**Laporan Pertemuan 6 – Machine Learning**

***Optimasi Model Random Forest untuk Prediksi Kelulusan Mahasiswa***

**Nama:** Galih Naufal Faturrohman  
**NIM:** 231011402731  
**Program Studi:** Informatika – Universitas Pamulang

**1. Pendahuluan**

Tahap ini merupakan kelanjutan dari analisis sebelumnya yang berfokus pada pembangunan model prediksi kelulusan mahasiswa menggunakan algoritma *Machine Learning*.  
Jika pada pertemuan ke-5 dilakukan perbandingan antara *Logistic Regression* dan *Random Forest*, maka pada pertemuan ke-6 ini fokus diarahkan pada **optimasi performa model Random Forest** dengan menggunakan *cross-validation* dan *hyperparameter tuning*.

Pendekatan ini bertujuan untuk meningkatkan akurasi, kestabilan, serta kemampuan generalisasi model dalam memprediksi kelulusan berdasarkan fitur akademik seperti IPK, jumlah absensi, dan waktu belajar.

**2. Tujuan**

Tujuan utama dari analisis ini adalah:

1 Melakukan pembagian dataset menjadi data latih, validasi, dan uji.

2 Membangun model **Random Forest** dengan preprocessing yang tepat.

3 Melakukan evaluasi awal model menggunakan metrik **F1-score**.

4 Mengoptimalkan parameter model menggunakan **GridSearchCV** dan **StratifiedKFold**.

5 Mengevaluasi hasil akhir model dengan metrik **ROC-AUC**, **precision-recall**, dan **confusion matrix**.

**3. Metode Analisis**

Analisis dilakukan menggunakan dataset *processed\_kelulusan.csv* yang telah melalui tahap pembersihan dan rekayasa fitur pada pertemuan sebelumnya.  
Langkah-langkah utama yang dilakukan adalah:

**1 Preprocessing Data:**

◦ Menangani nilai kosong dengan *SimpleImputer* (strategi median).

◦ Menstandarkan fitur numerik menggunakan *StandardScaler*.

**2 Modeling:**

◦ Menggunakan algoritma **Random Forest Classifier** dengan parameter awal:

▪ n\_estimators = 300

▪ max\_features = "sqrt"

▪ class\_weight = "balanced"

**3 Validasi Model:**

◦ Menggunakan *StratifiedKFold* sebanyak 2 fold untuk menjaga proporsi kelas.

◦ Melakukan *cross-validation* dengan metrik F1-macro untuk mengukur stabilitas model.

**4 Hyperparameter Tuning:**

◦ Menggunakan *GridSearchCV* dengan variasi parameter:

▪ max\_depth = [None, 12, 20, 30]

▪ min\_samples\_split = [2, 5, 10]

◦ Pemilihan model terbaik berdasarkan nilai rata-rata F1-score tertinggi.

**5 Evaluasi Model:**

◦ Menggunakan metrik **F1-score**, **ROC-AUC**, **precision-recall curve**, dan **confusion matrix**.

◦ Analisis pentingnya fitur (*feature importance*) dilakukan untuk mengetahui variabel paling berpengaruh terhadap kelulusan.

**4. Hasil dan Pembahasan**

Model awal **Random Forest** menghasilkan performa yang baik pada data validasi dengan nilai F1-macro yang stabil.  
Setelah dilakukan tuning parameter, didapatkan peningkatan performa pada data validasi dan uji.

**Temuan utama:**

*• Cross-validation* menunjukkan hasil yang konsisten dengan rata-rata F1-macro relatif tinggi.

• Parameter terbaik diperoleh pada kombinasi kedalaman pohon menengah (max\_depth=20) dan jumlah minimum sampel split kecil (min\_samples\_split=2).

*• ROC Curve* menunjukkan model memiliki kemampuan klasifikasi yang baik dengan area di bawah kurva (*AUC*) mendekati nilai maksimum.

• Hasil *feature importance* menunjukkan bahwa variabel **IPK** dan **IPK\_x\_Study** memiliki pengaruh paling signifikan terhadap status kelulusan mahasiswa.

Dengan demikian, model yang telah dioptimasi memiliki kemampuan prediktif yang lebih baik dibandingkan model baseline dari pertemuan sebelumnya.

**5. Kesimpulan**

1 Model **Random Forest** yang dioptimasi menggunakan *GridSearchCV* menunjukkan peningkatan performa dalam memprediksi kelulusan mahasiswa.

2 Proses *cross-validation* membantu menjaga keandalan model dengan memastikan hasil evaluasi tidak tergantung pada satu subset data saja.

3 Variabel akademik seperti **IPK** dan **jumlah absensi** menjadi faktor dominan yang memengaruhi hasil prediksi.

4 Model yang dihasilkan memiliki kemampuan diskriminatif yang baik berdasarkan nilai ROC-AUC dan F1-score.

5 Untuk penelitian lanjutan, disarankan memperbanyak jumlah data serta mencoba algoritma lain seperti Gradient Boosting atau XGBoost untuk perbandingan performa.

**Catatan:**  
Seluruh proses analisis dilakukan menggunakan bahasa Python pada file main 6.py, yang mencakup preprocessing, pelatihan model, validasi, serta visualisasi kurva ROC dan Precision-Recall.