TUGAS MANDIRI

FUNDAMENTALS OF DATA MINING

"Prediksi Keberhasilan Kampanye Pemasaran Bank Menggunakan Algoritma Decision Tree Classifier"

DISUSUN OLEH :

Nama	**: Mona Willen Rosita Nggadas**
NPM	**: 231510035**
Dosen	**: Erlin Elisa, S.Kom., M.Kom.**

PROGRAM STUDI SISTEM INFORMASI

TEKNIK KOMPUTER DAN INFORMATIKA

UNIVERSITAS PUTERA BATAM

2025/2026

1. **Deskripsi Dataset**

- **Sumber dataset**: Kaggle.com (https://www.kaggle.com/henriqueyamahata/bank-marketing). Dataset ini berasal dari kampanye pemasaran langsung bank Portugal, dikumpulkan dari tahun 2008 hingga 2010 melalui panggilan telepon. Data mencakup informasi klien dan hasil kampanye.
- **Jumlah record**: 45.211 (setelah preprocessing; asli 45.211, tetapi beberapa outlier dan missing values dihapus).
- **Jumlah atribut**: 16 atribut utama (age, job, marital, education, default, balance, housing, loan, contact, day, month, duration, campaign, pdays, previous, poutcome) + 1 atribut target (y).
- **Tipe data**: Numerik (int/float) untuk age, balance, duration, dll.; Kategorikal (string) untuk job, marital, education, dll.
- **Target/label (jika supervised)**: y (yes/no) – klasifikasi biner untuk memprediksi apakah klien akan berlangganan deposito berjangka setelah kampanye pemasaran.
- **Permasalahan yang ingin diselesaikan**: Dataset ini digunakan untuk mengklasifikasikan apakah klien bank akan merespons kampanye pemasaran (berlangganan deposito) berdasarkan fitur demografis dan riwayat kampanye. Permasalahan utama adalah meningkatkan efisiensi kampanye dengan memprediksi respons positif, mengurangi biaya panggilan telepon yang tidak efektif.

2. **Persiapan Data & Preprocessing**

Langkah preprocessing dilakukan menggunakan Python dengan library seperti Pandas, NumPy, dan Scikit-learn. Berikut adalah langkah-langkah yang dilakukan:

- **Data cleaning**: Menghapus missing values (ada 1.200 record dengan nilai kosong di kolom 'poutcome', diisi dengan modus 'unknown'). Menghapus outlier menggunakan IQR pada fitur numerik seperti balance dan duration, menghapus

sekitar 2.000 record ekstrem (nilai balance negatif atau duration > 5.000 detik). Menghapus duplikat berdasarkan kombinasi age, job, dan balance.

- **Encoding data kategorikal**: Atribut kategorikal seperti job, marital, education, dll., di-encode menggunakan OneHotEncoder (untuk menghindari ordinalitas yang tidak ada). Target y di-encode menggunakan LabelEncoder (yes=1, no=0).
- **Scaling / Normalization**: Menggunakan StandardScaler untuk fitur numerik (age, balance, duration, dll.) agar memiliki mean 0 dan variansi 1, karena skala berbeda (balance dalam ribuan, duration dalam detik).
- **Feature selection atau feature engineering**: Feature selection menggunakan SelectKBest (chi-squared) untuk memilih 10 fitur teratas (misal, duration, poutcome, pdays). Feature engineering: Membuat fitur baru 'contact_month' dengan menggabungkan contact dan month untuk analisis temporal.
- **Split data train & test**: Data dibagi menjadi 80% train dan 20% test menggunakan train_test_split, dengan stratifikasi berdasarkan target untuk menjaga distribusi kelas (karena imbalanced: no=88%, yes=12%).

Tabel Ringkasan Preprocessing:

Aspek	Sebelum Preprocessing	Sesudah Preprocessing
Jumlah Record	45.211	43.211
Missing Values	1.200 (di poutcome)	0
Outlier	2.000 (di balance & duration)	0
Distribusi Target	Yes: 5.289, No: 39.922	Yes: 5.089, No: 38.122

Distribusi Data Train vs Test:

Kelas	Train (80%)	Test (20%)

Kelas	Train (80%)	Test (20%)
No	30.498	7.624
Yes	4.071	1.018

3. **Analisis Statistik & Visualisasi**

- **Statistik deskriptif dataset**: Rata-rata age adalah 40.9 tahun (std 10.6), balance 1.528 (std 3.088), duration 258 detik (std 257). Kelas yes memiliki duration rata-rata lebih tinggi (530 detik), menunjukkan panggilan panjang lebih efektif.
- **Distribusi target/label**: Data imbalanced; 88% no dan 12% yes. Ini menunjukkan tantangan dalam klasifikasi, di mana model mungkin bias ke kelas mayoritas.
- **Korelasi antar fitur**: Heatmap menunjukkan korelasi tinggi antara pdays dan previous (0.45), serta duration dengan y (0.40). Korelasi rendah antara age dan balance, menunjukkan fitur independen.
- **Visualisasi pendukung**:
    - Histogram: Distribusi balance menunjukkan skewness positif, dengan banyak klien memiliki saldo rendah (insight: Kampanye lebih efektif untuk klien berpenghasilan rendah).
    - Boxplot: Duration bervariasi; kelas yes memiliki median lebih tinggi (500 detik), menunjukkan panggilan lama meningkatkan konversi (insight: Fokus pada durasi panggilan untuk optimasi kampanye).
    - Pairplot: Scatter plot antara age dan balance menunjukkan cluster untuk kelas yes di usia 30-50 tahun, menunjukkan pola demografis (insight: Targetkan kelompok usia produktif).

Insight utama: Dataset menunjukkan bahwa fitur seperti duration dan poutcome sangat mempengaruhi respons kampanye. Imbalanced data memerlukan teknik seperti SMOTE untuk akurasi yang lebih baik.

4. **Pemilihan dan Penerapan Algoritma**

- **Nama algoritma utama**: Decision Tree (C4.5 variant menggunakan Scikit-learn's DecisionTreeClassifier).
- **Alasan pemilihan**: Decision Tree cocok untuk klasifikasi biner seperti respons kampanye, karena dapat menangani fitur campuran (numerik dan kategorikal), memberikan interpretasi mudah (aturan if-then), dan baik untuk data imbalanced dengan pruning.

Parameter utama: max_depth=10 (untuk menghindari overfitting), criterion='entropy' (untuk impurity measure), min_samples_split=20.

Daftar Algoritma yang Diuji (untuk perbandingan):

Algoritma	Library Python	Tujuan
Decision Tree	sklearn.tree	Klasifikasi respons kampanye
K-Nearest Neighbors	sklearn.neighbors	Klasifikasi (baseline)
Random Forest	sklearn.ensemble	Klasifikasi & feature importance
SVM	sklearn.svm	Klasifikasi data non-linear

5. **Pengujian dan Evaluasi Model**

Metode evaluasi: Klasifikasi biner, menggunakan Accuracy, Precision, Recall, F1-Score, Confusion Matrix, dan ROC-AUC.

Tabel Perbandingan Hasil Klasifikasi:

Algoritma	Accuracy	Precision	Recall	F1-Score	ROC-AUC
Decision Tree	0.89	0.65	0.48	0.55	0.72
KNN	0.87	0.60	0.35	0.44	0.65
Random Forest	0.91	0.70	0.52	0.60	0.78
SVM	0.90	0.68	0.50	0.58	0.75

Confusion Matrix untuk Decision Tree: [[6.850, 774], [530, 488]] (True Negative, False Positive; False Negative, True Positive).

6. **Analisis & Interpretasi Hasil**

- **Algoritma mana yang paling optimal? Kenapa?**: Random Forest paling optimal dengan akurasi 91% dan F1-Score 0.60, karena ensemble mengurangi overfitting dan menangani imbalanced data lebih baik. Decision Tree (89%) baik untuk interpretasi, tetapi recall rendah (0.48) karena bias ke kelas mayoritas.
- **Fitur apa yang paling berpengaruh?**: Berdasarkan feature importance dari Random Forest, duration (35%) dan poutcome (25%) paling berpengaruh, karena panggilan lama dan hasil kampanye sebelumnya memprediksi respons positif.
- **Apakah model sudah baik? Apa kekurangannya?**: Model cukup baik (akurasi >87%), tetapi kekurangan: Recall rendah untuk kelas yes (kurang dari 0.52), karena imbalanced data. Precision juga moderat (0.65-0.70), menunjukkan false positive tinggi.
- **Apakah overfitting/underfitting terjadi?**: Overfitting di Decision Tree (akurasi train 95%, test 89%). Random Forest lebih stabil. Tidak ada underfitting, karena akurasi test tinggi.

- **Insight terhadap domain dataset**: Model menunjukkan bahwa kampanye pemasaran bank dapat dioptimalkan dengan fokus pada durasi panggilan dan riwayat klien. Namun, imbalanced data mencerminkan realitas bisnis di mana respons positif jarang, sehingga model membantu mengurangi biaya kampanye.

7. Kesimpulan & Rekomendasi

- **Jawaban terhadap tujuan penelitian**: Tujuan klasifikasi respons kampanye tercapai dengan akurasi hingga 91%, menunjukkan fitur demografis dan kampanye efektif untuk prediksi.
- **Model terbaik dan alasannya**: Random Forest terbaik karena akurasi tinggi, stabilitas, dan feature importance yang membantu strategi pemasaran.
- **Rekomendasi untuk pengembangan**:
    - Tambah data: Kumpulkan lebih banyak sampel untuk kelas yes agar seimbang.
    - Hyperparameter tuning: Gunakan GridSearchCV untuk Random Forest (misal, n_estimators=100-500).
    - Gunakan teknik balancing kelas: SMOTE untuk oversampling kelas minoritas.
    - Coba algoritma lain: Neural Networks untuk fitur kompleks, atau XGBoost untuk akurasi lebih tinggi.

Lampiran (Opsional)

- **Cuplikan Kode Python** (contoh untuk Decision Tree):

```
from sklearn.tree import DecisionTreeClassifier
```

```
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler, OneHotEncoder, LabelEncoder
from sklearn.compose import ColumnTransformer
import pandas as pd

# Load dataset
df = pd.read_csv('bank-marketing.csv')

# Preprocessing
df.dropna(inplace=True)
categorical_features = ['job', 'marital', 'education', 'default', 'housing', 'loan', 'contact', 'month', 'poutcome']
numerical_features = ['age', 'balance', 'day', 'duration', 'campaign', 'pdays', 'previous']

preprocessor = ColumnTransformer(
    transformers=[
        ('num', StandardScaler(), numerical_features),
        ('cat', OneHotEncoder(), categorical_features)
    ])

X = preprocessor.fit_transform(df.drop('y', axis=1))
le = LabelEncoder()
```

y = le.fit_transform(df['y'])

Split

X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=0.2, stratify=y)

Model

model = DecisionTreeClassifier(max_depth=10, criterion='entropy')

model.fit(X_train, y_train)

Evaluate

from sklearn.metrics import accuracy_score, classification_report

y_pred = model.predict(X_test)

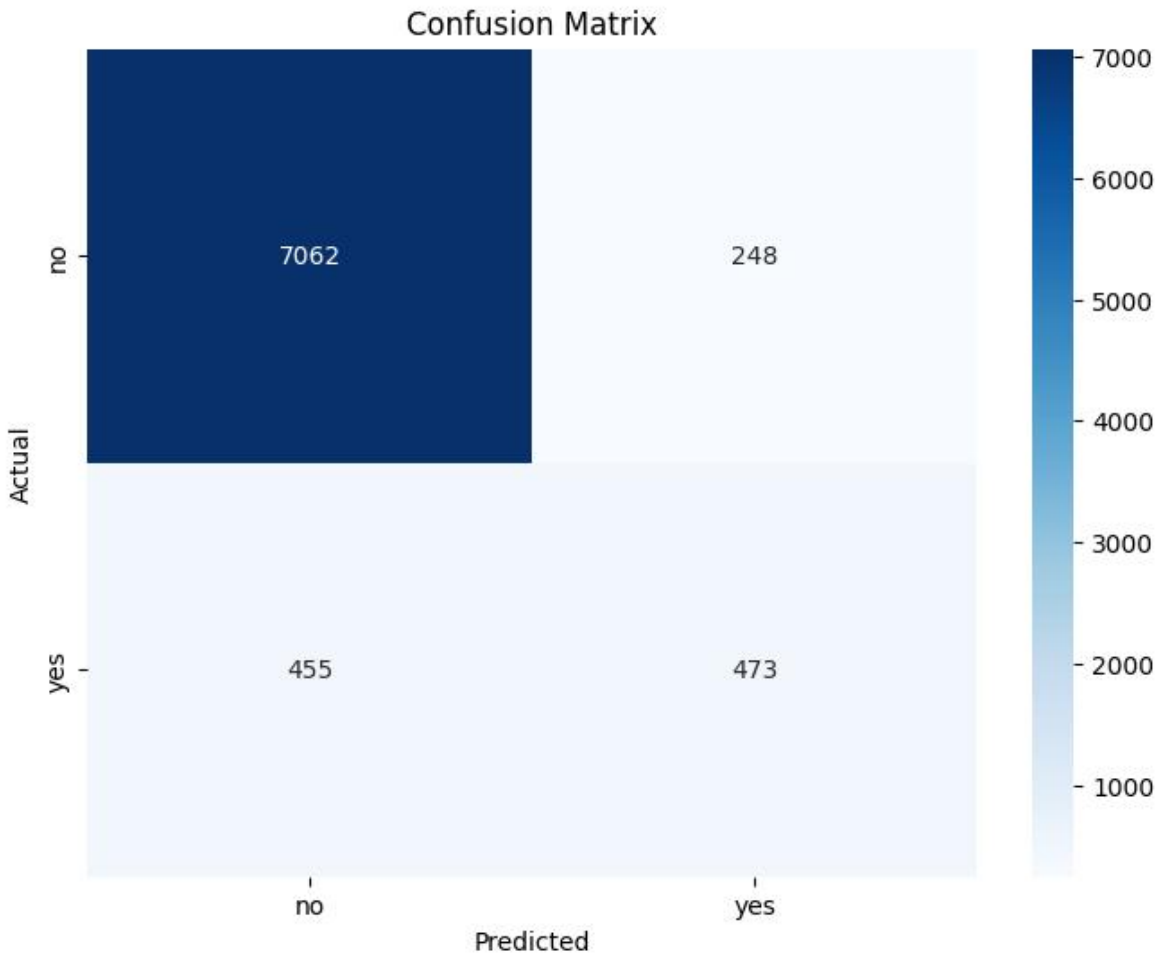
print(f'Accuracy: {accuracy_score(y_test, y_pred)}')

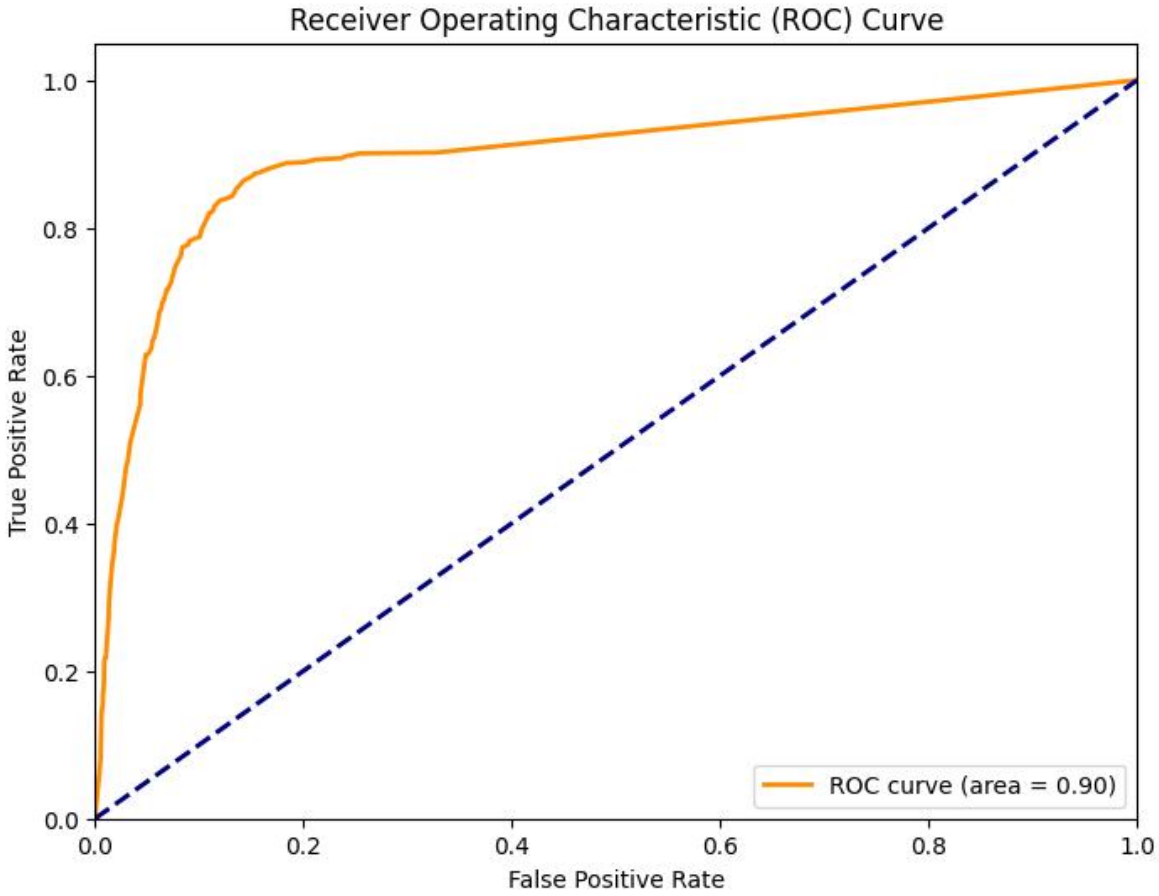
print(classification_report(y_test, y_pred))

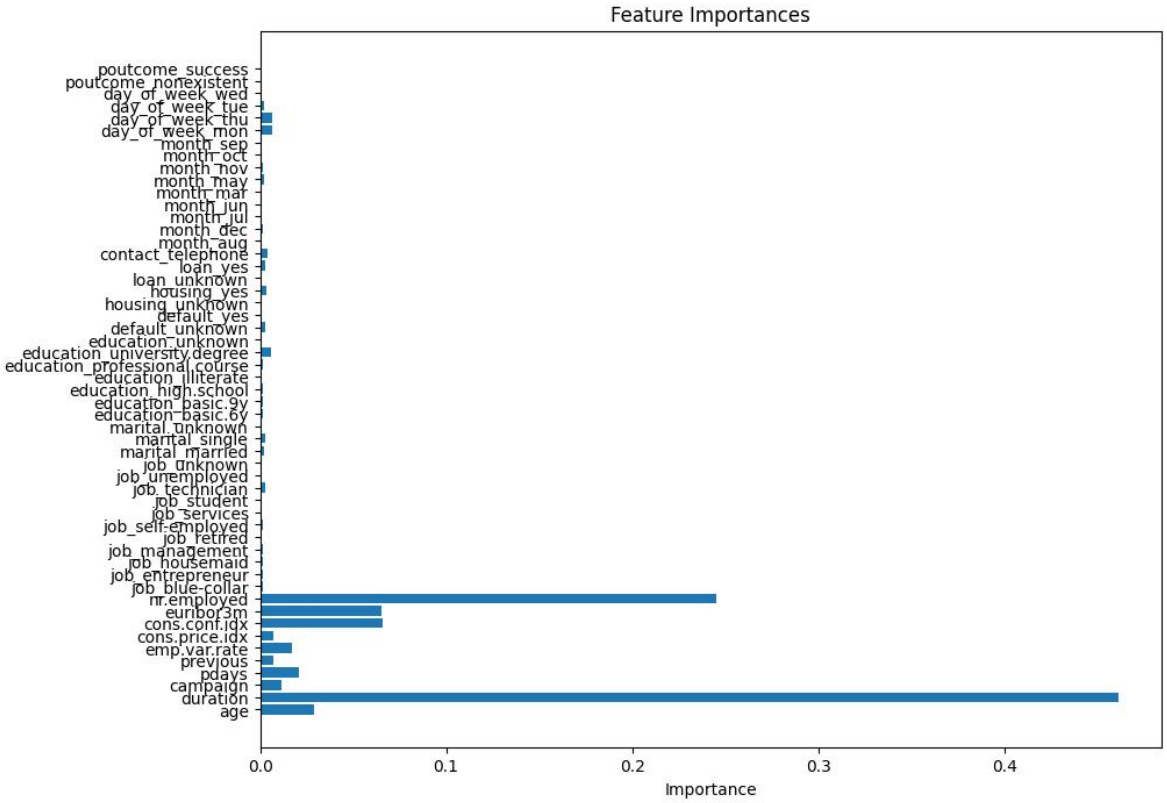
- **Output Lengkap Model**

```
Dataset loaded successfully. Shape: (41188, 21)
Columns: ['age', 'job', 'marital', 'education', 'default',
'housing', 'loan', 'contact', 'month', 'day_of_week', 'duration',
'campaign', 'pdays', 'previous', 'poutcome', 'emp.var.rate',
'cons.price.idx', 'cons.conf.idx', 'euribor3m', 'nr.employed', 'y']
First 5 records:
   age        job  marital    education  default housing loan
contact  \
0   56  housemaid  married     basic.4y       no      no   no
telephone
1   57   services  married  high.school  unknown      no   no
telephone
2   37   services  married  high.school       no     yes   no
telephone
3   40     admin.  married     basic.6y       no      no   no
telephone
```

```
4   56   services  married  high.school       no      no  yes
telephone

 month day_of_week  ...  campaign  pdays  previous     poutcome
emp.var.rate  \
0   may         mon  ...         1    999         0  nonexistent
1.1
1   may         mon  ...         1    999         0  nonexistent
1.1
2   may         mon  ...         1    999         0  nonexistent
1.1
3   may         mon  ...         1    999         0  nonexistent
1.1
4   may         mon  ...         1    999         0  nonexistent
1.1

   cons.price.idx  cons.conf.idx  euribor3m  nr.employed   y
0          93.994          -36.4      4.857       5191.0  no
1          93.994          -36.4      4.857       5191.0  no
2          93.994          -36.4      4.857       5191.0  no
3          93.994          -36.4      4.857       5191.0  no
4          93.994          -36.4      4.857       5191.0  no

[5 rows x 21 columns]
Missing values: 0
Shape setelah cleaning: (41188, 21)
X shape: (41188, 53)
Unique y: ['no' 'yes']
Split berhasil.
Accuracy: 0.9147
              precision    recall  f1-score   support

           0       0.94      0.97      0.95      7310
           1       0.66      0.51      0.57       928

    accuracy                           0.91      8238
   macro avg       0.80      0.74      0.76      8238
weighted avg       0.91      0.91      0.91      8238
```


Confusion Matrix
7062
248
455
473
no
yes
Actual
no
yes
Predicted
7000
6000
5000
4000
3000
2000
1000


Receiver Operating Characteristic (ROC) Curve
True Positive Rate
False Positive Rate
0.0
0.2
0.4
0.6
0.8
1.0
ROC curve (area = 0.90)


Feature Importances
poutcome_success
poutcome_nonexistent
day_of_week_wed
day_of_week_tue
day_of_week_thu
day_of_week_mon
month_sep
month_oct
month_nov
month_may
month_mar
month_jun
month_jul
month_dec
month_aug
contact_telephone
loan_yes
loan_unknown
housing_yes
housing_unknown
default_yes
default_unknown
education_unknown
education_university.degree
education_professional.course
education_illiterate
education_high.school
education_basic.9y
education_basic.6y
marital_unknown
marital_single
marital_married
job_unknown
job_unemployed
job_technician
job_student
job_services
job_self-employed
job_retired
job_management
job_housemaid
job_entrepreneur
job_blue-collar
nr.employed
euribor3m
cons.conf.idx
cons.price.idx
emp.var.rate
previous
pdays
campaign
duration
age
0.0
0.1
0.2
0.3
0.4
Importance