: [3, 5, 7, 9] yang diujikan untuk mendapatkan *best score* dengan parameter terbaik.

4.4.2. *Decision Tree*

Pada model yang dibangun dengan algoritma klasifikasi *Decision Tree* menerapkan *split training testing* 80:20 dimana 80 % data *training* dan 20% data *testing*, kemudian model yang dibangun dioptimalkan menggunakan GridsearchCV dengan cv=10, artinya data dibagi menjadi 10 bagian yaitu 9 data *training* dan 1 data *testing* dan dilakukan 10 kali dengan parameter criterion='gini', splitter='best', max_depth=22, min_samples_split=3, min_samples_leaf=1, random_state=0 dan memvariasikannya seperti criterion: ['gini', 'entropy'], splitter: ['best', 'random'], max_depth: [20, 22, 28, 32, 37, 38, 42, 45, 50, 70], min_samples_leaf: [1, 2, 3, 4, 5] yang diujikan untuk mendapatkan *best score* dengan parameter terbaik.

4.4.3. *K-Nearest Neighbor*

Pada model yang dibangun dengan algoritma klasifikasi *K-Nearest Neighbor* menerapkan *split training testing* 80:20 dimana 80 % data *training* dan 20% data *testing*, kemudian model yang dibangun dioptimalkan menggunakan GridsearchCV dengan cv=10, artinya data dibagi menjadi 10 bagian yaitu 9 data *training* dan 1 data *testing* dan dilakukan 10 kali dengan parameter n_neighbors=6, weights='uniform', algorithm='auto', leaf_size=30, p=2, metric='minkowski', metric_params=None, n_jobs=-1 dan memvariasikannya seperti n_neighbors: [1, 2, 3, 4, 5, 6, 7, 8], weights: ["uniform", "distance"], leaf_size: [10, 20, 30, 40] yang diujikan untuk mendapatkan *best score* dengan parameter terbaik.

4.4.4. *Logistic Regression*

Pada model yang dibangun dengan algoritma klasifikasi *logistic Regression* menerapkan *split training testing* 80:20 dimana 80 % data *training* dan 20% data *testing*, kemudian model yang dibangun dioptimalkan menggunakan GridsearchCV dengan cv=10, artinya data dibagi menjadi 10 bagian yaitu 9 data *training* dan 1 data *testing* dan dilakukan 10 kali dengan parameter penalty='l2', C=1.0, random_state=None, solver='lbfgs', max_iter=150, multi_class='auto', verbose=0

dan memvariasikannya seperti C: [1.0, 2.0, 3.0, 4.0, 5.0, 6.0], max_iter: [10, 100, 1000], penalty: ['l1', 'l2'] multi_class: ['auto', 'ovr', 'multinomial'] yang diujikan untuk mendapatkan *best score* dengan parameter terbaik.

4.5. *Evaluation*

Dari hasil *modeling* yang telah dilakukan sebelumnya didapat *best score* dan parameter terbaik yang kemudian diterapkan untuk menghasilkan *Confusion Matrix* dan Kurva ROC dari algoritma klasifikasi *XGBoost, Decision Tree, KNN* dan *Logistic Regression* untuk dibandingkan hasil akurasi yang telah didapat.

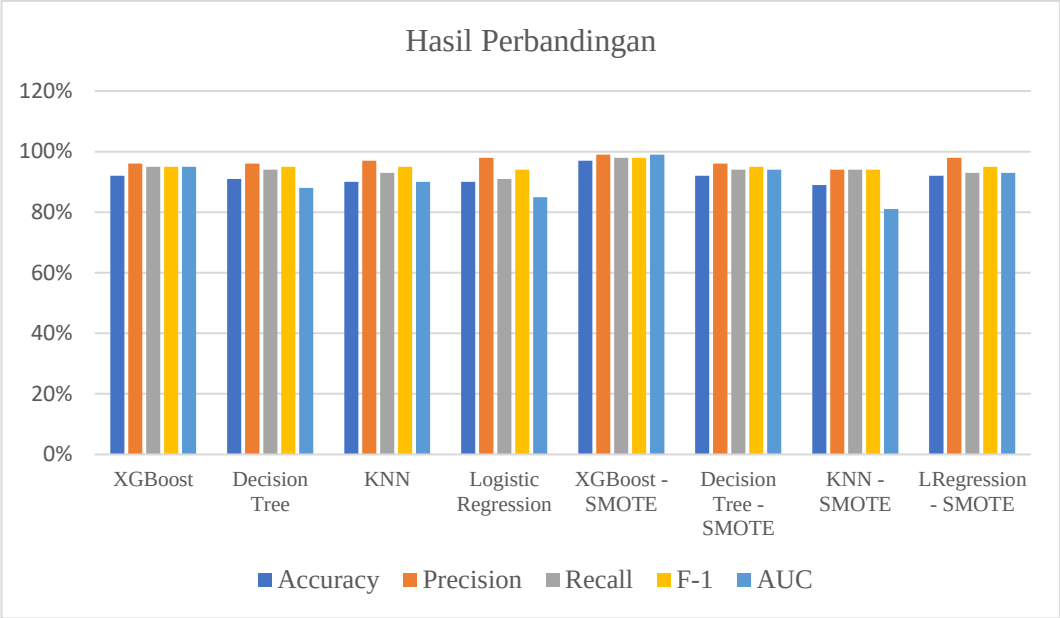
4.5.1. Rangkuman Hasil

Adapun perbandingan rangkuman hasil *accuracy, precision, recall* dan AUC Algoritma klasifikasi *Extreme Gradient Boosting (XGBoost)* yang dikomparasi dengan algoritma lain seperti, *Decision Tree, K-Nearest Neighbor* dan *Logistic Regression* yang dibedakan menjadi dua data tidak seimbang dan data seimbang sebagai berikut:

Tabel 4.12 Perbandingan hasil penelitian

Algoritma	***Accuracy***	***Precision***	***Recall***	***F-1***	***AUC***
XGBoost	92%	96%	95%	95%	0.95
Decision Tree	91%	96%	94%	95%	0.88
KNN	90%	97%	93%	95%	0.90
Logistic Regression	90%	98%	91%	94%	0.85
XGBoost - SMOTE	**97%**	99%	98%	98%	**0.99**
Decision Tree - SMOTE	92%	96%	94%	95%	0.94
KNN - *SMOTE*	89%	94%	94%	94%	0.81
L Regression - *SMOTE*	92%	98%	93%	95%	0.93

Sumber: Hasil penelitian (2021)



Sumber: Hasil penelitian (2021)

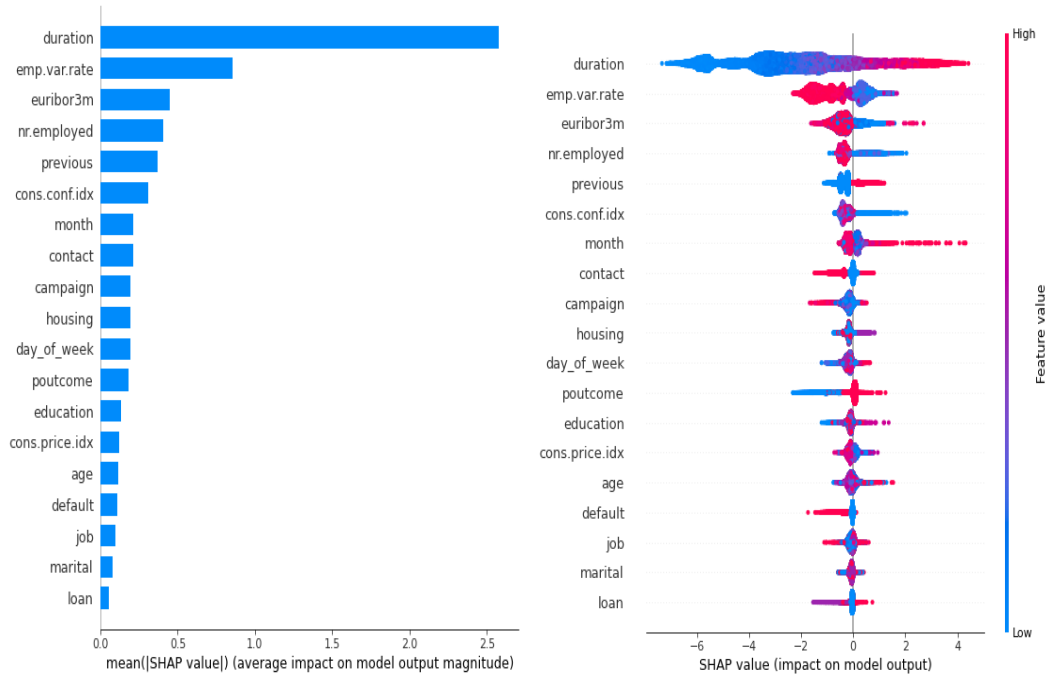
Gambar 4.22 Gambar Hasil perbandingan

Berdasarkan Tabel 4.22 Hasil perbandingan perhitungan dengan algoritma *XGBoost* mendapatkan nilai akurasi 92% dengan AUC 0.95, perhitungan dengan algoritma *Decision Tree* mendapatkan nilai akurasi 91% dengan AUC 0.88, perhitungan dengan algoritma KNN mendapatkan nilai akurasi 90% dengan AUC 0.90, perhitungan dengan algoritma *Logistic Regression* mendapatkan nilai akurasi 90% dengan AUC 0.85, Sedangkan perhitungan dengan algoritma *XGBoost – SMOTE* mendapatkan nilai akurasi 97% dengan AUC 0.99, perhitungan dengan algoritma *Decision Tree – SMOTE* mendapatkan nilai akurasi 92% dengan AUC 0.92, perhitungan dengan algoritma KNN – *SMOTE* mendapatkan nilai akurasi 89% dengan AUC 0.81, perhitungan dengan algoritma *Logistic Regression – SMOTE* mendapatkan nilai akurasi 92% dengan AUC 0.93.

Dapat disimpulkan dari hasil komparasi diatas algoritma *XGBoost-SMOTE* mendapat akurasi tertinggi dibanding algoritma lainnya berdasarkan nilai akurasi 97% dan AUC 0.99 dan *best score* 0.953 dengan parameter max_depth= 7, min_child_weight= 1. Kemudian berdasarkan hasil tersebut alhoritma *XGBoost-SMOTE* akan diterapkan pada *feature important* dengan *SHAPE value* untuk melihat atribut apa yang paling berkontribusi terhadap model yang dibangun.

4.5.2. *Feature Important* dengan *SHAP Value*

Untuk menginterpretasikan model yang diimplementasikan pada *XGBoost* dengan baik, penelitian ini menerapkan SHAP *value*, hal ini dilakukan untuk mengetahui *feature* yang paling mempengaruhi akurasi terhadap model yang telah dibuat berdasarkan tingkatan atau ranking, terutama pada algoritma *XGBoost* yang memiliki hasil akurasi tertinggi dari algoritma lainya. Berikut adalah *feature important* berdasarkan hasil penelitian yang sudah dilakukan:



Sumber: Hasil penelitian (2021)

Gambar 4.23 *SHAPE value*

Berdasarkan gambar SHAP *Value* yang dihasilkan dengan algoritma *XGBoost – SMOTE* yang merupakan algoritma dengan akurasi terbaik. *feature* diurutkan berdasarkan besaran efek terhadap model yang dibangun. Pertama dapat disimpulkan bahwa kontribusi fitur bervariasi dan berbeda, dengan beberapa fitur spesifik yang berkontribusi jauh lebih banyak daripada fitur lainnya. Seperti *feature* duration yang lebih mendominasi dari *feature* lainya. Kemudian diketahui bahwa dalam beberapa *feature* berkontribusi sangat sedikit pada keluaran model, seperti loan, merital, job.

BAB V
PENUTUP

Berdasarkan dari penelitian yang telah dilakukan maka dapat diambil beberapa kesimpulan yaitu penerapan data mining dapat digunakan untuk memprediksi calon nasabah yang akan menabung deposito. Dataset yang digunakan merupakan dataset tidak seimbang (*imbalance data*), kemudian data yang tidak seimbang diolah dengan *oversampling* menggunakan teknik *SMOTE* pada data *training*, dan dari data yang tidak seimbang dan seimbang dimodelkan dengan algoritma *Extreme Gradient Boosting (XGBoost) Decision Tree*, *K-Nearest Neighbor* dan *Logistic Regression*. Menunjukan bahwa data yang melalui proses *oversampling* dengan *SMOTE* mengalami peningkatan serta memiliki akurasi dan AUC lebih tinggi.

Pada semua model yang diuji algoritma *XGBoost-SMOTE* memiliki hasil paling tinggi yaitu akurasi 97% dan AUC 99% termasuk kategori *Excellent Classification*. Hal ini menunjukan bahwa *XGBoost-SMOTE* mampu memprediksi nasabah yang akan menabung deposito di perbankan lebih baik dibanding algoritma lain yang diujikan.

Berdasarkan *Feature Important* dari model yang dibangun dengan algoritma *XGBoost-SMOTE* menggunakan SHAP *Value* didapatkan informasi bahwa *feature "duration"* memiliki kontribusi lebih terhadap model yang dibangun dibandingkan dengan *feature* yang lain. Dapat disimpulkan bahwa ketertarikan nasabah terhadap produk yang ditawarkan terutama deposito melalui panggilan telepon dapat dipengaruhi oleh nasabah yang menggali informasi lebih saat panggilan berlangsung. Artinya strategi yang harus dilakukan oleh perbankan dalam melakukan penawaran produk secara langsung harus melatih *customer service* terutama yang berada dalam divisi *marketing* untuk menguasai *public speaking* yang bagus dan *product knowledge* atau penguasaan produk yang baik. Sehingga nasabah yang ditawarkan deposito akan tertarik dan akan menabung deposito karena penyampaian yang jelas dan mudah dipahami serta informasi yang lengkap.

Berdasarkan hasil penelitian dapat diberikan beberapa saran yang dapat dilakukan untuk penelitian selanjutnya sebagai berikut:

1. Penelitian selanjutnya dapat menerapkan metode *over sampling* yang berbeda dalam mengatasi *imbalance* data agar mendapatkan hasil yang lebih maksimal.
2. Penelitian selanjutnya dapat menggunakan algoritma yang lain serta dapat juga menggunakan algoritma *Deep Learning*.
3. Penelitian selanjutnya dapat memvariasikan lebih banyak *hyperparameter* yang diuji untuk melihat performa model yang lebih baik.
4. Diharapkan untuk pihak yang berkaitan dengan *bank marketing* ataupun pemasaran lain yang dilakukan secara langsung, untuk bisa meningkatkan kualitas layanan dan penyampaian informasi yang jelas untuk bisa menarik pelanggan.

DAFTAR REFERENSI

[1] T. Online *et al.*, “ANALISIS FATWA DSN MUI TENTANG DEPOSITO DITINJAU DARI ASPEK USHUL FIQH Oleh : Kartini Kata Kunci : analisis fatwa dsn MUI , deposito , Aspek ushul fiqh Pendahuluan Sebagaimana syariah dewasa diketehui ini bank menggunakan sistim dan operasi nya berdasarka,” vol. 1, no. 1, 2021.

[2] N. 10 T. 1998 UU RI, *Undang-Undang RI No. 10 Tahun 1998 tentang Perbankan*. 1998, p. 182.

[3] R. Ula, “PENGARUH CAPITAL ADEQUACY RATIO (CAR), INFLASI , DAN SUKU BUNGA SERTIFIKAT BANK INDONESIA (SBI) TERHADAP TINGKAT SUKU BUNGA DEPOSITO BERJANGKA (Studi Pada Perusahaan Bank Pembangunan Daerah di Indonesia Periode 2010-2015),” vol. 56, no. 1, 2018.

[4] S. M. Kostic, M. Duricic, M. I. Simic, and M. V. Kostic, “Data Mining and Modeling use Case in Banking Industry,” *2018 26th Telecommun. Forum, TELFOR 2018 - Proc.*, pp. 1–4, 2018, doi: 10.1109/TELFOR.2018.8611897.

[5] M. Arhami and M. Nasir, *Data Mining - Algoritma dan Implementasi*. Penerbit Andi, 2020.

[6] J. Suntoro, *Data Mining: Algoritma dan Implementasi dengan Pemrograman PHP*. 2019.

[7] P. R. S. Moro,P. Cortez, “Bank Marketing Data Set,” *2012*, 2012. https://archive.ics.uci.edu/ml/datasets/Bank+Marketing.

[8] J. SUNTORO, “DATA MINING : Algoritma dan Implementasi dengan Pemrograman php.” p. 179, 2019.

[9] Sudirman, A. P. Windarto, and A. Wanto, “Data mining tools | rapidminer: K-means method on clustering of rice crops by province as efforts to stabilize food crops in Indonesia,” *IOP Conf. Ser. Mater. Sci. Eng.*, vol. 420, no. 1, 2018, doi: 10.1088/1757-899X/420/1/012089.

[10] P. Wongchinsri and W. Kuratach, “A survey - Data mining frameworks in credit card processing,” *2016 13th Int. Conf. Electr. Eng. Comput. Telecommun. Inf. Technol. ECTI-CON 2016*, 2016, doi:

10.1109/ECTICon.2016.7561287.

[11] A. Hemeida, R. Mansour, and M. E. Hussein, “Multilevel Thresholding for Image Segmentation Using an Improved Electromagnetism Optimization Algorithm,” *Int. J. Interact. Multimed. Artif. Intell.*, vol. 5, no. 4, p. 102, 2019, doi: 10.9781/ijimai.2018.09.001.

[12] L. Muflikhah, D. E. Ratnawati, and R. R. M. Putri, *Buku Ajar Data Mining*, Cetakan pe. Malang: Universitas Brawijaya Press, 2018.

[13] T. Chen and C. Guestrin, “XGBoost: A Scalable Tree Boosting System,” *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discov. Data Mining, San Fr. CA, USA*, pp. 785–794, 2016.

[14] H. Shi, H. Wang, Y. Huang, L. Zhao, C. Qin, and C. Liu, “A hierarchical method based on weighted extreme gradient boosting in ECG heartbeat classification,” *Comput. Methods Programs Biomed.*, vol. 171, pp. 1–10, 2019, doi: 10.1016/j.cmpb.2019.02.005.

[15] D. Yu *et al.*, “Copy number variation in plasma as a tool for lung cancer prediction using Extreme Gradient Boosting (XGBoost) classifier,” *Thorac. Cancer*, vol. 11, no. 1, pp. 95–102, 2020, doi: 10.1111/1759-7714.13204.

[16] C. Hu, J. Yan, and C. Wang, “Advanced Cyber-Physical Attack Classification with Extreme Gradient Boosting for Smart Transmission Grids,” *IEEE Power Energy Soc. Gen. Meet.*, vol. 2019-Augus, 2019, doi: 10.1109/PESGM40551.2019.8973679.

[17] M. Lin, X. Zhu, T. Hua, X. Tang, G. Tu, and X. Chen, “Detection of Ionospheric Scintillation Based on XGBoost Model Improved by SMOTE-ENN Technique,” *Remote Sens*, pp. 1–22, 2021, doi: https://doi.org/10.3390/rs13132577.

[18] M. Alqahtani, A. Gumaei, H. Mathkour, and M. M. Ben Ismail, “A genetic-based extreme gradient boosting model for detecting intrusions in wireless sensor networks,” *Sensors (Switzerland)*, vol. 19, no. 20, 2019, doi: 10.3390/s19204383.

[19] S. B. Yang and T. L. Chen, “Uncertain decision tree for bank marketing classification,” *J. Comput. Appl. Math.*, vol. 371, p. 112710, 2020, doi: 10.1016/j.cam.2020.112710.

[20] H. H. Patel and P. Prajapati, “Study and Analysis of Decision Tree Based Classification Algorithms,” *Int. J. Comput. Sci. Eng.*, vol. 6, no. 10, pp. 74–78, 2018, doi: 10.26438/ijcse/v6i10.7478.

[21] S. Rizkia, E. Budi Setiawan, and D. Puspandari, “Analisis Sentimen Kepuasan Pelanggan Terhadap Internet Provider Indihome di Twitter Menggunakan Metode Decision Tree dan Pembobotan TF-IDF,” *e-Proceeding Eng.*, vol. 6, no. 2, pp. 9683–9693, 2019.

[22] I. A. Nikmatun and I. Waspada, “Implementasi Data Mining untuk Klasifikasi Masa Studi Mahasiswa Menggunakan Algoritma K-Nearest Neighbor,” *J. SIMETRIS*, vol. 10, no. 2, pp. 421–432, 2019.

[23] W. Hou, D. Li, C. Xu, H. Zhang, and T. Li, “An Advanced k Nearest Neighbor Classification Algorithm Based on KD-tree,” *Proc. 2018 IEEE Int. Conf. Saf. Prod. Informatiz. IICSPI 2018*, pp. 902–905, 2019, doi: 10.1109/IICSPI.2018.8690508.

[24] D. A. Adeniyi, Z. Wei, and Y. Yongquan, “Automated web usage data mining and recommendation system using K-Nearest Neighbor (KNN) classification method,” *Appl. Comput. Informatics*, vol. 12, no. 1, pp. 90–108, 2016, doi: 10.1016/j.aci.2014.10.001.

[25] I. Aghaei and A. Sokhanvar, “Factors influencing SME owners’ continuance intention in Bangladesh: a logistic regression model,” *Eurasian Bus. Rev.*, vol. 10, no. 3, pp. 391–415, 2020, doi: 10.1007/s40821-019-00137-6.

[26] S. Solutions, “What is Logistic Regression?,” *Statistics Solutions*. https://www.statisticssolutions.com/free-resources/directory-of-statistical-analyses/what-is-logistic-regression/ (accessed Jul. 02, 2021).

[27] S. S. Nasim, C. M. Mizan, T. Chakroborty, S. Ghosh, and S. Karmakar, *A Survey of Load Balanced Job Scheduling Schemes in Cloud Computing*, vol. 1187. 2021.

[28] S. Maldonado, J. López, and C. Vairetti, “An alternative SMOTE oversampling strategy for high-dimensional datasets,” *Appl. Soft Comput. J.*, vol. 76, pp. 380–389, 2019, doi: 10.1016/j.asoc.2018.12.024.

[29] J. He, Y. Hao, and X. Wang, “An interpretable aid decision-making model for flag state control ship detention based on SMOTE and XGboost,” *J. Mar.*

Sci. Eng., vol. 9, no. 2, pp. 1–19, 2021, doi: 10.3390/jmse9020156.

[30] B. Raharjo, *Mudah Belajar Python Untuk Aplikasi Dekstop Dan Web Edisi Revisi*. Bandung, 2019.

[31] J. Enterprise, *Python untuk Programmer Pemula*. Jakarta: PT Elex Media Komputindo, 2019.

[32] J. Hao and T. K. Ho, "Machine Learning Made Easy: A Review of Scikit-learn Package in Python Programming Language," *J. Educ. Behav. Stat.*, vol. 44, no. 3, pp. 348–361, 2019, doi: 10.3102/1076998619832248.

[33] K. Gede and R. Aditya, "Program Menghitung Banyak Bata pada Ruangan Menggunakan Bahasa Python," vol. 2, no. 1, 2021.

[34] Jubilee Enterprise, *Python untuk Programmer Pemula*. PT Elex Media Kamputindo, 2019.

[35] C. Diana and M. S. Tanjung, "SISTEM TRANSAKSI DAN PERHITUNGAN BUNGA DEPOSITO PADA BANK NAGARI CABANG PARIAMAN," *CC-By Attrib. 4.0 Int.*, pp. 1–8, 2019, doi: 10.31219/osf.io/exct7.

[36] S. Jaggia, A. Kelly, K. Lertwachara, and L. Chen, "Applying the CRISP-DM Framework for Teaching Business Analytics," *Decis. Sci. J. Innov. Educ.*, vol. 18, no. 4, pp. 612–634, 2020, doi: 10.1111/dsji.12222.

[37] T. Darmawan, A. S. Birawa, E. Eryanto, and T. Mauritsius, "Credit classification using crisp-dm method on bank abc customers," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 6, pp. 2375–2380, 2020, doi: 10.30534/ijeter/2020/28862020.

[38] M. Syukron, R. Santoso, and T. Widiharih, "PERBANDINGAN METODE SMOTE RANDOM FOREST DAN SMOTE XGBOOST UNTUK KLASIFIKASI TINGKAT PENYAKIT HEPATITIS C PADA IMBALANCE CLASS DATA," *J. Gaussian*, vol. 9, no. 3, pp. 227–236, 2020, doi: 10.14710/j.gauss.v9i3.28915.

[39] S. Wang *et al.*, "A new method of diesel fuel brands identification: SMOTE oversampling combined with XGBoost ensemble learning," *Fuel*, vol. 282, no. July, p. 118848, 2020, doi: 10.1016/j.fuel.2020.118848.

[40] S. M. H. Mahmud, W. Chen, H. Jahan, Y. Liu, N. I. Sujan, and S. Ahmed,

"IDTi-CSsmoteB: Identification of Drug-Target Interaction Based on Drug Chemical Structure and Protein Sequence Using XGBoost with Over-Sampling Technique SMOTE," *IEEE Access*, vol. 7, pp. 48699–48714, 2019, doi: 10.1109/ACCESS.2019.2910277.

[41] S. R. Mousa, P. R. Bakhit, and S. Ishak, "An extreme gradient boosting method for identifying the factors contributing to crash/near-crash events: A naturalistic driving study," *Can. J. Civ. Eng.*, vol. 46, no. 8, pp. 712–721, 2019, doi: 10.1139/cjce-2018-0117.

[42] A. Abu-Srhan, S. Al zghoul, B. Alhammad, and R. Al-Sayyed, "Visualization and analysis in bank direct marketing prediction," *Int. J. Adv. Comput. Sci. Appl.*, vol. 10, no. 7, pp. 651–657, 2019, doi: 10.14569/ijacsa.2019.0100785.

[43] M. S. Basarslan and I. D. Argun, "Classification of a bank data set on various data mining platforms | Bir Banka Müşteri Verilerinin Farkli Veri Madenciliği Platformlarinda Siniflandirilmasi," in *2018 Electric Electronics, Computer Science, Biomedical Engineerings' Meeting, EBBT 2018*, 2018, pp. 1–4.

[44] L. Geetha, K. Anusha, D. Ganeshgoud, S. M. Gouse, and V. N. Mandhala, "Predicting the success of bank marketing system using classification techniques," *J. Adv. Res. Dyn. Control Syst.*, vol. 12, no. 2, pp. 1961–1968, 2020, doi: 10.5373/JARDCS/V12I2/S20201241.

[45] V. U. Pugliese, C. M. Hirata, and R. D. Costa, "Comparing supervised classification methods for financial domain problems," *ICEIS 2020 - Proc. 22nd Int. Conf. Enterp. Inf. Syst.*, vol. 1, no. Iceis, pp. 440–451, 2020, doi: 10.5220/0009426204400451.

[46] Y. Chen, "Analysis of applying Genetic Algorithm to Simple Neural Network Based on Bank Direct Marketing Dataset," pp. 1–6, 2018.

[47] K. Nizam, A. Halim, A. Syukor, M. Jaya, A. Firdaus, and A. Fadzil, "Data Pre-Processing Algorithm for Neural Network Binary Classification Model in Bank Tele-Marketing," *Int. J. Innov. Technol. Explor. Eng.*, vol. 9, no. 3, pp. 272–277, 2020, doi: 10.35940/ijitee.c8472.019320.

[48] S. Mishra, "Handling Imbalanced Data : SMOTE vs . Random

Undersampling," *Int. Res. J. Eng. Technol.*, vol. 4, no. 8, pp. 317–320, 2017, [Online]. Available: https://irjet.net/archives/V4/i8/IRJET-V4I857.pdf.

[49] G. Marinakos and S. Daskalaki, "Imbalanced customer classification for bank direct marketing," *J. Mark. Anal.*, vol. 5, no. 1, pp. 14–30, 2017, doi: 10.1057/s41270-017-0013-7.

[50] I. Syarif, A. Prugel-Bennett, and G. Wills, "SVM Parameter Optimization using Grid Search and Genetic Algorithm to Improve Classification Performance," *TELKOMNIKA (Telecommunication Comput. Electron. Control.*, vol. 14, no. 4, p. 1502, 2016, doi: 10.12928/telkomnika.v14i4.3956.

[51] A. Bode, "K-Nearest Neighbor Dengan Feature Selection Menggunakan Backward Elimination Untuk Prediksi Harga Komoditi Kopi Arabika," *Ilk. J. Ilm.*, vol. 9, no. 2, pp. 188–195, 2017, doi: 10.33096/ilkom.v9i2.139.188-195.

[52] Y. Meng, N. Yang, Z. Qian, and G. Zhang, "What makes an online review more helpful: An interpretation framework using xgboost and shap values," *J. Theor. Appl. Electron. Commer. Res.*, vol. 16, no. 3, pp. 466–490, 2021, doi: 10.3390/jtaer16030029.

[53] J. Muschelli, "ROC and AUC with a Binary Predictor: a Potentially Misleading Metric," *J. Classif.*, vol. 37, no. 3, pp. 696–708, 2020, doi: 10.1007/s00357-019-09345-1.