

LAMPIRAN



Lampiran 1. Surat Izin Penelitian



KEMENTERIAN PENDIDIKAN TINGGI, SAINS DAN TEKNOLOGI
UNIVERSITAS PENDIDIKAN GANESHA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
JURUSAN MATEMATIKA

Alamat : Jalan Udayana Singaraja-Bali

Telepon (0362) 25072 Fax. (0362) 25335 Pos 81116 Laman: www.undiksha.ac.id

Singaraja, 16 Maret 2026

Nomor : 235/UN48.9.5/PT.01.04/2026
 Lampiran : -
 Perihal : Permohonan Izin Observasi

Kepada Yth. Bapak/Ibu Kepala SMK Negeri 1 Sukasada
 di tempat

Dengan hormat,

Sehubungan dalam rangka penyusunan proposal skripsi mahasiswa Jurusan Matematika, kami mohon kepada Bapak/Ibu berkenan untuk memberikan izin untuk melaksanakan observasi kepada mahasiswa berikut.

Nama : Dwita Ega Lestari
 NIM : 2213101005
 Program Studi : S1 Matematika
 Judul Penelitian : Perbandingan Random Forest dan SVM dalam Memprediksi Pemilihan Jurusan Siswa SMK

Demikian permohonan kami, atas perhatian Bapak/Ibu kami sampaikan terima kasih.

Mengetahui,
 Ketua Jurusan Matematika



I Putu Wisna Ariawan
 NIP 196805191993031001

Lampiran 2. Surat Surat Izin Balasan Penelitian



Sukasada, 17 Maret 2026

Nomor : B.10.400.3/2125/SMKN1SKSD/DIKPORA
 Sifat : Biasa
 Lampiran : -
 Hal : Balasan Izin Observasi

Yth. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam
 Universitas Pendidikan Ganesha
 Cq. Ketua Jurusan Matematika
 di Tempat

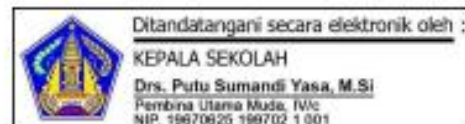
Dengan hormat,

Menindaklanjuti surat dari Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Pendidikan Ganesha, Nomor: 235/UN48.9.5/PT.01.04/2026 tanggal 16 Maret 2026 perihal Permohonan Izin Observasi, dapat kami sampaikan bahwa mahasiswa berikut:

Nama : Dwita Ega Lestari
 NIM : 2213101005
 Program Studi : S1 Matematika
 Judul Penelitian : Perbandingan Random Forest dan SVM dalam Memprediksi
 Pemilihan Jurusan Siswa SMK

diberikan ijin untuk melakukan Observasi di SMK Negeri 1 Sukasada.

Demikian surat ini disampaikan, atas perhatian dan kerjasamanya kami ucapkan terima kasih.



Dokumen ini telah ditandatangani secara elektronik (TTE).
 Scan QR Code untuk informasi TTE.
 Upload file pada <https://tts.kemendikgo.id/verifyPDF> untuk cek keaslian file.



Lampiran 3. *Data Training*

No	Jenis Kelamin	Bahasa Indonesia	Matematika	IPA	IPS	Bahasa Inggris	Jurusan
1	Laki-Laki	89.0	83.0	88.4	72.6	81.25	TSM
2	Laki-Laki	76.17	84.0	93.0	83.1	70.2	TSM
3	Laki-Laki	79.0	81.0	88.7	84.0	68.7	TSM
4	Laki-Laki	80.9	85.7	78.87	80.78	80.0	TSM
5	Laki-Laki	75.2	93.0	81.8	74.0	74.6	TSM
6	Laki-Laki	70.0	84.0	81.0	75.5	75.5	TSM
8	Laki-Laki	73.4	89.8	84.3	90.0	71.6	TSM
9	Laki-Laki	70.84	81.4	91.31	75.7	66.0	TSM
⋮							
351	Laki-Laki	77.66	80.0	89.32	73.0	76.1	TSM
352	Laki-Laki	65.0	72.0	84.4	69.83	70.9	TB
353	Laki-Laki	79.23	79.0	80.0	66.0	77.9	TB
354	Perempuan	66.1	73.1	79.87	67.82	66.29	TB
355	Laki-Laki	79.0	86.58	88.6	67.0	87.19	TB
357	Perempuan	74.7	85.0	90.34	79.02	82.41	TB
358	Perempuan	71.0	87.35	85.0	67.13	74.7	TB
359	Laki-Laki	66.0	67.0	94.0	85.0	88.0	TB
⋮							
641	Perempuan	84.0	80.23	76.56	76.0	77.1	TB
642	Perempuan	77.18	71.14	90.0	82.99	73.52	PH
643	Laki-Laki	82.0	80.0	70.4	82.2	78.6	PH
644	Laki-Laki	88.0	82.6	89.43	84.0	74.4	PH
645	Laki-Laki	75.98	81.5	80.9	79.97	77.0	PH
647	Perempuan	84.0	84.4	70.0	90.0	72.0	PH
648	Laki-Laki	87.67	71.8	87.41	82.0	85.0	PH
649	Perempuan	90.0	87.76	86.92	76.15	88.0	PH
650	Perempuan	80.7	76.0	80.3	87.04	77.3	PH

Lampiran 4. *Data testing*

No	Jenis Kelamin	Bahasa Indonesia	Matematika	IPA	IPS	Bahasa Inggris	Jurusan
1	Laki- Laki	83.4	71.3	88.0	71.33	93.0	PH
2	Perempuan	79.83	71.5	78.0	84.11	75.6	PH
3	Laki- Laki	72.0	91.6	92.6	67.2	79.9	TSM
4	Perempuan	84.87	77.63	86.31	89.0	85.3	PH
5	Laki- Laki	74.12	78.9	94.0	72.92	76.57	TB
6	Perempuan	74.0	84.53	79.2	82.27	76.74	PH
8	Laki- Laki	73.9	73.0	71.5	70.5	82.0	PH
9	Perempuan	79.2	90.96	87.83	77.0	94.7	AKL
10	Perempuan	86.0	79.34	88.6	73.0	83.5	PH
⋮							
80	Laki- Laki	79.0	88.0	90.0	72.83	77.04	TB
81	Laki- Laki	72.6	66.0	94.0	78.0	84.67	TB
82	Perempuan	78.73	90.11	80.36	65.85	85.92	AKL

83	Perempuan	93.0	69.47	84.89	75.0	91.13	DKV
84	Laki- Laki	75.61	72.13	84.44	75.9	72.0	PH
85	Laki- Laki	73.9	76.3	86.0	76.7	85.5	PH
86	Laki- Laki	94.9	74.5	84.8	71.0	95.0	DKV
87	Perempuan	80.23	78.01	78.0	86.7	79.0	PH
88	Laki- Laki	85.71	80.0	75.0	77.9	84.92	PH
89	Perempuan	83.4	71.47	72.74	76.67	83.83	PH
:							
154	Laki- Laki	84.2	88.9	90.0	87.33	76.47	PH
155	Laki- Laki	83.11	89.0	71.0	74.53	78.0	TSM
156	Laki- Laki	80.9	79.97	84.0	72.0	87.53	PH
157	Perempuan	69.0	73.0	84.0	72.5	82.8	DKV
158	Laki- Laki	82.09	76.0	65.0	82.9	94.71	TB
159	Laki- Laki	87.04	78.6	88.6	83.1	91.1	DKV
160	Laki- Laki	77.0	67.4	81.0	84.0	66.7	PH
161	Perempuan	89.3	68.23	75.9	87.0	80.92	TB
162	Laki- Laki	85.0	76.0	83.38	88.4	88.0	DKV
163	Laki- Laki	71.63	71.8	82.64	70.87	66.83	PH

Lampiran 5. Algoritma Random Forest

5.1 Persiapan Data

5.1.1 Import Library

```
import pandas as pd
import numpy as np
import plotly.express as px
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import StandardScaler
from sklearn.metrics import accuracy_score, classification_report
from sklearn.svm import SVC
from sklearn.ensemble import RandomForestClassifier
from imblearn.over_sampling import SMOTE
import joblib

print("✅ Library berhasil diimpor")
```

5.1.2 Upload File

```
from google.colab import files
import pandas as pd

# Upload file dataset
uploaded = files.upload()

# Membaca file Excel
file_name = 'Data Siswa SMK.xlsx'

print(f"Reading data from: {file_name}")
df = pd.read_excel(file_name)

print("✅ Data berhasil dimuat.")
```

5.1.3 Pembagian Data

```
X_train, X_test, y_train, y_test = train_test_split(
    X, y,
    test_size=0.2,
    random_state=42,
    stratify=y # menjaga proporsi kelas target seimbang antara training & testing
)

print(f'Jumlah data training : {X_train.shape[0]} sampel ({X_train.shape[0]/len(X)*100:.1f}%)')
print(f'Jumlah data testing : {X_test.shape[0]} sampel ({X_test.shape[0]/len(X)*100:.1f}%)')

print('\nDistribusi kelas pada data training:')
print(y_train.value_counts().sort_index().to_string())

print('\nDistribusi kelas pada data testing:')
print(y_test.value_counts().sort_index().to_string())
```

5.2 Preprocessing Data

5.2.1 Pemeriksaan Missing Value

```
missing = df.isnull().sum()

print(missing)

if missing.sum() == 0:
    print("Tidak ada missing value")

Bahasa Indonesia      0
Matematika            0
IPA                   0
IPS                   0
Bahasa Inggris        0
Jurusan               0
dtype: int64
Tidak ada missing value
```

5.2.2 Transformasi Kategorikal

```
print("0 = Laki-laki, 1 = Perempuan")
print("0 = TSM, 1 = AKL, 2 = DKV, 3 = TB, 4 = PH")

0 = Laki-laki, 1 = Perempuan
0 = TSM, 1 = AKL, 2 = DKV, 3 = TB, 4 = PH
```

5.2.3 Menentukan Fitur X dan Target Y

```
X = df[['Bahasa Indonesia', 'Matematika',
        'IPA', 'IPS', 'Bahasa Inggris']]

y = df['Jurusan']

print(X.shape)
print(y.shape)

(813, 5)
(813,)
```

5.3 Program *Random Forest*

5.3.1 Sintax *Random Forest*

```
# Library Random Forest
from sklearn.ensemble import RandomForestClassifier

# Validasi dan tuning
from sklearn.model_selection import GridSearchCV, RepeatedStratifiedKFold

# Evaluasi model
from sklearn.metrics import (
    classification_report,
    confusion_matrix,
    precision_score,
    recall_score,
    f1_score
)

print("Library Random Forest siap")

from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import GridSearchCV, RepeatedStratifiedKFold

model = RandomForestClassifier(random_state=42)

param_grid = {
    'n_estimators': [100, 200],
    'max_depth': [None, 20],
    'max_features': ['sqrt'],
    'min_samples_split': [2, 5],
    'min_samples_leaf': [1, 2]
}

cv = RepeatedStratifiedKFold(n_splits=5, n_repeats=2, random_state=1)

grid_search = GridSearchCV(
    estimator=model,
    param_grid=param_grid,
    scoring='f1_weighted',
    cv=cv,
    n_jobs=-1,
    verbose=1
)

best_model_rf = grid_search.fit(X_train, y_train)

rf_pred = best_model_rf.predict(X_test)
```

5.3.2 Hasil Output *Random Forest*

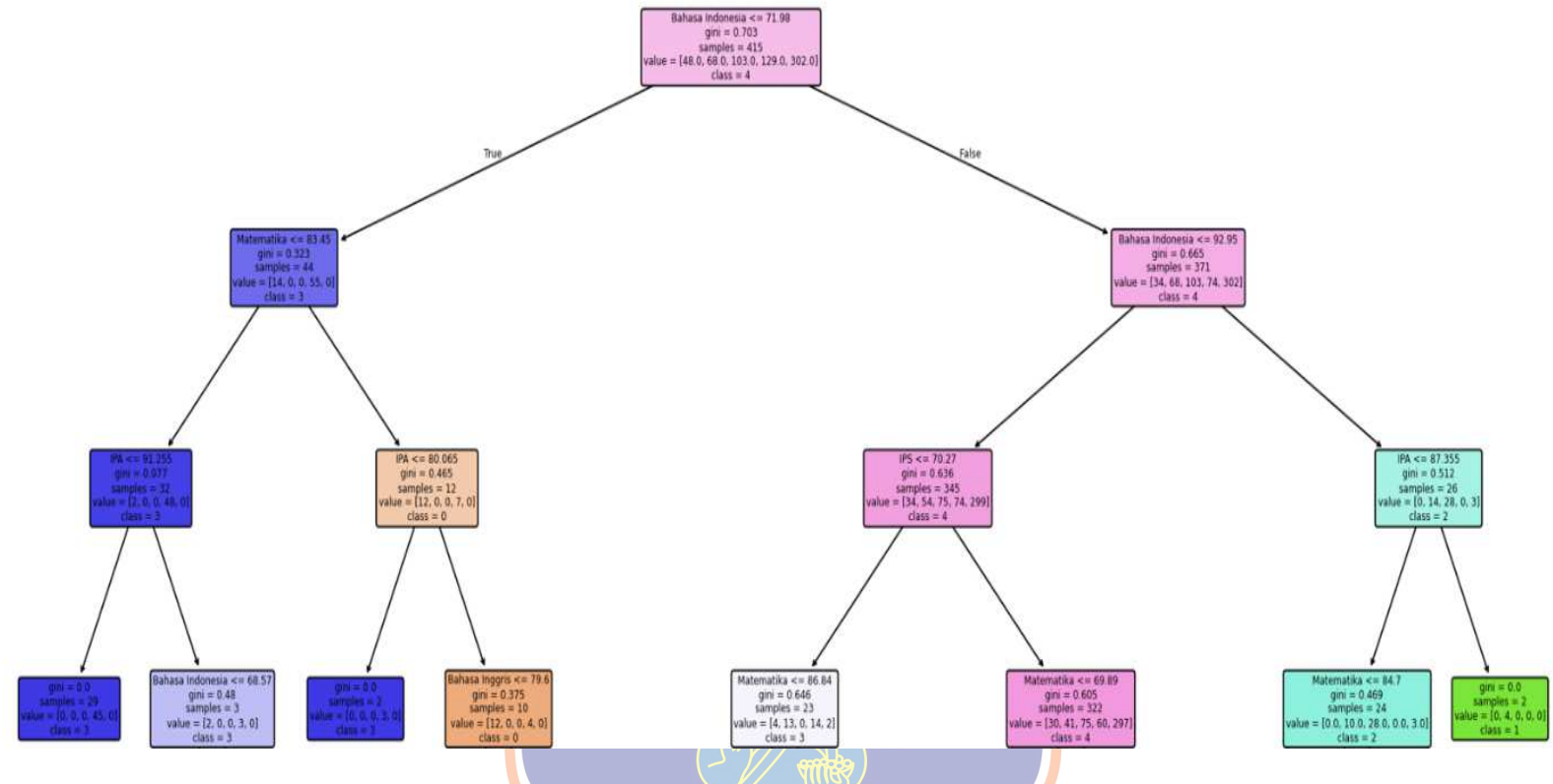
Best Model : RandomForestClassifier(n_estimators=200, random_state=42)

Best Score : 0.8072866538764224

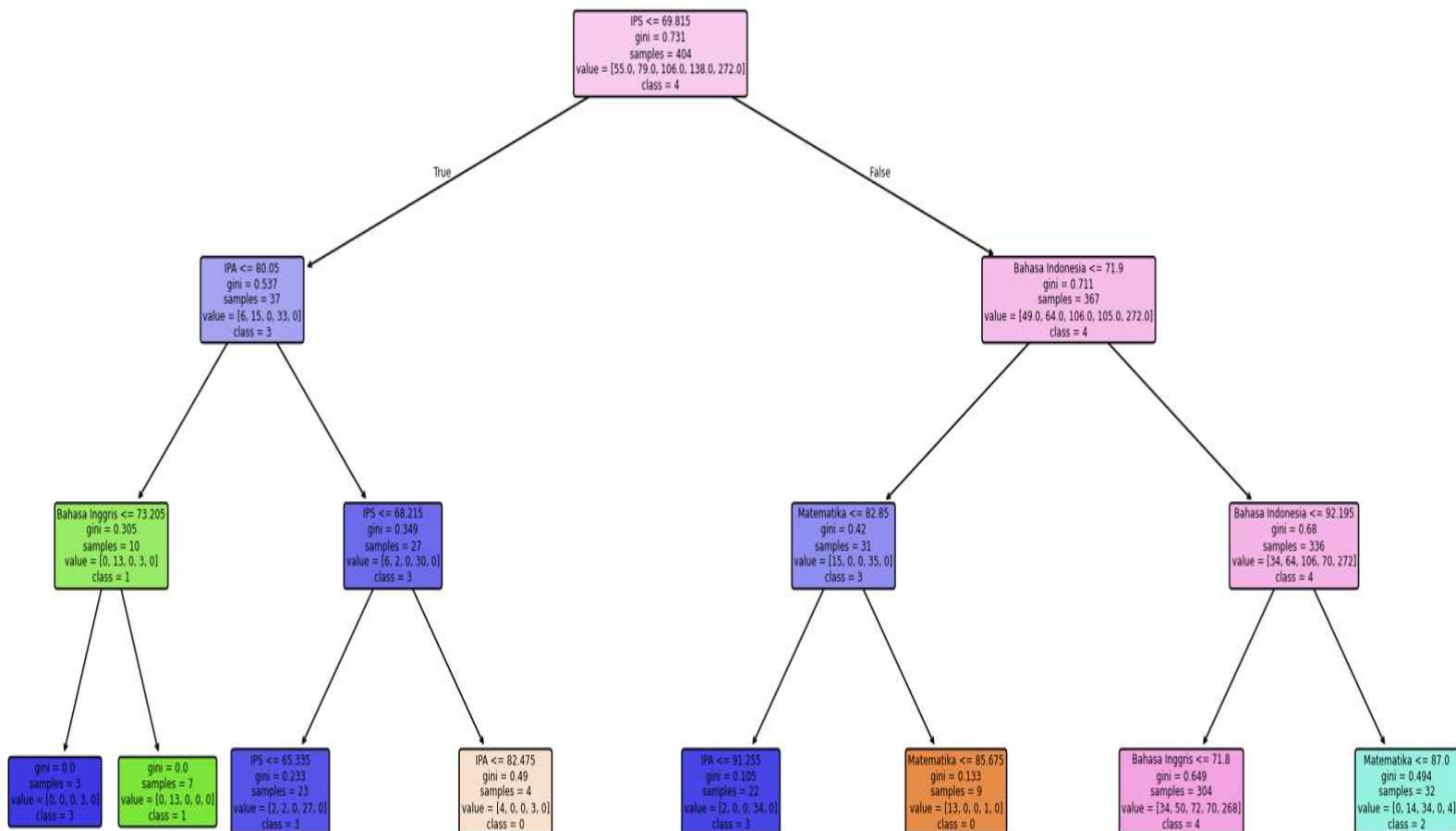
Best Params : {'max_depth': None, 'max_features': 'sqrt', 'min_samples_leaf': 1, 'min_samples_split': 2, 'n_estimators': 200}

5.3.3 Visualisasi Desicion Tree

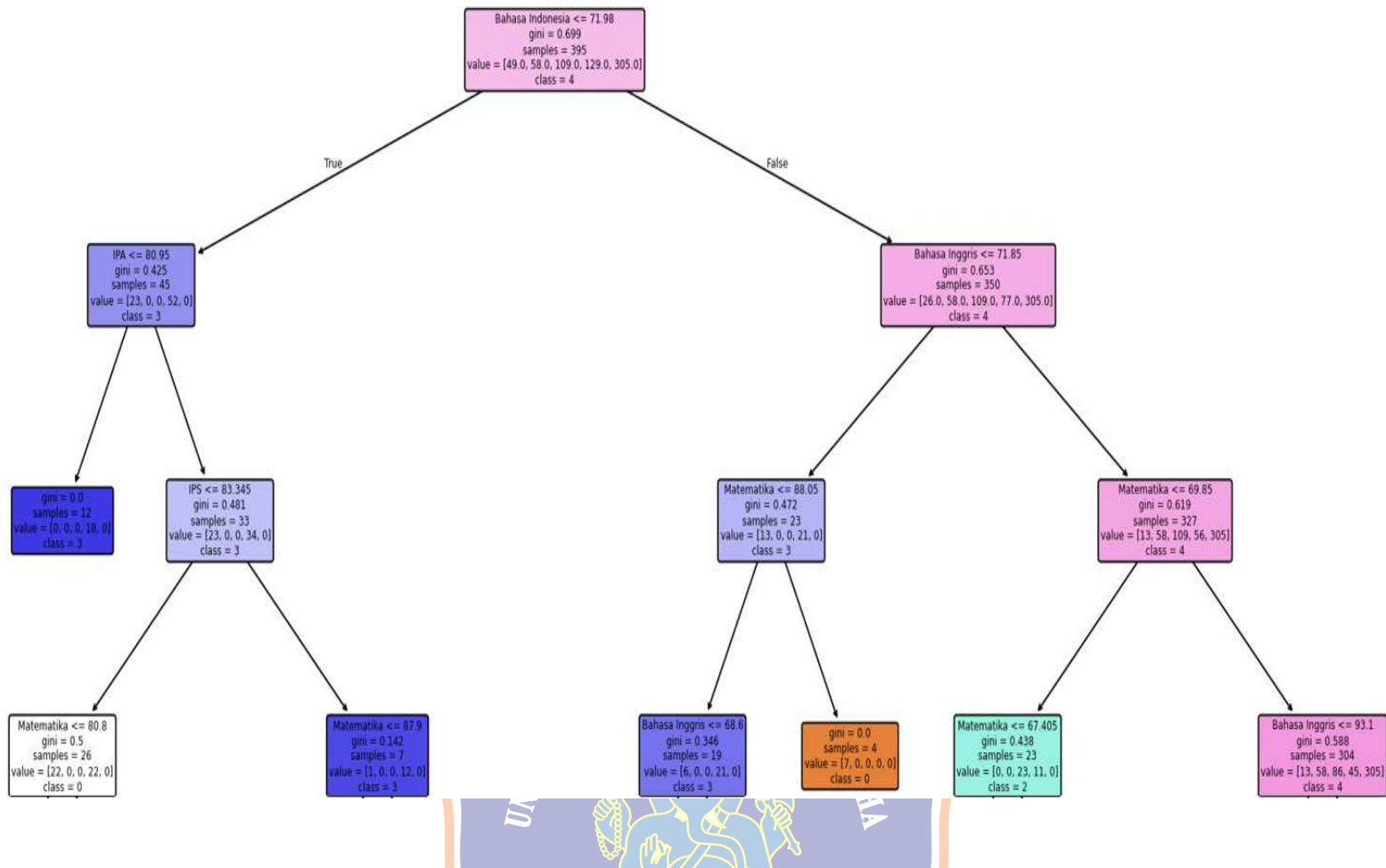
Random Forest - Pohon ke-1 dari 200



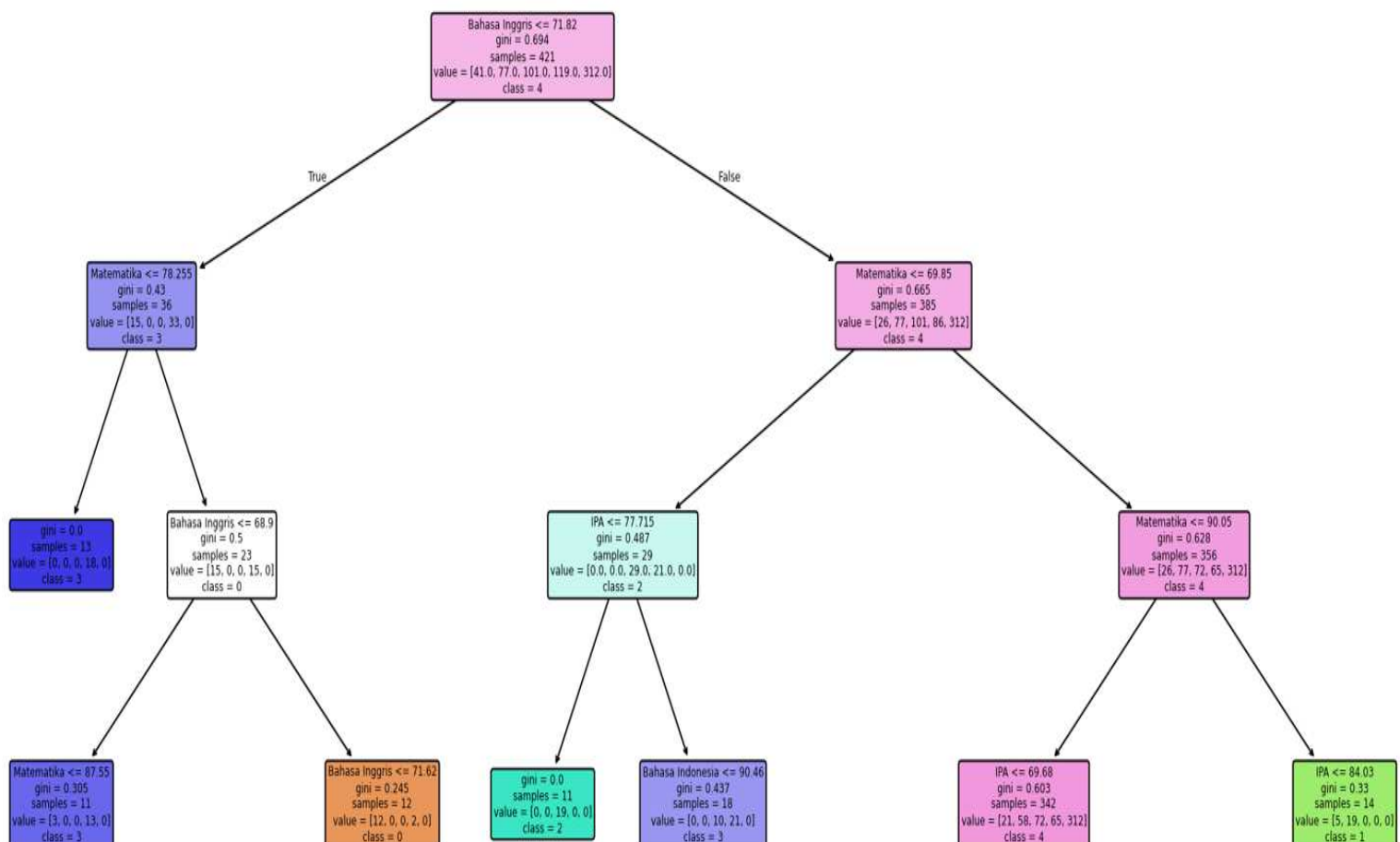
Random Forest - Pohon ke-30 dari 200



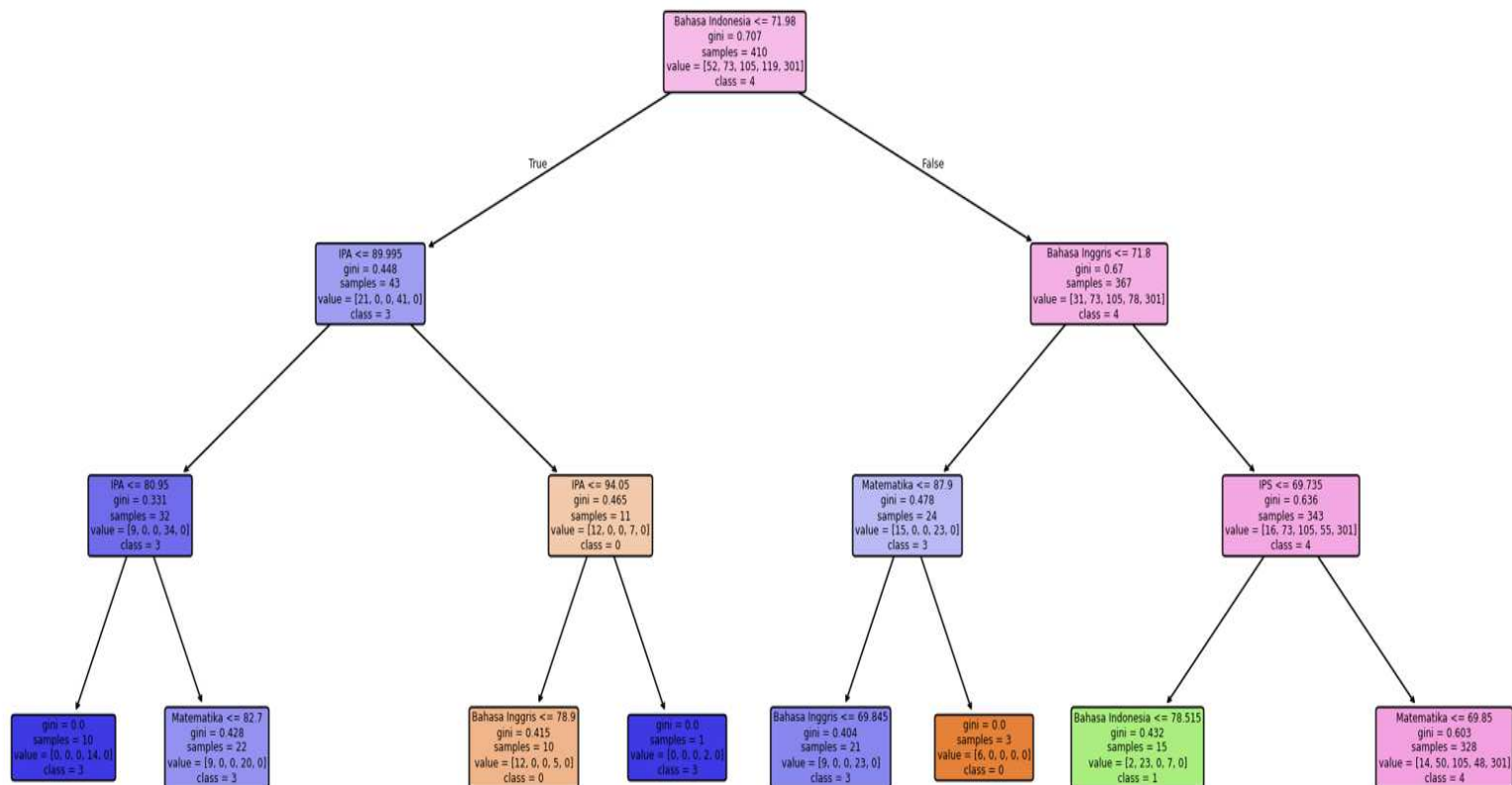
Random Forest - Pohon ke-60 dari 200



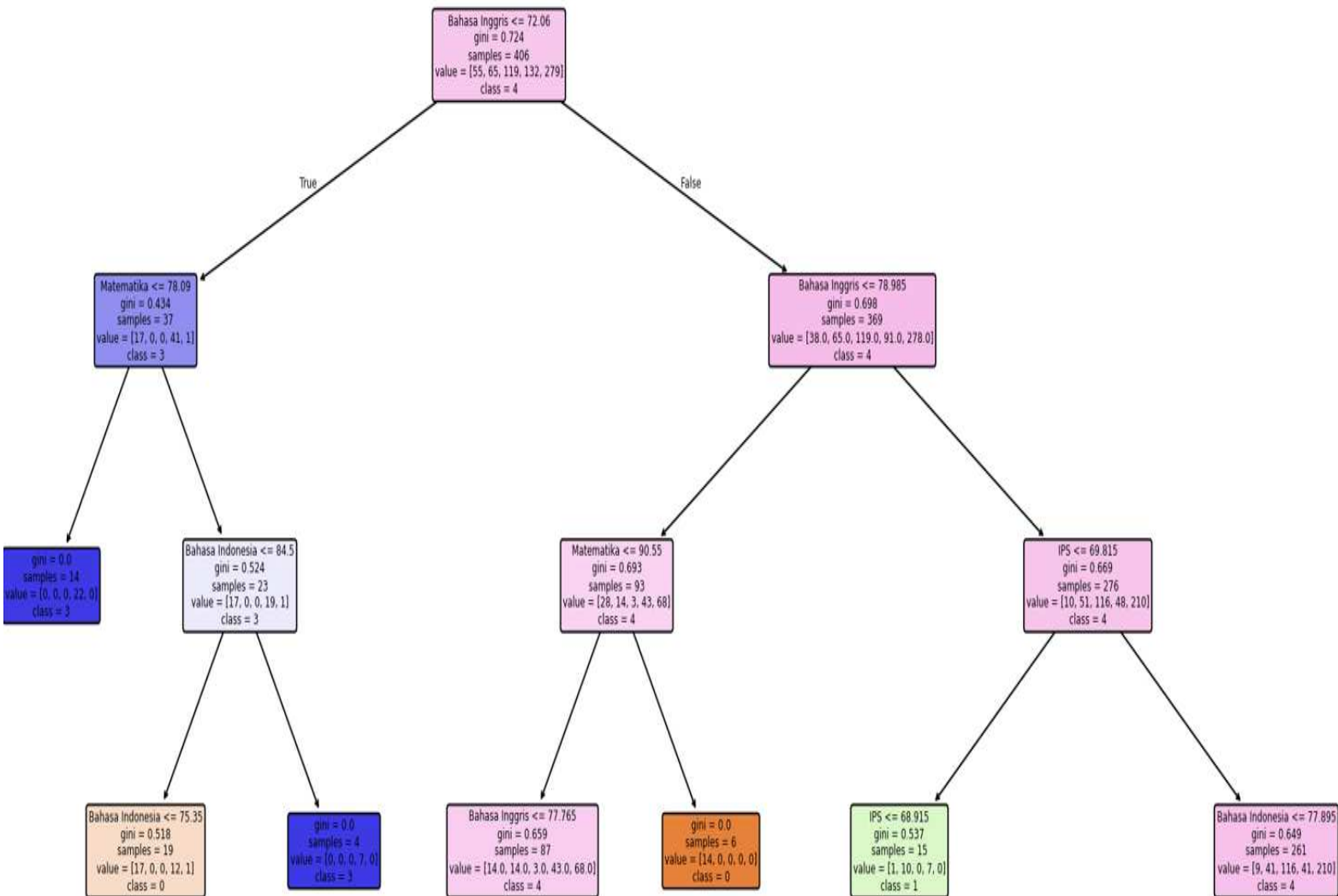
Random Forest - Pohon ke-100 dari 200



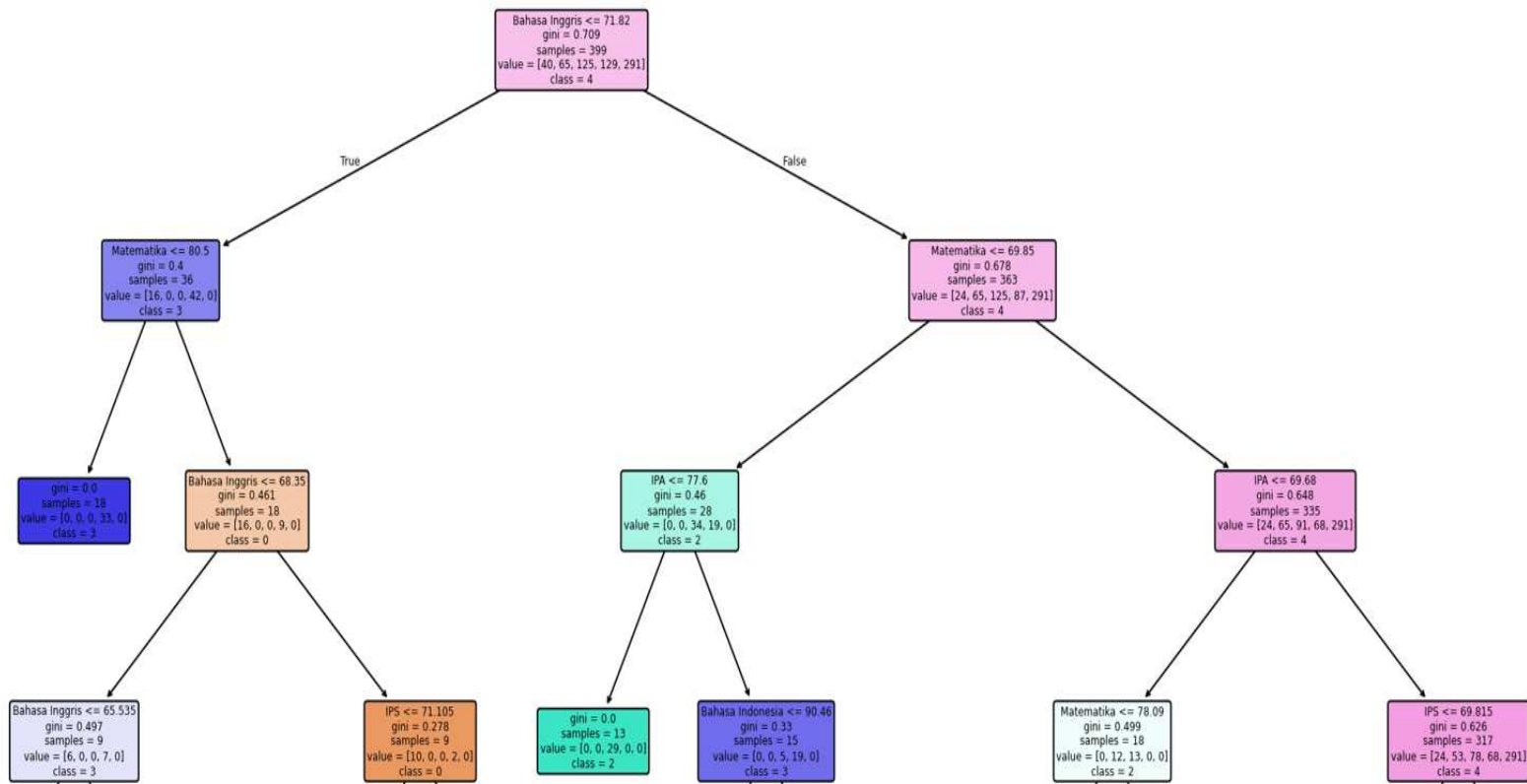
Random Forest - Pohon ke-120 dari 200



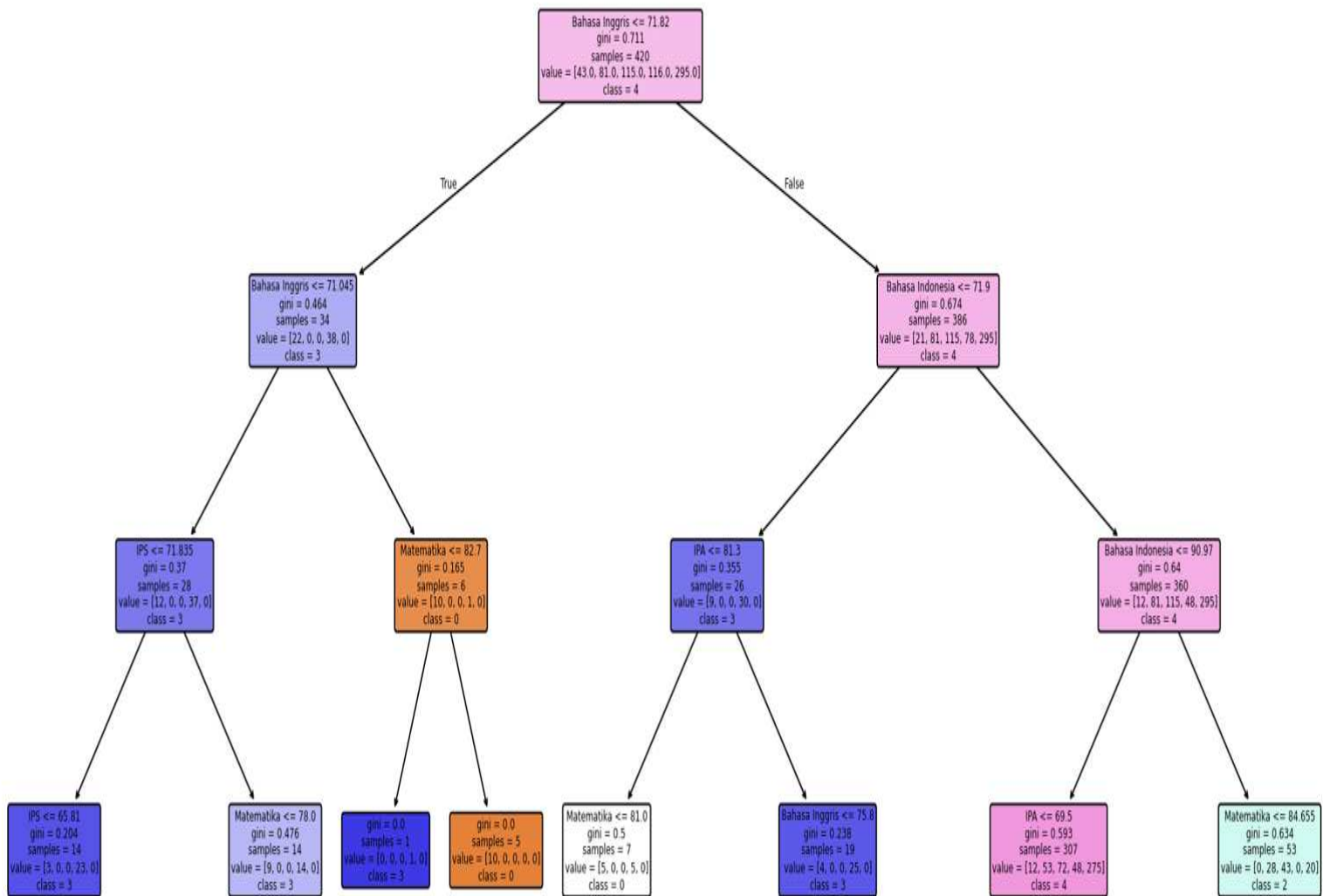
Random Forest - Pohon ke-140 dari 200



Random Forest - Pohon ke-160 dari 200



Random Forest - Pohon ke-200 dari 200



5.3.4 Hasil Prediksi *Random Forest*

No.	Jurusan Aktual	Jurusan Prediksi	Keterangan
1	PH	PH	Benar
2	PH	PH	Benar
3	TSM	TSM	Benar
4	PH	PH	Benar
5	TB	TB	Benar
6	PH	PH	Benar
7	PH	PH	Benar
8	AKL	AKL	Benar
9	PH	PH	Benar
10	TB	TB	Benar
⋮			
80	PH	PH	Benar
81	TB	TB	Benar
82	AKL	AKL	Benar
83	DKV	DKV	Benar
84	PH	TB	Salah
85	PH	PH	Benar
86	DKV	DKV	Benar
87	PH	PH	Benar
89	PH	PH	Benar
⋮			
154	TSM	TSM	Benar
155	PH	PH	Benar
156	DKV	DKV	Benar
157	TB	TB	Benar
158	DKV	DKV	Benar
159	PH	PH	Benar
160	TB	TB	Benar
161	DKV	DKV	Benar
162	PH	PH	Benar
163	TB	TB	Benar

5.4 Program SVM

5.4.1 Sintax Program SVM

```
# Pipeline (WAJIB buat SVM)
pipeline = Pipeline([
    ('scaler', StandardScaler()),
    ('svm', SVC(class_weight='balanced'))
])

# Parameter grid (lebih luas & proper)
param_grid = {
    'svm__kernel': ['rbf', 'poly'],
    'svm__C': [0.1, 1, 10, 50, 100],
    'svm__gamma': ['scale', 'auto', 0.1, 0.01],
    'svm__degree': [2, 3] # cuma kepake kalau poly
}

# Cross validation
cv = RepeatedStratifiedKFold(n_splits=10, n_repeats=3, random_state=1)

# GridSearch
grid_search = GridSearchCV(
    estimator=pipeline,
    param_grid=param_grid,
    scoring='f1_macro', # FIX multi-class
    cv=cv,
    n_jobs=-1,
    verbose=1
)

# FIT
model_terbaik_svm = grid_search.fit(X, y)

# Predict
svm_pred = model_terbaik_svm.predict(X_test)
```

5.4.2 Hasil Output SVM

Best Model : Pipeline(steps=[('scaler', StandardScaler()),
('svm', SVC(C=100, class_weight='balanced', degree=2))])

Best Score : 0.6701338717238434

Best Params : {'svm__C': 100, 'svm__degree': 2, 'svm__gamma': 'scale', 'svm__kernel': 'rbf'}

5.4.3 Hasil Prediksi SVM

Siswa	Level TSM	Level AKL	Level DKV	Level TB	Level PH	Prediksi
1	-1	-1	-1	-1	1	PH
2	-1	-1	-1	-1	1	PH
3	1	-1	-1	-1	-1	TSM
4	-1	-1	-1	-1	1	PH
5	-1	-1	-1	1	-1	TB
6	-1	-1	-1	-1	1	PH
7	-1	-1	-1	-1	1	PH
8	-1	1	-1	-1	-1	AKL
9	-1	-1	-1	-1	1	PH
10	-1	-1	-1	1	-1	TB
:						
80	-1	-1	-1	-1	1	PH
81	-1	-1	-1	1	-1	TB
82	-1	1	-1	-1	-1	AKL

83	-1	-1	1	-1	-1	DKV
84	-1	-1	-1	1	-1	TB
85	-1	-1	-1	-1	1	PH
86	-1	-1	1	-1	-1	DKV
87	-1	-1	-1	-1	1	PH
88	-1	-1	-1	-1	1	PH
89	-1	-1	1	-1	-1	DKV
:						
154	1	-1	-1	-1	-1	TSM
155	-1	-1	-1	-1	1	PH
156	-1	-1	1	-1	-1	DKV
157	-1	-1	-1	1	-1	TB
158	-1	-1	1	-1	-1	DKV
159	-1	-1	-1	-1	1	PH
160	-1	-1	-1	1	-1	TB
161	-1	-1	1	-1	-1	DKV
154	1	-1	-1	-1	-1	TSM
155	-1	-1	-1	-1	1	PH

5.4.4 Hasil Prediksi dan Kelas Aktual pada Data Testing

No.	Jurusan Aktual	Jurusan Prediksi	Keterangan
1	PH	PH	Benar
2	PH	PH	Benar
3	TSM	TB	Salah
4	PH	PH	Benar
5	TB	TB	Benar
6	PH	PH	Benar
7	PH	PH	Benar
8	AKL	AKL	Benar
9	PH	PH	Benar
10	PH	PH	Benar
:			
80	PH	PH	Benar
81	TB	TB	Benar
82	AKL	AKL	Benar
83	DKV	DKV	Benar
84	PH	PH	Benar
85	PH	PH	Benar
86	DKV	DKV	Benar
87	PH	PH	Benar
89	PH	PH	Benar
:			
154	TSM	PH	Salah
155	PH	PH	Benar
156	DKV	PH	Salah
157	TB	TB	Benar

158	DKV	DKV	Benar
159	PH	PH	Benar
160	TB	TB	Benar
161	DKV	DKV	Benar
162	PH	PH	Benar
163	TB	TB	Benar

Lampiran 6. Dokumentasi Penelitian

