

LAPORAN TUGAS BESAR

Implementasi Algoritma Decision Tree (ID3) untuk Prediksi Keputusan Nasabah dalam Kampanye Marketing Bank

KECERDASAN ARTIFISIAL



**Telkom
University**

Disusun Oleh:

Arkanzi Suharjanto

103012300055

Velisia Nihan Rahmawati

103012300203

Arsyad Fillah

103012300260

Kelompok 7
IF-47-08

PROGRAM STUDI S1 INFORMATIKA
FAKULTAS INFORMATIKA
TELKOM UNIVERSITY
2025

BAB I

PENDAHULUAN DAN PAPARAN DATA

1. Pendahuluan

Dalam perbankan, kampanye pemasaran merupakan strategi penting untuk meningkatkan minat nasabah terhadap produk yang ditawarkan. Salah satu permasalahan yang sering dihadapi adalah rendahnya tingkat respons nasabah terhadap penawaran produk, sehingga diperlukan metode yang mampu membantu bank dalam memprediksi keputusan nasabah secara lebih akurat.

Penelitian ini membahas implementasi algoritma Decision Tree (ID3) untuk memprediksi keputusan nasabah dalam kampanye marketing bank. Dengan model prediksi ini, bank diharapkan dapat menargetkan nasabah yang lebih potensial sehingga meningkatkan efektivitas dan efisiensi kampanye pemasaran.

Dataset yang digunakan adalah Bank Marketing Dataset yang diperoleh dari Kaggle, yang berisi data hasil kampanye pemasaran bank melalui panggilan telepon. Dataset ini mencakup berbagai atribut terkait profil nasabah dan riwayat interaksi pemasaran, dengan variabel target berupa keputusan nasabah terhadap penawaran deposito berjangka (*yes* atau *no*).

2. Paparan Data

Pada pemaparan dataset ini kami mengambil dataset dari Bank marketing dataset yang diperoleh dari platform kaggle. Dalam dataset tersebut berisi hasil dari kampanye pemasaran Bank melalui panggilan telepon kepada nasabah.

Untuk statistik dataset, terdiri dari:

- Jenis masalah: Klasifikasi
- Variabel target: *y* (*yes/no*)
- Data berisi informasi profil nasabah, riwayat kontak pemasaran, serta indikator sosial dan ekonomi
- Label target menunjukkan apakah nasabah berlangganan deposito berjangka atau tidak

Atribut	Tipe Data	Deskripsi
Id	Numerik	Identitas unik setiap data
Age	Numerik	Usia nasabah
Job	Kategorikal	Jenis pekerjaan nasabah
Marital	Kategorikal	Status pernikahan
Education	Kategorikal	Tingkat pendidikan

Default	Kategorikal	Status kredit macet
Housing	Kategorikal	Kepemilikan kredit perumahan
Loan	Kategorikal	Kepemilikan pinjaman pribadi
Contact	Kategorikal	Jenis media komunikasi
Month	Kategorikal	Bulan kontak terakhir
day_of_week	Kategorikal	Hari kontak terakhir
duration	Numerik	Durasi kontak terakhir
campaign	Numerik	Jumlah kontak selama kampanye
pdays	Numerik	Jarak hari sejak kontak sebelumnya
previous	Numerik	Jumlah kontak sebelum kampanye saat ini
poutcome	Kategorikal	Hasil kampanye sebelumnya
emp.var.rate	Numerik	Tingkat variasi tenaga kerja
cons.price.idx	Numerik	Indeks harga konsumen
cons.conf.idx	Numerik	Indeks kepercayaan konsumen
euribor3m	Numerik	Suku bunga Euribor 3 bulan
nr.employed	Numerik	Jumlah tenaga kerja

Variabel Target merupakan “y” yang memiliki tipe data kategorikal hanya tersedia pada data training, dan digunakan sebagai target klasifikasi dalam penelitian ini. Keputusan nasabah terhadap deposito berjangka (yes/no).

Beberapa atribut seperti *id* dan *duration* tidak digunakan dalam pemodelan karena tidak relevan atau tidak realistik untuk prediksi sebelum kampanye dilakukan.

BAB II

METODE DAN EKSPERIMENT

1. Metode yang Digunakan

Metode yang digunakan dalam penelitian ini adalah Decision Tree dengan algoritma ID3 (Iterative Dichotomiser 3). Algoritma ID3 membangun pohon keputusan dengan memilih atribut terbaik berdasarkan nilai Information Gain, yang dihitung dari nilai entropi setiap atribut.

Metode ini digunakan untuk menyelesaikan permasalahan klasifikasi biner, yaitu memprediksi apakah seorang nasabah akan berlangganan deposito berjangka (yes) atau tidak (no) berdasarkan data pemasaran bank.

Mengapa Decision Tree ID3 dipilih:

- Bersifat interpretable dan mudah dipahami
- Cocok untuk data kategorikal
- Menghasilkan aturan keputusan berbasis *if-then*.

2. Pengukuran Kinerja yang Digunakan (Matrix Evaluasi)

Kinerja model dievaluasi menggunakan beberapa matrix evaluasi klasifikasi, yaitu:

- **Accuracy**, untuk mengukur persentase prediksi yang benar.
- **Precision**, untuk mengukur ketepatan prediksi kelas positif.
- **Recall**, untuk mengukur kemampuan model dalam mendeteksi kelas positif.
- **F1-Score**, sebagai rata-rata harmonik antara precision dan recall.
- **Confusion Matrix**, untuk melihat distribusi prediksi benar dan salah.

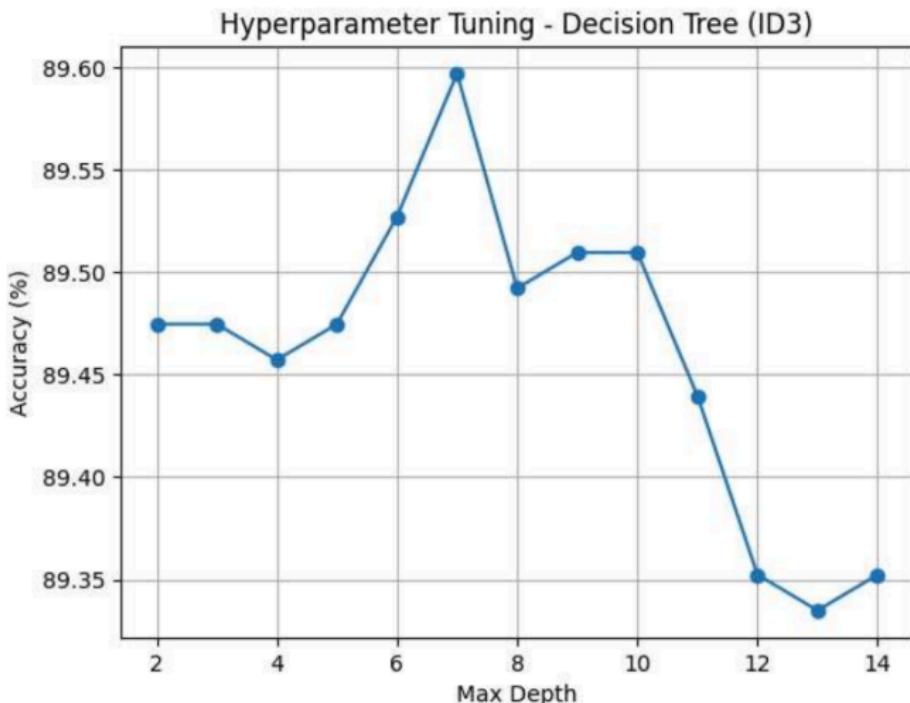
Pemilihan matrix precision, recall, dan F1-score dilakukan karena dataset memiliki ketidakseimbangan kelas, sehingga accuracy saja tidak cukup merepresentasikan kinerja model.

3. Setting Hyperparameter (Hyperparameter Tuning) yang Digunakan

Pada penelitian ini, dilakukan proses *hyperparameter tuning* untuk meningkatkan kinerja model Decision Tree dan mengurangi risiko overfitting. Hyperparameter yang digunakan dan diatur pada penelitian ini adalah **max_depth**, yaitu parameter yang menentukan kedalaman maksimum pohon keputusan yang dibentuk oleh algoritma ID3.

Pengaturan **max_depth** bertujuan untuk mengontrol kompleksitas model. Nilai **max_depth** yang terlalu besar dapat menyebabkan pohon keputusan menjadi sangat kompleks dan cenderung menyesuaikan diri secara berlebihan terhadap data training

(*overfitting*). Sebaliknya, nilai **max_depth** yang terlalu kecil dapat menyebabkan model tidak mampu menangkap pola penting dalam data (*underfitting*).



Nilai **max_depth** terbaik dipilih berdasarkan nilai accuracy tertinggi pada data validation, dengan tetap memperhatikan stabilitas performa antara data training dan validation. Pendekatan ini bertujuan untuk memperoleh model yang tidak hanya memiliki kinerja tinggi pada data training, tetapi juga mampu melakukan generalisasi dengan baik pada data yang belum pernah dilihat sebelumnya.

Melalui proses *hyperparameter tuning* ini, diharapkan model Decision Tree ID3 yang dihasilkan memiliki struktur pohon keputusan yang optimal, dengan keseimbangan antara kompleksitas model dan kemampuan prediksi.

BAB III

HASIL DAN ANALISIS

1. Kinerja dan Analisis Model (Training, Validation, dan Testing)

Pada tahap ini dilakukan analisis kinerja model Decision Tree menggunakan algoritma ID3 yang meliputi proses training, validation, dan testing. Proses ini bertujuan untuk mengevaluasi kemampuan model dalam mempelajari pola dari data, menentukan parameter optimal, serta menguji kemampuan generalisasi model terhadap data baru.

```
#Set variable for training model
top6 = ['nr.employed', 'emp.var.rate', 'pdays', 'poutcome', 'cons.conf.idx', 'eurib
X = train_sel[top6]
y = train_sel['y']
```

Setelah tahap preprocessing dan pemilihan 6 fitur terbaik menggunakan metode feature importance dari XGBoost, fitur-fitur tersebut digunakan sebagai input dalam pemodelan Decision Tree ID3. Dataset kemudian dibagi menjadi data training dan data validation dengan rasio 80% untuk training dan 20% untuk validation. Pembagian ini bertujuan agar model dapat dilatih dengan cukup data sekaligus diuji kemampuannya dalam memprediksi data yang belum pernah dilihat sebelumnya.

a. Proses Training dan Validation Model

Pada tahap training, model Decision Tree ID3 dilatih menggunakan data training dengan batas kedalaman pohon (max_depth) awal sebesar 5. Model yang telah dilatih kemudian dievaluasi menggunakan data validation untuk mengukur performa awal model. Evaluasi dilakukan menggunakan matrix accuracy dan confusion matrix guna melihat ketepatan prediksi serta distribusi kesalahan klasifikasi.

```

from sklearn.model_selection import train_test_split
X_train, X_valid, y_train, y_valid = train_test_split(
    X, y, test_size=0.2, random_state=42
)

from sklearn.tree import DecisionTreeClassifier

model = DecisionTreeClassifier(
    criterion='entropy',
    max_depth=5,
    min_samples_split=20,
    min_samples_leaf=10,
    random_state=42
)

model.fit(X_train, y_train)

```

▼ DecisionTreeClassifier ? 1

► Parameters

```

from sklearn.metrics import accuracy_score, classification_report, confusion_matrix
y_pred = model.predict(X_valid)

print("Accuracy:", accuracy_score(y_valid, y_pred))
print("\nClassification Report:\n", classification_report(y_valid, y_pred))

```

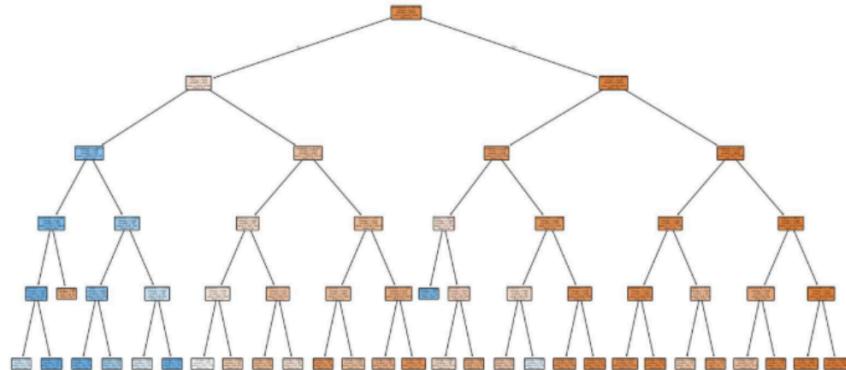
Accuracy: 0.8936987257811136

Classification Report:				
	precision	recall	f1-score	support
0	0.90	0.99	0.94	5060
1	0.67	0.18	0.28	669
accuracy			0.89	5729
macro avg	0.79	0.58	0.61	5729
weighted avg	0.87	0.89	0.87	5729

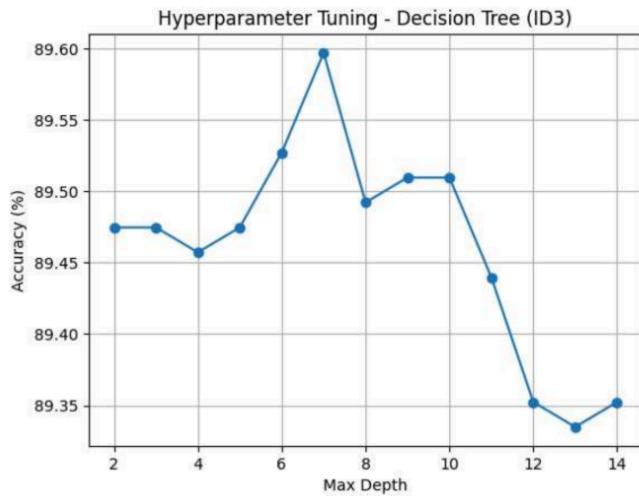
Hasil evaluasi awal menunjukkan bahwa model mampu memprediksi kelas *no* dengan cukup baik, namun masih memiliki keterbatasan dalam mengenali kelas *yes*. Hal ini disebabkan oleh ketidakseimbangan distribusi kelas pada dataset, di mana jumlah data nasabah yang tidak berlangganan deposito jauh lebih banyak dibandingkan nasabah yang berlangganan.



TREE WITH 5 DEPTH



Untuk meningkatkan performa model dan mengurangi risiko overfitting maupun underfitting, dilakukan proses hyperparameter tuning terhadap parameter `max_depth`. Pengujian dilakukan dengan mencoba beberapa nilai `max_depth`, mulai dari 2 hingga 14. Setiap model yang dihasilkan kemudian dievaluasi menggunakan nilai accuracy pada data validation.



Berdasarkan hasil pengujian, diperoleh nilai `max_depth = 7` sebagai parameter terbaik karena menghasilkan nilai accuracy tertinggi pada data validation dibandingkan kedalaman lainnya. Nilai ini dianggap mampu memberikan keseimbangan antara kompleksitas model dan kemampuan generalisasi.

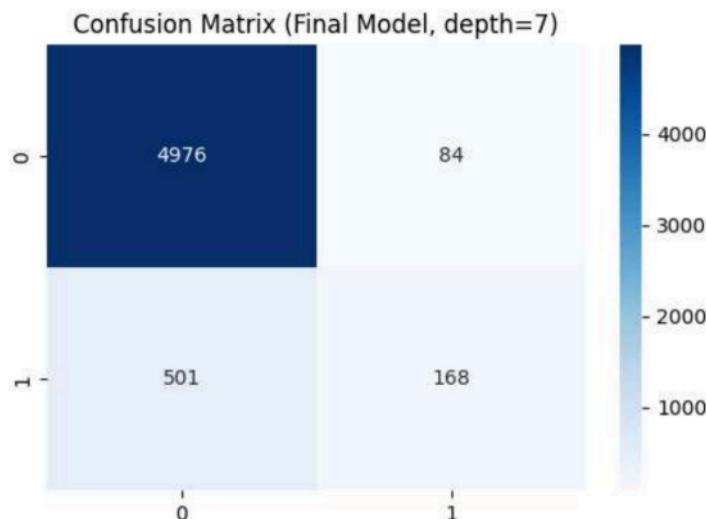
b. Model Final dan Analisis Performa

Setelah diperoleh parameter optimal, model Decision Tree ID3 kemudian dilatih ulang menggunakan nilai `max_depth = 7` untuk membentuk model final. Model ini selanjutnya digunakan untuk melakukan prediksi pada data validation guna mengevaluasi performa akhir sebelum diterapkan pada data testing.

Berdasarkan confusion matrix hasil model final, terlihat adanya peningkatan kemampuan model dalam memprediksi kelas *yes* dibandingkan model awal. Meskipun demikian, model masih menunjukkan kecenderungan lebih baik dalam memprediksi kelas *no*. Kondisi ini dipengaruhi oleh distribusi data yang tidak seimbang, di mana kelas *yes* memiliki jumlah data yang jauh lebih sedikit dibandingkan kelas *no*.

Hasil ini menunjukkan bahwa proses hyperparameter tuning berhasil meningkatkan performa model secara keseluruhan, terutama dalam mengklasifikasikan kelas minoritas.

	precision	recall	f1-score	support
0	0.91	0.98	0.94	5060
1	0.67	0.25	0.36	669
accuracy			0.90	5729
macro avg	0.79	0.62	0.65	5729
weighted avg	0.88	0.90	0.88	5729



Setelah melakukan validasi dengan *Confusion matrix*, model mengalami *imbalance*. dimana model memprediksi dataset dengan perbandingan yang tidak seimbang antara prediksi yes dan no. Model lebih banyak memprediksi no dibandingkan yes. Hal ini terlihat dari nilai **recall kelas yes yang hanya sebesar 25%**, sementara **recall kelas no mencapai 98%**. Kondisi ini menunjukkan bahwa meskipun model memiliki nilai akurasi yang cukup tinggi, kemampuan model dalam mendeteksi kelas minoritas (yes) masih rendah.

Ketidakseimbangan ini terjadi karena distribusi data yang tidak seimbang, di mana jumlah data dengan label *no* jauh lebih dominan dibandingkan *yes*. Akibatnya, model lebih fokus mempelajari pola kelas mayoritas dan kurang sensitif terhadap kelas minoritas. Oleh karena itu, untuk mengurangi jumlah *false negative* dan meningkatkan kemampuan model dalam mendeteksi kelas *yes*, dilakukan optimisasi lanjutan menggunakan **Algoritma Genetika (Genetic Algorithm)** melalui penyesuaian threshold keputusan model.

```
==== FINAL EVAL (GA Threshold) ====
Accuracy: 87.01%
Precision(yes): 45.17%
Recall(yes): 52.47%
F1(yes): 48.55%
Confusion Matrix:
[[4634 426]
 [ 318 351]]
```



Berdasarkan hasil evaluasi setelah dilakukan optimisasi threshold menggunakan Algoritma Genetika (GA), diperoleh peningkatan yang signifikan pada kemampuan model dalam mendekripsi kelas minoritas (yes). Nilai recall kelas yes meningkat dari 25% menjadi 52,47%, yang menunjukkan bahwa model menjadi lebih sensitif dalam mengenali nasabah yang berpotensi menerima penawaran deposito. Selain itu, nilai F1-score kelas yes juga meningkat menjadi 48,55%, menandakan keseimbangan yang lebih baik antara precision dan recall.

Meskipun terjadi penurunan nilai akurasi menjadi 87,01%, penurunan ini merupakan konsekuensi dari trade-off yang umum terjadi pada dataset yang bersifat imbalanced. Berdasarkan confusion matrix, terlihat bahwa jumlah false negative berkurang secara signifikan, sementara false positive mengalami peningkatan. Trade-off ini dapat diterima karena tujuan utama optimisasi adalah meningkatkan deteksi kelas minoritas, yang lebih bernilai secara praktis dalam konteks kampanye pemasaran bank. Dengan demikian, penerapan Algoritma Genetika terbukti efektif sebagai metode optimisasi lanjutan untuk menangani permasalahan imbalance data.

c. Analisis Tahap Testing

Tahap testing dilakukan menggunakan data uji yang tidak digunakan pada proses training maupun validation. Pengujian ini bertujuan untuk mengevaluasi kemampuan generalisasi model terhadap data baru yang merepresentasikan kondisi nyata.

Berdasarkan hasil prediksi model final terhadap data testing, diperoleh distribusi kelas sebagai berikut:

- Kelas *no*: 11.976 data
- Kelas *yes*: 567 data

```
submission["y_pred"].value_counts()  
  
y_pred  
no      11976  
yes      567  
Name: count, dtype: int64
```

Hasil tersebut menunjukkan bahwa mayoritas nasabah pada data uji diprediksi tidak menerima penawaran deposito berjangka, sedangkan hanya sebagian kecil yang diprediksi menerima penawaran. Pola ini konsisten dengan distribusi data training yang tidak seimbang.

Meskipun demikian, model mampu diterapkan pada data baru dan menghasilkan pola prediksi yang realistik sesuai dengan karakteristik dataset. Hal ini menunjukkan bahwa model Decision Tree ID3 yang dibangun memiliki kemampuan generalisasi yang cukup baik dan dapat digunakan sebagai alat bantu pengambilan keputusan dalam kampanye pemasaran bank.

```
# Asumsi test_sel sudah dibersihin dan pakai 6 fitur yang sama
X_test_final = test_sel[top6]

test_pred = model_final.predict(X_test_final)
test_pred_label = pd.Series(test_pred).map({0: "no", 1: "yes"})

# Gabung dengan id dan simpan
submission = test.copy()
submission["y_pred"] = test_pred_label
submission.to_csv("hasil_prediksi_test.csv", index=False)
submission.head()
```

	id	age	job	marital	education	default	housing	loan	contact
0	3	21	student	single	unknown	no	no	no	telephone
1	4	42	blue-collar	married	basic.6y	no	no	no	cellular
2	7	42	technician	married	professional.course	no	yes	no	cellular
3	9	33	services	married	professional.course	unknown	yes	no	cellular
4	13	29	unemployed	divorced	high.school	no	no	no	cellular

5 rows × 21 columns

Setelah dilakukan proses **optimisasi threshold menggunakan Algoritma Genetika (GA)** untuk menangani permasalahan *imbalanced data*, tahap pengujian dilakukan kembali menggunakan data **test**. Pada tahap ini, model tidak lagi menggunakan threshold default (0.5), melainkan threshold optimal hasil optimisasi GA untuk menentukan label prediksi *yes* atau *no*. Proses prediksi dilakukan berdasarkan nilai probabilitas kelas *yes* yang dihasilkan oleh model.i.

```
proba_test = model_ga.predict_proba(test_sel[top6])[:, 1]
test_pred_ga = (proba_test >= best_t).astype(int)

submission = test.copy()
submission["y_pred"] = pd.Series(test_pred_ga).map({0:"no", 1:"yes"})
submission.to_csv("hasil_prediksi_test_GA.csv", index=False)
submission.head()
```

	id	age	job	marital	education	default	housing	loan	contact
0	3	21	student	single	unknown	no	no	no	telephone
1	4	42	blue-collar	married	basic.6y	no	no	no	cellular
2	7	42	technician	married	professional.course	no	yes	no	cellular
3	9	33	services	married	professional.course	unknown	yes	no	cellular
4	13	29	unemployed	divorced	high.school	no	no	no	cellular

5 rows × 21 columns

```
submission["y_pred"].value_counts()
```

```
y_pred  
no      10844  
yes     1699  
Name: count, dtype: int64
```

Hasil prediksi kemudian dikonversi ke dalam bentuk label kategorikal dan disimpan ke dalam file **hasil_prediksi_test_GA.csv**. Berdasarkan distribusi hasil prediksi, diperoleh **10.844 data diprediksi sebagai “no”** dan **1.699 data diprediksi sebagai “yes”**. Dibandingkan dengan hasil prediksi sebelum optimisasi, jumlah prediksi *yes* mengalami peningkatan yang cukup signifikan. Hal ini menunjukkan bahwa optimisasi threshold menggunakan Algoritma Genetika berhasil membuat model lebih sensitif dalam mengidentifikasi nasabah yang berpotensi menerima penawaran deposito, meskipun data awal bersifat tidak seimbang.

BAB IV KESIMPULAN

Algoritma Decision Tree (ID3) mampu digunakan untuk memprediksi keputusan nasabah dalam kampanye pemasaran bank dengan kinerja yang cukup baik. Model dibangun menggunakan 6 fitur terbaik hasil seleksi XGBoost dan dievaluasi melalui tahap training, validation, dan testing.

Hasil percobaan menunjukkan bahwa hyperparameter kedalaman pohon (`max_depth`) berpengaruh terhadap performa model. Setelah dilakukan hyperparameter tuning, diperoleh kinerja terbaik pada `max_depth = 7`, yang menghasilkan akurasi tertinggi pada data validasi dibandingkan konfigurasi lainnya. Berdasarkan confusion matrix, model setelah tuning menunjukkan peningkatan kemampuan dalam mengklasifikasikan kelas *yes*, meskipun masih lebih baik dalam memprediksi kelas *no* akibat ketidakseimbangan data.

Pada tahap testing, selain melakukan prediksi menggunakan model ID3 final, dilakukan pula optimisasi tambahan menggunakan Algoritma Genetika (Genetic Algorithm) untuk menentukan threshold keputusan yang optimal. Optimisasi ini bertujuan untuk meningkatkan sensitivitas model terhadap kelas minoritas (*yes*). Hasil pengujian menunjukkan bahwa penerapan optimisasi threshold dengan Algoritma Genetika mampu meningkatkan kemampuan model dalam mendeteksi kelas *yes*, meskipun terjadi trade-off terhadap akurasi keseluruhan.

Secara keseluruhan, Decision Tree ID3 dengan `max_depth = 7` yang dikombinasikan dengan optimisasi threshold menggunakan Algoritma Genetika dapat

digunakan sebagai alat bantu pengambilan keputusan dalam kampanye pemasaran bank, terutama dalam mengidentifikasi nasabah yang berpotensi menerima penawaran, meskipun data bersifat tidak seimbang.