



## Penerapan *Machine Learning* untuk Klasifikasi Tingkat Produktivitas Pabrik Karet GAPKINDO Jambi

Wilujeng Niar Raharjanto<sup>1\*</sup>, Riza Pahlevi<sup>2\*</sup>, Lies Aryani<sup>3</sup>, Roby Setiawan<sup>4</sup>

<sup>1,2\*</sup>Fakultas Ilmu Komputer, Program Studi Teknik Informatika, Universitas Dinamika Bangsa, Jambi, Indonesia

<sup>3</sup>Fakultas Ilmu Komputer, Program Studi Sistem Informasi, Universitas Dinamika Bangsa, Jambi, Indonesia

<sup>4</sup>Fakultas Ilmu Manajemen Dan Bisnis, Program Studi Bisnis Digital, Universitas Dinamika Bangsa, Jambi, Indonesia

Email: [wljgniar@gmail.com](mailto:wljgniar@gmail.com)<sup>1\*</sup>, [rizapahlevi@unama.ac.id](mailto:rizapahlevi@unama.ac.id)<sup>2\*</sup>, [liesaryani6@gmail.com](mailto:liesaryani6@gmail.com)<sup>3</sup>, [robby.setiawan.jet@gmail.com](mailto:robby.setiawan.jet@gmail.com)<sup>4</sup>

Alamat: Jl. Jend. Sudirman, The Hok, Kec. Jambi Selatan, Kota Jambi, Jambi 36138

\*Penulis Korespondensi: [wljgniar@gmail.com](mailto:wljgniar@gmail.com)

**Abstract.** *Jambi Province is one of the largest natural rubber producing regions in Indonesia; however, rubber factories under GAPKINDO Jambi still face productivity issues, particularly the gap between production capacity and actual output, and productivity assessment that is still conducted manually by GAPKINDO Jambi. This study employs Decision Tree, Random Forest, KNN, and SVM algorithms within a structured pipeline involving preprocessing, feature selection, standardization, data balancing using SMOTE, and hyperparameter tuning. The proposed solution applies productivity level classification both individually and through paired combinations (ensemble voting). The results show that the Decision Tree + Random Forest model achieves the best performance with an accuracy of 0.84 and an F1-score of 0.83, confirming the effectiveness of ensemble methods in supporting productivity improvement decisions.*

**Keywords:** *Classification, decision tree, random forest, SVM, KNN, Gapkindo*

**Abstrak.** Provinsi Jambi merupakan salah satu sentra penghasil karet alam terbesar di Indonesia, namun pabrik-pabrik karet di bawah GAPKINDO Jambi masih menghadapi masalah produktivitas, terutama kesenjangan antara kapasitas dan realisasi produksi dan penilaian produktivitas pabrik karet yang selama ini masih dinilai secara manual oleh GAPKINDO Jambi. Penelitian ini menggunakan algoritma *Decision Tree*, *Random Forest*, KNN, dan SVM dalam *pipeline* terstruktur dengan pra-pemrosesan, seleksi fitur, standarisasi, penyeimbangan data menggunakan SMOTE, serta tuning hyperparameter. Solusi yang diterapkan berupa klasifikasi tingkat produktivitas pabrik secara individu maupun melalui kombinasi berpasangan (*ensemble voting*). Hasil menunjukkan *Decision Tree + Random Forest* memiliki performa terbaik dengan *accuracy* 0,84 dan *F1-score* 0,83, menegaskan efektivitas metode *ensemble* dalam mendukung keputusan peningkatan produktivitas

**Kata kunci:** *Klasifikasi, decision tree, random forest, SVM, KNN, Gapkindo*

### 1. LATAR BELAKANG

Provinsi Jambi merupakan salah satu sentra penghasil karet alam terbesar di Indonesia. Berdasarkan data Badan Pusat Statistik (BPS, 2025), luas areal perkebunan karet di provinsi ini mencapai lebih dari 600 ribu hektar dengan produksi ratusan ribu ton per tahun. Kondisi tersebut menunjukkan bahwa sektor perkaretan memiliki peran strategis dalam mendukung perekonomian daerah.

GAPKINDO (Gabungan Perusahaan Karet Indonesia) merupakan organisasi non-pemerintah yang menaungi perusahaan pengolah dan eksportir karet alam di Indonesia sejak didirikan pada tahun 1971. Organisasi ini berperan dalam pengembangan industri karet

nasional melalui peningkatan mutu, produktivitas, dan daya saing, serta menjadi mitra strategis pemerintah dalam perumusan kebijakan, penetapan standar mutu, pengumpulan data industri, dan pengembangan riset perkaretan. GAPKINDO memiliki struktur organisasi pusat di Jakarta dan daerah di berbagai provinsi penghasil karet, termasuk Provinsi Jambi.

Pabrik-pabrik karet di bawah naungan GAPKINDO Jambi berperan penting dalam mendukung industri dan ekspor karet, namun masih menghadapi permasalahan produktivitas, khususnya kesenjangan antara kapasitas dan realisasi produksi. Penilaian produktivitas oleh Gapkindo Jambi masih bersifat manual dan deskriptif belum mampu memberikan gambaran yang objektif dan prediktif, sehingga terdapat celah untuk dilakukan penelitian berbasis teknologi analitik data melalui pendekatan *machine learning* dengan teknik klasifikasi menggunakan algoritma *Decision Tree*, *Random Forest*, *SVM*, dan *KNN*, termasuk kombinasi berpasangan antar model, untuk menentukan tingkat produktivitas pabrik secara lebih akurat serta mendukung pengambilan keputusan dalam peningkatan kinerja industri karet GAPKINDO Jambi.

## 2. KAJIAN TEORITIS

*Machine learning* merupakan penerapan algoritma komputer dan model matematika yang belajar dari data secara otomatis tanpa instruksi eksplisit. Proses pembelajaran berlangsung melalui dua tahap, yaitu *training* untuk membangun model dari data, serta *testing* untuk mengevaluasi kinerjanya (Roihan, 2020).

*Decision Tree* merupakan algoritma *supervised* yang digunakan untuk tugas klasifikasi. Algoritma ini dapat diterapkan pada data numerik maupun kategorikal. Secara umum, struktur algoritmanya menyerupai sebuah *flowchart*. Proses pengelompokan data dimulai dari *root node* dan terus dipilah hingga mencapai *leaf node* sebagai hasil klasifikasi akhir. Pembentukan struktur *Decision Tree* dilakukan dengan memilih atribut terbaik sebagai pemisah data berdasarkan nilai entropy dengan rumus sebagai berikut (Chan, 2024).

$$Entropy (s) = \sum_{i=1}^n - p_i * \log_2 p_i \quad (1)$$

Keterangan :

- S : Himpunan Kasus
- n : Jumlah partisi S
- Pi : Proporsi dari Si terhadap S

*Random Forest* merupakan salah satu algoritma klasifikasi dan regresi yang populer dalam *machine learning*. Proses klasifikasi pada *Random Forest* dilakukan dengan membangun banyak pohon keputusan melalui training beberapa versi pohon, lalu menggabungkan hasilnya untuk mendapatkan hasil akhir, hasil Klasifikasi *Random Forest* dapat dijelaskan melalui persamaan berikut (Sebayang, 2023).

$$y = \operatorname{argmax}_c \sum_{n=1}^n (h_n(y) = c) \quad (2)$$

Keterangan :

$h_n(y)$  : prediksi dari pohon keputusan ke-n terhadap data y.

$(h_n(y) = c)$  : fungsi indikator (1 jika pohon memprediksi kelas c, 0 jika tidak).

$(\sum_{n=1}^n)$  : jumlah suara (vote) dari seluruh kelas.

$\operatorname{argmax}$  : memilih kelas dengan suara terbanyak.

*Support Vector Machine* (SVM) merupakan salah satu algoritma klasifikasi yang bekerja dengan mencari *hyperplane* sebagai batas pemisah antar kelompok data secara optimal. Tujuan utamanya adalah mencari garis pemisah dengan jarak margin terbesar agar hasil klasifikasi menjadi lebih akurat dengan menggunakan rumus (Utami, 2023).

$$f(x) = \operatorname{sign} \left( \sum_{i=1}^m \alpha_i y_i K(x_i, x) + b \right) \quad (3)$$

Keterangan :

$f(x)$  : Hasil fungsi keputusan untuk menentukan kelas data uji.

$K(x_i, x)$  : Fungsi kernel untuk menghitung kemiripan dua data.

$\alpha_i$  : Koefisien bobot dari setiap support vector

*K-Nearest Neighbor* (KNN) merupakan metode *supervised learning* yang digunakan untuk mengklasifikasikan data baru berdasarkan kedekatan jarak dengan data yang sudah diketahui kelasnya. Algoritma ini bekerja dengan menghitung jarak antara data uji dan data latih, kemudian menentukan kelas data baru berdasarkan mayoritas kelas dari sejumlah tetangga terdekat yang paling mirip dengan menggunakan rumus (Ginting, 2024).

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2} \quad (4)$$

Keterangan :

$x_i$  : Sample data

$x_i$  : Data uji

i : Variable data

d : Jarak

p : Dimensi data

Dalam bidang *machine learning*, *ensemble learning* menjadi salah satu pendekatan yang efektif karena menggabungkan beberapa algoritma untuk memperoleh prediksi yang lebih stabil dan akurat. Teknik ini memanfaatkan kekuatan model yang berbeda sehingga mampu memperbaiki kelemahan model tunggal, terutama pada kasus klasifikasi yang memiliki banyak variasi pola data (Nalatissifa, 2025).

Evaluasi model dilakukan untuk menilai kinerja algoritma dalam proses klasifikasi berdasarkan beberapa indikator pengukuran yaitu *confusion matrix* digunakan untuk mengevaluasi kinerja model dengan membandingkan hasil klasifikasi sebenarnya dan hasil prediksi (Setyobudi, 2025).

Akurasi (*Accuracy*) dapat menunjukkan sejauh mana sistem mampu mengklasifikasikan data dengan benar, nilai akurasi dihitung menggunakan rumus.

$$Accuracy = \frac{TP + TN}{TP + FN + FP + TN} \quad (5)$$

Presisi (*Precision*) menunjukkan tingkat ketepatan prediksi model, yaitu perbandingan antara jumlah prediksi yang benar dengan seluruh data yang diprediksi positif. Rumus presisi dinyatakan sebagai berikut.

$$Presisi = \frac{TP}{TP + FP} \quad (6)$$

*Recall* menunjukkan kemampuan model dalam mengenali data relevan pada kelas positif. Rumus recall dinyatakan sebagai berikut.

$$Recall = \frac{TP}{TP + FN} \quad (7)$$

*F1-Score* merupakan ukuran kinerja model yang menggabungkan nilai *precision* dan *recall* secara seimbang, dengan rumus sebagai berikut.

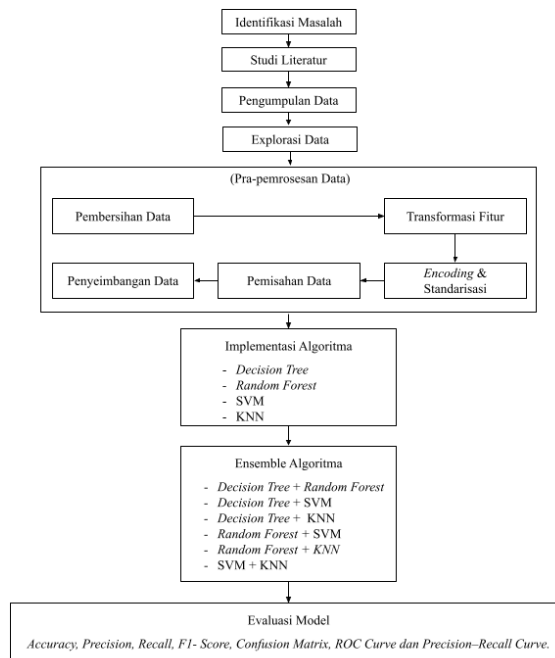
$$F1 - Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad (8)$$

*Receiver Operating Characteristic* (ROC) dan nilai *Area Under the Curve* (AUC) digunakan untuk menilai kemampuan model dalam membedakan antara kelas positif dan negatif. Semakin tinggi nilai AUC, semakin baik kemampuan model dalam melakukan klasifikasi secara akurat (Corbacioglu, 2023).

*Precision-Recall Curve* digunakan untuk menggambarkan hubungan antara ketepatan model (*precision*) dan (*recall*). Kurva ini memberikan gambaran yang lebih informatif dibandingkan metrik tunggal seperti akurasi karena mampu menunjukkan keseimbangan antara kedua ukuran tersebut dalam evaluasi performa model (Davis, 2020).

### 3. METODE PENELITIAN

Untuk memberikan gambaran tahapan – tahapan yang dilakukan penelitian ini, maka dibutuhkan sebuah kerangka kerja penelitian yang dijadikan sebagai pedoman agar penelitian dapat tersusun secara sistematis serta konsiste, sebagaimana ditunjukkan pada gambar berikut.



**Gambar 1.** Tahapan Penelitian

Berdasarkan kerangka kerja penelitian tersebut maka dapat diuraikan pembahasan pada masing - masing tahapan penelitian sebagai berikut :

Tahap identifikasi masalah dilakukan dengan meninjau kondisi saat ini, yaitu belum adanya metode khusus untuk mengklasifikasikan tingkat produktivitas pabrik karet yang tergabung dalam GAPKINDO Jambi, di mana penilaian masih dilakukan secara manual melalui laporan bulanan.

Selanjutnya dilakukan studi literatur dengan mengumpulkan jurnal dan artikel terkait produktivitas pabrik karet serta penerapan model klasifikasi pada sektor industri sebagai landasan teori penelitian.

Tahap pengumpulan data dilakukan melalui data sekunder berupa laporan produksi bulanan dari GAPKINDO Jambi, yang didukung dengan wawancara untuk memahami alur pelaporan dan permasalahan penilaian produktivitas.

Tahap eksplorasi data dilakukan dengan memeriksa struktur dataset, nilai kosong, dan duplikasi data, serta memvisualisasikan distribusi kelas produktivitas sebagai dasar penentuan normalisasi sebelum proses s klasifikasi.

Tahap pra-pemrosesan data bertujuan untuk membersihkan dan menyiapkan data agar konsisten serta siap digunakan dalam pemodelan. Proses ini meliputi,

**a. Pembersihan Data**

Penanganan nilai kosong menggunakan imputasi median untuk atribut numerik dan nilai paling sering muncul untuk atribut kategorikal.

**b. Transformasi Fitur**

Menambahkan fitur Rasio Produktivitas dan Persentase Sisa, serta mengonversi rasio produktivitas ke dalam kategori Rendah, Sedang dan Tinggi.

**c. Encoding dan Standarisasi**

Mengubah fitur kategorikal tertentu ke bentuk numerik menggunakan Label *Encoding* dan *One-Hot Encoding* pada fitur kategorikal lainnya, serta melakukan standarisasi fitur numerik menggunakan *StandardScaler*.

**d. Pemisahan Data**

Membagi data menjadi data latih dan data uji dengan perbandingan 80% data latih dan 20% data uji.

**e. Penyeimbangan Data**

Menangani ketidakseimbangan kelas pada data latih menggunakan metode SMOTE sebelum proses pemodelan.

**f. Implementasi Algoritma**

Pada tahap pemodelan, penelitian ini menggunakan algoritma *Decision Tree*, *Random Forest*, SVM dan KNN. Seluruh model dibangun menggunakan *Pipeline* yang mencakup tahapan *preprocessing* dan pemodelan, serta dilakukan tuning hyperparameter menggunakan *RandomizedSearchCV* dengan skema *5-fold StratifiedKFold*.

**g. Ensemble Algoritma**

Pada tahap *ensemble*, model dibangun secara *pairwise* dengan mengkombinasikan dua algoritma, yaitu *Decision Tree + Random Forest*, *Decision Tree + SVM*, *Decision Tree + KNN*, *Random Forest + SVM*, *Random Forest + KNN*, serta *SVM + KNN*.

**h. Evaluasi Model**

Evaluasi model dilakukan terhadap model individu dan model *ensemble* menggunakan data uji. Hasil utama ditinjau berdasarkan *accuracy*, *precision macro*, *recall macro*, dan *F1-score macro*. Hasil evaluasi disajikan dalam tabel perbandingan, serta analisis lebih lanjut menggunakan *confusion matrix*, kurva ROC, dan *precision-recall curve* (PRC) untuk menilai performa klasifikasi secara menyeluruh.

#### 4. HASIL DAN PEMBAHASAN

Bagian ini membahas hasil eksperimen yang mencakup eksplorasi dan pemaparan data penelitian, tahapan prapemrosesan data, pembagian data latih dan data uji, proses pemodelan klasifikasi menggunakan beberapa algoritma, serta evaluasi kinerja model individu dan model ensemble.

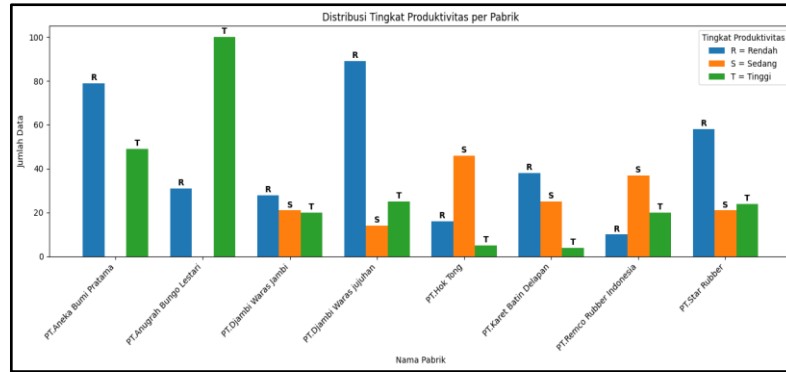
##### Data Penelitian

Tabel berikut menyajikan informasi dataset berupa nama dan jumlah fitur, tipe data masing-masing fitur, serta contoh data.

**Tabel 1.** Informasi Fitur Pada *Dataset*

| No | Nama Fitur                     | Tipe Data      | Contoh Data                       |
|----|--------------------------------|----------------|-----------------------------------|
| 1  | id_Pabrik                      | <i>Object</i>  | SAK                               |
| 2  | Nama_pabrik                    | <i>Object</i>  | PT. Aneka Bumi Pratama            |
| 3  | lokasi                         | <i>Object</i>  | Simpang Kubu Kandang,<br>Pemayung |
| 4  | tahun                          | <i>Float64</i> | 2020                              |
| 5  | bulan                          | <i>Float64</i> | 1                                 |
| 6  | Jenis_produk                   | <i>Object</i>  | SIR 20                            |
| 7  | Kapasitas_produksi_(ton)       | <i>Float64</i> | 16.000                            |
| 8  | Jml_produksi_(ton)             | <i>Float64</i> | 14.950                            |
| 9  | Jml_penjualan_(ton)            | <i>Float64</i> | 6.788                             |
| 10 | sisaproduksi_(ton)             | <i>Float64</i> | 8.158                             |
| 11 | kualitas_karet                 | <i>Object</i>  | B                                 |
| 12 | jml_karyawan                   | <i>Float64</i> | 792                               |
| 13 | jml_pekerja_pembelian_bahan    | <i>Float64</i> | 15                                |
| 14 | jml_pekerja_bongkar_muat       | <i>Float64</i> | 90                                |
| 15 | jml_proses_basah               | <i>Float64</i> | 250                               |
| 16 | jml_proses_kering              | <i>Float64</i> | 215                               |
| 17 | jml_pekerja_gudang_barang_jadi | <i>Float64</i> | 30                                |
| 18 | jml_pekerja_labor              | <i>Float64</i> | 20                                |
| 19 | jml_mekanik                    | <i>Float64</i> | 60                                |
| 20 | jml_pengawas                   | <i>Float64</i> | 15                                |
| 21 | Petugas_kebersihan             | <i>Float64</i> | 37                                |
| 22 | Staff_kantor                   | <i>Float64</i> | 20                                |
| 23 | keamanan                       | <i>Float64</i> | 30                                |
| 24 | supir                          | <i>Float64</i> | 10                                |

Hasil eksplorasi data menunjukkan bahwa dataset berada dalam kondisi layak digunakan. Meskipun distribusi tingkat produktivitas antar pabrik tidak seimbang. Temuan ini menjadi dasar dalam penerapan pra-pemrosesan yang tepat agar model klasifikasi dapat menghasilkan prediksi yang lebih akurat.



Gambar 2. Distribusi Produktivitas

### Tahapan Pemodelan Klasifikasi dan Evaluasi

Proses pemodelan diawali dengan pembacaan data dan pengisian nilai kosong menggunakan metode imputasi, serta pengecekan duplikasi data. Selanjutnya dilakukan penambahan fitur dengan membentuk rasio produktivitas, persentase sisa produksi, dan penentuan kelas tingkat produktivitas sebagai variabel target. Data kemudian dibagi menjadi data latih dan data uji. pemodelan dilakukan dengan membangun pipeline untuk setiap algoritma klasifikasi, yaitu *Decision Tree*, *Random Forest*, *SVM*, dan *KNN*, yang mencakup tahapan prapemrosesan, penyeimbangan data menggunakan *SMOTE*, serta tuning hyperparameter untuk memperoleh konfigurasi terbaik.

Selanjutnya, model-model tersebut dikombinasikan secara pairwise menggunakan metode ensemble voting, dengan *soft voting* atau *hard voting*. Bobot model ensemble ditentukan melalui *RandomizedSearchCV* menggunakan skema *StratifiedKfold* dan metrik *F1-score macro*. Hasil evaluasi kinerja model individu dan ensemble disajikan dalam tabel perbandingan berdasarkan *confusion matrix*, *kurva ROC*, *precision-recall curve*, serta metrik *accuracy*, *precision macro*, *recall macro*, dan *F1-score macro*.

Tabel 2. Perbandingan hasil Confusion Matrix

| Model                         | TP Rendah | FN Sedang | FN Tinggi | TP Sedang | FN Rendah | FN Tinggi | TP Tinggi | FN Rendah | FN Sedang |
|-------------------------------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|-----------|
| Decision Tree                 | 55        | 9         | 6         | 24        | 7         | 2         | 45        | 1         | 6         |
| Random Forest                 | 54        | 10        | 6         | 28        | 3         | 2         | 41        | 3         | 8         |
| SVM                           | 55        | 12        | 3         | 26        | 5         | 2         | 37        | 6         | 9         |
| KNN                           | 47        | 11        | 12        | 25        | 1         | 7         | 41        | 5         | 6         |
| Decision Tree + Random Forest | 56        | 9         | 5         | 29        | 3         | 1         | 45        | 1         | 6         |
| Decision Tree + SVM           | 55        | 9         | 6         | 24        | 7         | 2         | 46        | 0         | 6         |

| <b>Model</b>        | <b>TP Rendah</b> | <b>FN Sedang</b> | <b>FN Tinggi</b> | <b>TP Sedang</b> | <b>FN Rendah</b> | <b>FN Tinggi</b> | <b>TP Tinggi</b> | <b>FN Rendah</b> | <b>FN Sedang</b> |
|---------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|
| Decision Tree + KNN | 53               | 9                | 8                | 25               | 6                | 2                | 46               | 0                | 6                |
| Random Forest + SVM | 54               | 11               | 5                | 29               | 3                | 1                | 40               | 3                | 9                |
| Random Forest + KNN | 53               | 11               | 6                | 28               | 3                | 2                | 40               | 4                | 8                |
| SVM + KNN           | 56               | 10               | 4                | 23               | 6                | 4                | 40               | 6                | 6                |

**Tabel 2.** Perbandingan Hasil ROC

| <b>Model</b>                  | <b>AUC Rendah</b> | <b>AUC Sedang</b> | <b>AUC Tinggi</b> | <b>AUC Macro</b> |
|-------------------------------|-------------------|-------------------|-------------------|------------------|
| Decision Tree                 | 0.87              | 0.80              | 0.91              | 0.86             |
| Random Forest                 | 0.96              | 0.94              | 0.93              | 0.94             |
| SVM                           | 0.92              | 0.91              | 0.91              | 0.92             |
| KNN                           | 0.89              | 0.89              | 0.86              | 0.88             |
| Decision Tree + Random Forest | 0.96              | 0.93              | 0.93              | 0.95             |
| Decision Tree + SVM           | 0.93              | 0.90              | 0.96              | 0.93             |
| Decision Tree + KNN           | 0.94              | 0.90              | 0.92              | 0.92             |
| Random Forest + SVM           | 0.96              | 0.94              | 0.94              | 0.95             |
| Random Forest + KNN           | 0.95              | 0.94              | 0.92              | 0.94             |
| SVM + KNN                     | 0.93              | 0.92              | 0.91              | 0.92             |

**Tabel 3.** Perbandingan Hasil PRC

| <b>Model</b>                  | <b>AP Rendah</b> | <b>AP Sedang</b> | <b>AP Tinggi</b> |
|-------------------------------|------------------|------------------|------------------|
| Decision Tree                 | 0.82             | 0.49             | 0.82             |
| Random Forest                 | 0.95             | 0.77             | 0.90             |
| SVM                           | 0.90             | 0.76             | 0.86             |
| KNN                           | 0.87             | 0.65             | 0.75             |
| Decision Tree + Random Forest | 0.96             | 0.74             | 0.92             |
| Decision Tree + SVM           | 0.93             | 0.67             | 0.93             |
| Decision Tree + KNN           | 0.93             | 0.63             | 0.86             |
| Random Forest + SVM           | 0.95             | 0.77             | 0.91             |
| Random Forest + KNN           | 0.95             | 0.78             | 0.89             |
| SVM + KNN                     | 0.91             | 0.75             | 0.87             |

**Tabel 4.** Perbandingan Kinerja Semua Model

| <b>Model</b>                         | <b>Accuracy</b> | <b>Precision Macro</b> | <b>Recall Macro</b> | <b>F1 Macro</b> |
|--------------------------------------|-----------------|------------------------|---------------------|-----------------|
| <i>Decision Tree + Random Forest</i> | <b>0.84</b>     | <b>0.82</b>            | <b>0.85</b>         | <b>0.83</b>     |
| <i>Decision Tree + SVM</i>           | 0.81            | 0.78                   | 0.80                | 0.79            |
| <i>Decision Tree + KNN</i>           | 0.80            | 0.78                   | 0.80                | 0.79            |
| <i>Decision Tree</i>                 | 0.80            | 0.78                   | 0.79                | 0.78            |
| <i>Random Forest + SVM</i>           | 0.79            | 0.79                   | 0.81                | 0.78            |
| <i>Random Forest</i>                 | 0.79            | 0.78                   | 0.80                | 0.78            |

| Model               | Accuracy | Precision Macro | Recall Macro | F1 Macro |
|---------------------|----------|-----------------|--------------|----------|
| Random Forest + KNN | 0.78     | 0.77            | 0.79         | 0.77     |
| SVM + KNN           | 0.77     | 0.75            | 0.76         | 0.75     |
| SVM                 | 0.76     | 0.76            | 0.76         | 0.75     |
| KNN                 | 0.73     | 0.72            | 0.74         | 0.72     |

Hasil evaluasi menunjukkan bahwa model *ensemble Decision Tree + Random Forest* menunjukkan kinerja terbaik dengan nilai *accuracy* 0,84, *precision macro* 0,82, *recall macro* 0,85, dan *F1-macro* 0,83. Hasil ini mengindikasikan bahwa penggabungan dua algoritma berbasis pohon mampu meningkatkan kemampuan model dalam mengklasifikasikan tingkat produktivitas secara lebih seimbang pada seluruh kelas. Model *ensemble* lain seperti *Decision Tree + SVM* dan *Decision Tree + KNN* juga menunjukkan kinerja yang relatif lebih baik dibandingkan model tunggalnya, meskipun peningkatannya tidak terlalu signifikan.

Sementara itu, model KNN memiliki nilai *accuracy* sebesar 0,73 dan *F1-score macro* sebesar 0,72, yang menunjukkan bahwa algoritma ini masih mampu melakukan klasifikasi dengan tingkat performa yang cukup baik, meskipun kinerjanya berada di bawah model berbasis pohon dan model *ensemble* lainnya. Secara keseluruhan, pendekatan *ensemble voting* terbukti lebih efektif dibandingkan model tunggal, dengan kombinasi *Decision Tree + Random Forest* sebagai model paling optimal pada penelitian ini.

## 5. KESIMPULAN DAN SARAN

Hasil penelitian menunjukkan bahwa kombinasi *Decision Tree* dan *Random Forest* merupakan model terbaik dengan nilai *accuracy* sebesar 0,84 dan *F1-macro* sebesar 0,83, sehingga mampu memberikan performa klasifikasi yang paling stabil dan seimbang pada seluruh kelas produktivitas. Temuan ini menunjukkan bahwa penggabungan dua algoritma berbasis pohon dapat saling melengkapi dalam menangkap pola data yang kompleks. Secara umum, pendekatan *ensemble* terbukti menghasilkan kinerja yang lebih baik dibandingkan algoritma tunggal, sehingga pemilihan kombinasi algoritma yang tepat berperan penting dalam meningkatkan efektivitas klasifikasi tingkat produktivitas pabrik karet.

Sebagai pengembangan ke depan, penelitian selanjutnya dapat meningkatkan akurasi dan kestabilan model dengan menambahkan jumlah data, menyertakan fitur tambahan lainnya, menguji model pada dataset lain atau periode berbeda, serta mengeksplorasi algoritma atau kombinasi algoritma lain untuk meningkatkan performa.

## DAFTAR REFERENSI

- A. Roihan, P. A. Sunarya, dan A. S. Rafika, “Pemanfaatan Machine Learning dalam Berbagai Bidang: Review paper,” *IJCIT Indones. J. Comput. Inf. Technol.*, vol. 5, no. 1, May 2020, doi: 10.31294/ijcit.v5i1.7951.
- M. Chan, “Evaluasi Efektivitas Algoritma Klasifikasi Beban Penggunaan Listrik pada Mesin Pabrik Baja,” vol. 19, no. 2, 2024.
- E. R. B. Sebayang, Y. H. Chrisnanto, dan M. , “Klasifikasi Data Kesehatan Mental di Industri Teknologi Menggunakan Algoritma Random Forest,” *IJESPG Journal*, vol. 1, no. 3, p. 237, 2023.
- M. P. Utami, F. Suroso, F. Lailasari H., F. P. J. Sibuea, dan K. Chandra, “Integrasi Algoritma Support Vector Machine dengan Java untuk Memprediksi Kualitas Komponen Otomotif dalam Industri 4.0,” *Techno.Com*, vol. 24, no. 3, pp. 790–797, Aug. 2025, doi: 10.62411/tc.v24i3.12719.
- B.D. Ginting, Y.Yusfrizal, dan L.A.N.Kadim, “Penerapan Algoritma K-Nearest Neighbor untuk Klasifikasi Usaha Masyarakat Berdasarkan Jenis Izin Usaha,” *Modem J. Inform. Dan Sains Teknol.*, vol. 2, no. 4, pp. 92–101, Sept. 2024, doi: 10.62951/modem.v2i4.233.
- H. Nalatissifa, “Implementasi Ensemble untuk Prediksi Produktivitas Tenaga Kerja pada Industri Garmen,” *CONTEN: Computer and Network Technology*, vol. 5, no. 1, pp. 1–7, Juni 2025.
- B. B. Setyobudi, “Strategi Pemeliharaan Preskriptif: Optimalisasi Keandalan Mesin Berbasis Machine Learning Guna Mencegah Terjadinya Downtime pada Mesin Industri,” vol. 2, 2025.
- Ş. K. Corbacioglu and G. Aksel, “Receiver operating characteristic curve analysis in diagnostic accuracy studies: A guide to interpreting the area under the curve value,” *Turk. J. Emerg. Med.*, vol. 23, no. 4, pp. 195–198, Oct. 2023, doi: 10.4103/tjem.tjem\_182\_23.
- J. Davis and M. Goadrich, “The relationship between Precision-Recall and ROC curves,” in *Proceedings of the 23rd international conference on Machine learning - ICML '06*, Pittsburgh, Pennsylvania: ACM Press, pp. 233–240. doi: 10.1145/1143844.1143874. 2020.