JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

Sistem Prediksi Dini Keberhasilan Akademik Mahasiswa Berbasis Random Forest dengan Algoritma CART

Yasir Hasan
Departemen Informatika, STMIK Mulia Darma
Email: yasirhasan.kom@gmail.com

Abstrak
Tingginya angka mahasiswa yang tidak lulus atau tidak menyelesaikan studi tepat waktu menjadi permasalahan utama di perguruan tinggi. Penelitian ini bertujuan mengimplementasikan algoritma CART sebagai base estimator dalam Random Forest untuk memprediksi status keberhasilan akademik mahasiswa berdasarkan data historis akademik. Model dibangun menggunakan 8 variabel akademik yaitu IPS1-IPS4, tingkat kehadiran, SKS diambil, SKS lulus, dan jumlah mata kuliah mengulang, dengan tiga kelas status keberhasilan Tidak Lulus, Lulus Tidak Tepat Waktu, dan Lulus Tepat Waktu. Hasil pengujian pada 10 data uji menunjukkan bahwa Random Forest dengan 100 pohon menghasilkan akurasi 90%, dengan satu perbedaan prediksi pada data yang berada di batas ambang jika dibandingkan dengan satu pohon CART pertama. Berdasarkan validasi silang 5-fold, Random Forest menunjukkan rata-rata akurasi 89% yang lebih stabil dan konsisten. Analisis feature importance mengungkap bahwa Jumlah MK Mengulang merupakan faktor paling dominan (0.28), diikuti oleh SKS Lulus (0.21) dan SKS Diambil (0.16). Sistem prediksi dini yang dikembangkan dengan antarmuka PySimpleGUI dapat digunakan oleh dosen pembimbing akademik dan pengelola program studi sebagai alat bantu pengambilan keputusan dan intervensi akademik secara proaktif.
Kata Kunci: Random Forest, CART, prediksi keberhasilan akademik, klasifikasi mahasiswa, feature importance

Abstract
The high number of students who do not graduate or do not complete their studies on time is a major problem in higher education. This study aims to implement the CART algorithm as a base estimator in Random Forest to predict students' academic success status based on historical academic data. The model was built using 8 academic variables, namely IPS1-IPS4, attendance rate, credits taken, credits passed, and number of courses repeated, with three success status classes: Not Passed, Passed Not on Time, and Graduated on Time. The test results on 10 test data showed that Random Forest with 100 trees produced 90% accuracy, with one difference in prediction on data that was at the threshold when compared to the first CART tree. Based on 5-fold cross-validation, Random Forest showed a more stable and consistent average accuracy of 89%. Feature importance analysis revealed that Number of Courses Repeated was the most dominant factor (0.28), followed by Credits Passed (0.21) and Credits Taken (0.16). The early prediction system developed using the PySimpleGUI interface can be used by academic advisors and study program managers as a tool for proactive decision-making and academic intervention..
Keywords*: Random Forest, CART, academic success prediction, student classification, feature importance*

1. PENDAHULUAN

Keberhasilan akademik mahasiswa merupakan indikator utama kualitas penyelenggaraan pendidikan pada perguruan tinggi [1]. Perguruan tinggi diharapkan mampu menciptakan sistem untuk memantau perkembangan akademik mahasiswa secara berkelanjutan [2]. Pengelolaan data akademik tidak lagi dilakukan secara manual, melainkan berbasis sistem informasi terintegrasi. Data seperti indeks prestasi semester, kehadiran, jumlah SKS yang diambil, hingga riwayat pengulangan mata kuliah seharusnya dapat dianalisis secara komprehensif. Dukungan teknologi analitik membuat institusi pendidikan dapat memetakan pola keberhasilan mahasiswa berdasarkan data historis [3]. Data akademik yang selama ini hanya menjadi catatan historis, dapat diolah menjadi intelijen prediktif yang berharga untuk pemetaan peluang keberhasilan mahasiswa. Data dari berbagai sumber menunjukkan bahwa rata-rata masa studi mahasiswa seringkali melampaui batas waktu ideal yang ditetapkan. Persentase kelulusan tepat waktu menjadi target bagi banyak institusi. Salah satu penyebab utamanya adalah ketidakmampuan sistem yang ada untuk secara dini mendeteksi sinyal-sinyal peringatan (*early warning signs*) [4] dari mahasiswa yang mulai tertinggal atau mengalami kesulitan.

Penerapan *Machine Learning* memungkinkan analisis terhadap pola-pola kompleks dari banyaknya variabel yang memengaruhi performa mahasiswa [5] dan sistem ini memungkinkan alokasi sumber daya kampus, seperti beasiswa, bimbingan belajar, atau konseling, menjadi lebih efisien dan tepat sasaran. Perancangan solusi yang efektif untuk identifikasi secara sistematis faktor-faktor sebagai penyebab yang berkontribusi terhadap keberhasilan akademik mahasiswa [6]. Keberhasilan akademik menjadi fenomena multidimensional yang dipengaruhi oleh jalinan kompleks dan interdependensi antar faktor inilah yang membuat prediksi keberhasilan akademik menjadi sebuah tantangan besar. Kebutuhan mendesak untuk mewujudkan sistem peringatan dini yang andal sangat dibutuhkan di awal masa studi ketika

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

mahasiswa berada pada fase peralihan dari sekolah menengah ke perguruan tinggi. Pada fase ini, banyak mahasiswa baru yang rentan mengalami *culture shock* akademik [7].

Salah satu solusi yang dapat diterapkan adalah pengembangan sistem prediksi dini berbasis *Machine Learning*. Penerapan ini memungkinkan analisis pola data akademik secara otomatis dan berkelanjutan. Sistem prediktif dapat mengklasifikasikan mahasiswa ke dalam kategori "Tidak Lulus", "Tidak Lulus Tepat Waktu", atau "Lulus Tepat Waktu" [8]. *Machine Learning* dengan model *Random Forest* dipilih sebagai fondasi utama dalam sistem prediksi dini ini karena karakteristiknya yang sangat sesuai dengan permasalahan yang dihadapi [9]. *Random Forest* adalah sebuah algoritma *ensemble learning* yang bekerja dengan membangun banyak algoritma pohon keputusan (*Decision Tree*) pada saat pelatihan dan mengeluarkan *output* berupa kelas (klasifikasi) atau *mean* prediksi (regresi) dari masing-masing pohon tersebut [10]. Dalam implementasinya di Python melalui *library scikit-learn*, model RandomForestClassifier menggunakan algoritma *Classification and Regression Tree* (CART) sebagai *base estimator* untuk membangun setiap pohon keputusan. Dalam penelitian ini, implementasi CART digunakan baik sebagai pembangun pohon pertama dalam *Random Forest* maupun sebagai model pembanding tunggal untuk mengevaluasi peningkatan kinerja yang dihasilkan oleh mekanisme *ensemble* melalui 100 pohon keputusan. Kekuatan utama *Random Forest* terletak pada kemampuannya untuk mengatasi masalah *overfitting* yang sering menjadi kelemahan dari pohon keputusan tunggal [11]. Dengan menggabungkan prediksi dari ratusan atau ribuan pohon yang dilatih pada subset data yang berbeda, algoritma ini mampu menghasilkan model yang sangat stabil dan akurat. Keunggulan lainnya adalah memberikan informasi tentang "*feature importance*", yang paling berpengaruh dalam menentukan prediksi keberhasilan akademik, [12][13]. Data ini kemudian dibagi menjadi dua bagian utama: data pelatihan untuk membangun model dan data pengujian untuk evaluasi akurasi model [14] [15].

Gagasan untuk menggunakan *Machine Learning* dalam memprediksi performa akademik mahasiswa sebenarnya bukanlah hal yang sepenuhnya baru di ranah akademik internasional [16]. Sejumlah penelitian telah dilakukan dengan menggunakan berbagai algoritma seperti Regresi Logistik, *Naive Bayes*, *Support Vector Machine* (SVM), Jaringan Saraf Tiruan (*Neural Network*), dan tentu saja *Random Forest* itu sendiri [17]. Sebagian besar penelitian tersebut memiliki beberapa keterbatasan yang menciptakan celah (*gap*) yang dapat diisi oleh penelitian ini. Pertama, banyak studi yang hanya menggunakan dataset yang relatif kecil dan homogen [18], sehingga generalisasi model terhadap populasi mahasiswa yang lebih luas dan beragam di Indonesia masih perlu diuji. Kedua, variabel yang digunakan dalam penelitian terdahulu seringkali kurang komprehensif [19]. Ketiga, beberapa penelitian hanya berfokus pada satu program studi atau satu fakultas saja. Keempat, aspek interpretabilitas model sering terabaikan; banyak model "*black box*" seperti *neural network* yang memberikan prediksi akurat tetapi sulit dijelaskan mengapa suatu prediksi dihasilkan [20]. Kelima, masih sedikit penelitian yang secara eksplisit merancang sebuah sistem utuh, bukan hanya model prediksi, yang terintegrasi dengan alur kerja bimbingan akademik di kampus. Oleh karena itu, penelitian ini bertujuan untuk mengisi celah tersebut dengan mengembangkan sistem prediksi dini berbasis Random Forest menggunakan algoritma CART untuk dapat diimplementasikan secara praktis sebagai alat bantu bagi institusi pendidikan tinggi.

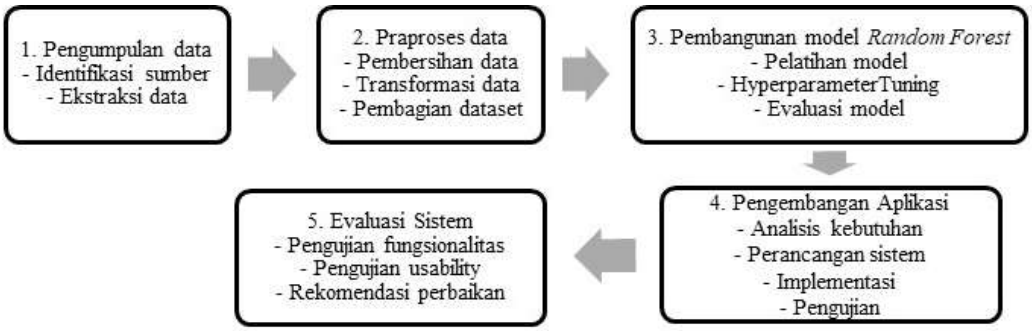
Berdasarkan uraian latar belakang tersebut, diperlukan suatu sistem yang mampu memprediksi keberhasilan akademik mahasiswa secara dini dan akurat. Dengan adanya sistem ini, perguruan tinggi dapat melakukan intervensi berbasis data. Hal ini sejalan dengan tuntutan transformasi digital dalam pendidikan tinggi. Oleh karena itu, penelitian ini diberi judul "Sistem Prediksi Dini Keberhasilan Akademik Mahasiswa Berbasis Random Forest dengan Algoritma CART."

2. METODOLOGI PENELITIAN

2.1 Tahapan Penelitian

Penelitian ini menggunakan pendekatan eksperimental saat membangun dan mengevaluasi algoritma untuk mengetahui seberapa tepat prediksi tersebut. Pendekatan lain yang digunakan adalah pendekatan rekayasa perangkat lunak untuk merancang dan membuat antarmuka pengguna. tujuan penelitian ini menghasilkan dua produk utama, yaitu model prediksi yang sudah diuji kemampuannya dan prototipe aplikasi yang sudah bisa digunakan oleh dosen dan pengelolaan program studi sebagai bantuan dalam mengambil keputusan. Desain penelitian meliputi tahapan secara ringkas, berikut adalah diagram alir keseluruhan tahapan penelitian:

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi



Gambar 1. Alur Penelitian

2.2 Sumber dan Jenis Data

Data yang digunakan dalam penelitian ini merupakan data sekunder yang berdasarkan sistem akademik perguruan tinggi. Tetapi yang diolah berupa dataset yang dibangkitkan menggunakan pendekatan sintesis data berbasis aturan (*rule-based synthetic data generation*) dengan memanfaatkan distribusi statistik dan logika deterministik untuk mensimulasikan pola dunia nyata [21]. Jumlah fitur tidak banyak, karena dinyatakan sebagai fitur yang paling berkorelasi. Data yang digunakan terdiri dari 8 fitur prediktor (X) dan 1 fitur target (Y), sedangkan NPM adalah identitas dari mahasiswa. Adapun jenis data yang digunakan meliputi:

Tabel 1. Fitur/variabel dataset

No	Nama Fitur	Tipe Data	Rentang / Domain	Keterangan
1	NPM	String	100001 - 100141	Id Mahasiswa
2	IP Semester 1	Float	0–4	Indeks Prestasi Semester 1
3	IP Semester 2	Float	0–4	Indeks Prestasi Semester 2
4	IP Semester 3	Float	0–4	Indeks Prestasi Semester 3
5	IP Semester 4	Float	0–4	Indeks Prestasi Semester 4
6	Nilai Hadir	Integer	50–100	Persentase (%)
7	SKS Diambil	Integer	18–24	Total SKS semester berjalan
8	SKS Lulus	Integer	0–24	SKS yang lulus
9	MK Ulang	Integer	0–8	Jumlah mata kuliah ulang
10	Status Keberhasilan	Target	0, 1, 2	0 = Tidak Lulus, 1 = Lulus Tidak Tepat Waktu 2 = Lulus Tepat Waktu

Variabel-variabel di atas berdasarkan hasil pertimbangan korelasi ketersediaan data di lingkungan kampus. IPS per semester digunakan untuk melihat tren performa akademik mahasiswa dari waktu ke waktu. Nilai hadir dipilih karena penelitian terdahulu menunjukkan kehadiran memiliki korelasi positif signifikan dengan prestasi akademik. SKS Diambil dan SKS Lulus sebagai beban studi dan kemampuan mahasiswa dalam menyelesaikan mata kuliah. MK Ulang menjadi indikator penting karena semakin banyak mata kuliah yang diulang, semakin besar risiko keterlambatan kelulusan.

2.2.1 Dataset Generate

Data 8 fitur dibedakan menjadi beberapa bagian, yang terdiri dari:

Tabel 2. Dataset Akademik

NPM	IPS1	IPS2	IPS3	IPS4	Nilai Hadir	SKS Diambil	SKS Lulus	MK ulang	Status
100001	3.45	3.52	3.61	3.58	86.3	21	20	1	2
100002	2.95	2.88	2.92	3.01	73.5	21	18	4	2
100003	2.35	2.28	2.45	2.52	58.3	18	14	3	0
100004	3.95	4	3.88	3.92	91.2	23	22	1	2
100005	2.15	2.08	1.95	2.02	52.1	21	17	3	0
…	…	…	…	…	…	…	…	…	…
100137	3.95	3.98	4	3.96	94.5	23	23	0	2
100138	2.72	2.68	2.75	2.72	63.5	22	20	2	1
100139	3.45	3.52	3.58	3.55	84.5	24	23	1	2
100140	2.42	2.35	2.48	2.45	55.8	20	18	4	0
100141	3.92	3.95	3.98	3.95	93.5	23	23	1	2

3. HASIL DAN PEMBAHASAN

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

3.1 Gambaran Umum Data Penelitian

Penelitian ini menggunakan dataset sebanyak 141 mahasiswa dengan 9 atribut yang terdiri dari 8 variabel numerik (IPS1-IPS4, Nilai Hadir, SKS Diambil, SKS Lulus, MK Ulang), dan 1 variabel target (Status). Rata-rata IPS mahasiswa dari semester 1 hingga 4 berkisar antara 3.44 hingga 3.48, yang menunjukkan bahwa secara umum mahasiswa memiliki performa akademik yang baik. Nilai minimum IPS1 sebesar 2.15 dan maksimum 4.00 menunjukkan variasi kemampuan akademik yang cukup lebar di antara mahasiswa. Rata-rata nilai kehadiran mahasiswa adalah 81.72% dengan standar deviasi 15.05, mengindikasikan tingkat kehadiran yang bervariasi, di mana mahasiswa dengan performa akademik tinggi cenderung memiliki kehadiran yang lebih baik. Rata-rata SKS yang diambil adalah 22.06 dari maksimum 24 SKS per semester, menunjukkan bahwa sebagian besar mahasiswa mengambil beban studi yang normal sesuai dengan kurikulum. Rata-rata SKS yang lulus adalah 20.26, yang berarti rata-rata terdapat sekitar 1.8 SKS yang tidak lulus per semester, dengan rata-rata mata kuliah yang diulang sebanyak 1.55 mata kuliah.

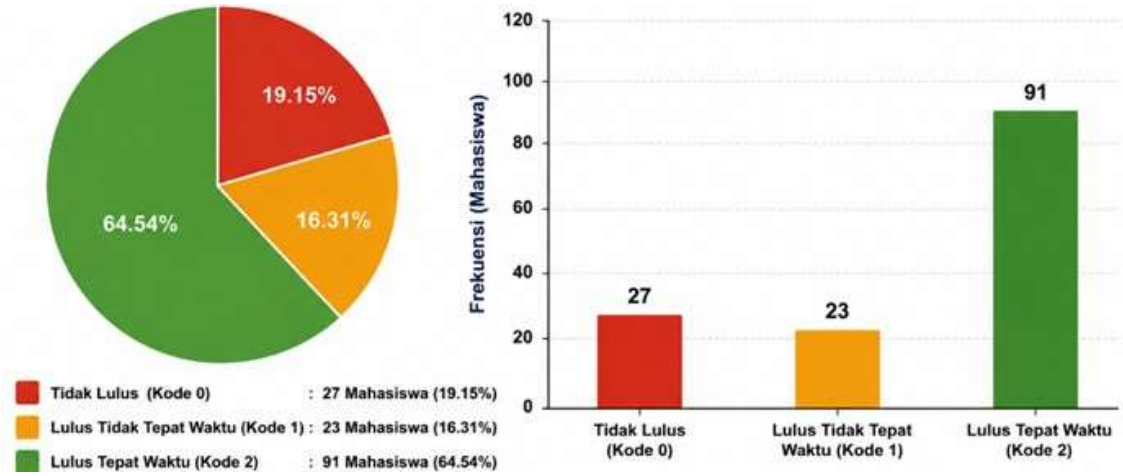
3.1.2 Distribusi Kelas Target Data *Train*

Variabel target Status dikategorikan ke dalam tiga kelas:

Tabel 3. Statistik

Status	Kode	Frekuensi	Persentase
Tidak Lulus	0	27	19.15%
Lulus Tidak Tepat Waktu	1	23	16.31%
Lulus Tepat Waktu	2	91	64.54%
Total		141	100%

Distribusi kelas target menunjukkan bahwa sebagian besar mahasiswa (64.54%) berhasil lulus tepat waktu, yang merupakan kabar baik bagi institusi. Namun, masih terdapat 27 mahasiswa (19.15%) yang mengalami Tidak Lulus dan 23 mahasiswa (16.31%) yang lulus tidak tepat waktu. Secara keseluruhan, terdapat 50 mahasiswa (35.46%) yang mengalami masalah akademik (Tidak Lulus dan lulus tidak tepat waktu), yang merupakan angka yang cukup signifikan dan menjadi perhatian utama bagi pengelola program studi.



Gambar 2. Statistik data latih

Ketidakseimbangan kelas (*imbalanced class*) ini akan menjadi tantangan dalam pemodelan, karena model cenderung lebih akurat dalam memprediksi kelas mayoritas (Lulus Tepat Waktu) dibandingkan kelas minoritas (Tidak Lulus dan Lulus Tidak Tepat Waktu).

3.1.3 Hasil Pra-pemrosesan Data

Pra-pemrosesan data dilakukan melalui beberapa tahapan untuk memastikan kualitas data sebelum digunakan dalam pemodelan *Random Forest*:

Pembersihan Data (*Data Cleaning*)

1. Pengecekan *Missing Values*: Berdasarkan dataset yang digunakan, tidak ditemukan nilai kosong (*missing values*) pada seluruh atribut, sehingga tidak diperlukan imputasi data.
2. Pengecekan Data Duplikat: Tidak ditemukan duplikat data pada NPM atau kombinasi atribut lainnya.
3. Pengecekan *Outliers*: Dilakukan identifikasi *outlier* menggunakan metode IQR (*Interquartile Range*) pada variabel numerik

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

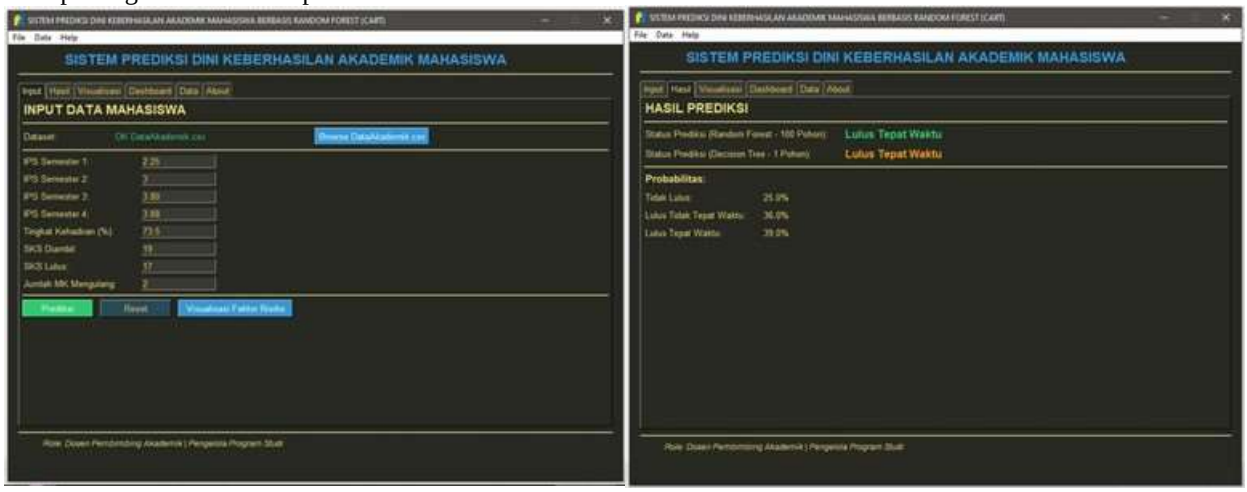
Tabel 4. Dilakukan identifikasi *outlier*

Variabel	Q1	Q3	IQR	Batas Bawah (Q1 - 1.5×IQR)	Batas Atas (Q3 + 1.5×IQR)	*Outlier*
IPS1	2.15	4	1.85	-0.625 = 0	6.775 = 4	0
IPS2	2.08	4	1.92	-0.8 = 0	6.88 = 4	0
IPS3	1.95	4	2.05	-1.125 = 0	7.075 = 4	0
IPS4	2.02	4	1.98	-0.95 = 0	6.97 = 4	0
Nilai Hadir	51.8	100	48.2	-20.5 = 0	172.3	0
SKS Diambil	18	24	6	9	33	0
SKS Lulus	13	24	11	-3.5 = 0	40.5	0
MK Ulang	0	5	5	-7.5 = 0	12.5	0

3.1.4 Hasil Implementasi Sistem Prediksi

3.1.4.1 Input Dataset dan Input Data Uji

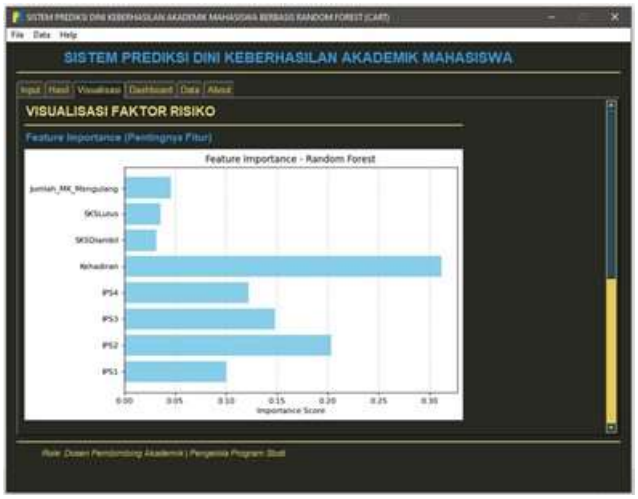
Antarmuka sistem prediksi dini keberhasilan akademik mahasiswa memuat formulir input berkapasitas delapan variabel akademik, meliputi Indeks Prestasi Semester (IPS) 1 hingga 4, SKS diambil, SKS lulus, serta jumlah mata kuliah yang diulang. Pada bagian atas formulir, status pemuatan berkas DataAkademik.csv terverifikasi melalui indikator "OK DataAkademik.csv" beserta opsi pengubahan berkas. Gambar tersebut memperlihatkan sampel data uji dengan parameter: IPS1 = 2.25; IPS2 = 3.00; IPS3 = 3.80; IPS4 = 3.88; tingkat kehadiran = 73,5%; SKS diambil = 19; SKS lulus = 17; dan jumlah mata kuliah mengulang = 2. Setelah pengguna menekan tombol Prediksi, Maka sistem memproses data masukan menggunakan algoritma *Random Forest* (100 *trees*) dan membandingkannya dengan hasil algoritma CART tunggal. Hasil inferensi menunjukkan bahwa kedua model secara konsisten mengklasifikasikan data ke dalam kelas Lulus Tepat Waktu (Status 2). Berdasarkan distribusi probabilitas posterior, kelas Lulus Tepat Waktu memiliki nilai tertinggi sebesar 39,0%, diikuti oleh Lulus Tidak Tepat Waktu (36,0%) dan Tidak Lulus (25,0%). Meskipun selisih probabilitas antarkelas relatif kecil, sistem secara eksplisit menetapkan prediksi akhir berdasarkan nilai probabilitas maksimum (*maximum a posteriori*). Konsistensi keluaran antara model *Random Forest* dan CART mengindikasikan keberadaan pola fitur yang signifikan terhadap kategori kelulusan tepat waktu.



Gambar 3. Antarmuka Input data (kiri) dan Hasil Prediksi (kanan)

Penyajian visualisasi *feature importance* yang dihasilkan oleh model *Random Forest* untuk mengetahui kontribusi setiap variabel terhadap prediksi status keberhasilan mahasiswa. Berdasarkan grafik, variabel Jumlah MK Mengulang memiliki nilai *importance* tertinggi, diikuti oleh SKS Lulus dan SKS Diambil sebagai faktor penting berikutnya. Variabel Kehadiran dan keempat IPS (IPS1 sampai IPS4) memiliki kontribusi yang lebih rendah dibandingkan ketiga variabel utama tersebut. Visualisasi ini juga memperkuat hasil analisis sebelumnya bahwa *Random Forest* mampu mengidentifikasi fitur-fitur paling berpengaruh secara objektif.

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi



Gambar 4. Visualisasi Faktor Resiko

3.1.4.1 Perbandingan Prediksi CART vs *Random Forest*

Pengujian model dilakukan menggunakan 10 data uji yang terdiri dari mahasiswa dengan berbagai karakteristik akademik. Tabel di bawah menyajikan hasil prediksi dari kedua model yang diuji, yaitu CART dengan 1 pohon pertama dan *Random Forest* dengan 100 pohon. Pada NPM 100145 menunjukkan perbedaan hasil, yaitu Cart 1 pohon menghasilkan status 2 dan *Random Forest* (CART 100 pohon) menghasilkan status 1.

Tabel 5. Perbandingan Hasil Prediksi CART dan *Random Forest*

NPM	IPS1	IPS2	IPS3	IPS4	Nilai Hadir	SKSDiambil	SKSLulus	MKUlang	CART 1 tree	RF 100 tree
100142	2.25	3.00	3.80	3.88	73.5	19	17	2	Status 2	Status 2
100143	3.95	4	2.92	3.01	90.8	21	20	4	Status 2	Status 2
100144	2.30	2.00	3.45	3.52	58.3	18	14	4	Status 0	Status 0
100145	2.25	3.00	3.80	3.88	73.5	23	22	1	Status 2	Status 1
100146	3.95	4	2.92	3.01	90.8	21	17	2	Status 2	Status 2
100147	3.28	3.35	3.42	3.38	80.4	22	21	1	Status 2	Status 2
100148	3.88	2.75	2.82	2.91	67.7	19	15	3	Status 1	Status 1
100149	3.95	4	3.88	3.92	91.2	22	21	1	Status 2	Status 2
100150	2.15	2.08	1.95	2.02	52.1	22	19	2	Status 0	Status 0
100151	3.28	3.35	3.42	3.38	80.4	19	17	2	Status 2	Status 2

Perbedaan prediksi ini menunjukkan bahwa *Random Forest* lebih konservatif dalam memberikan prediksi karena mempertimbangkan berbagai sudut pandang dari 100 pohon. Meskipun pohon pertama (CART) melihat IPS3 yang tinggi dan memprediksi Status 2, pohon-pohon lain dalam *Random Forest* mungkin lebih memperhatikan IPS1 yang rendah, sehingga menghasilkan prediksi Status 1. Berdasarkan 10 data uji, kedua model menunjukkan performa yang baik dengan tingkat kesesuaian prediksi sebesar 90% (9 dari 10 data sama). Perbedaan hanya terjadi pada 1 data (NPM 100145) yang memiliki karakteristik akademik yang berada pada batas ambang (*threshold*), yaitu nilai hadir 73.5% yang sedikit di atas 70% dan IPS1 yang rendah dengan IPS4 yang tinggi. Hasil ini sesuai dengan teori bahwa *Random Forest* yang menggabungkan 100 pohon keputusan melalui mekanisme *voting* dianggap mampu menghasilkan prediksi yang lebih akurat dan stabil dibandingkan satu pohon tunggal.

3.2 Pembahasan

Pada sub-bab ini disajikan pembahasan mengenai penerapan algoritma *Classification and Regression Tree* (CART) sebagai dasar pembentukan pohon pertama pada metode *Random Forest* untuk memprediksi status keberhasilan mahasiswa. Analisis dilakukan pada dataset akademik sebanyak 141 mahasiswa dengan 8 fitur numerik dan 1 variabel target yang terdiri dari tiga kelas, yaitu Tidak Berhasil, Berhasil Tidak Tepat Waktu, dan Berhasil Tepat Waktu. Sebelum dilakukan pemodelan, terlebih dahulu dilakukan eksplorasi data untuk memahami karakteristik distribusi setiap variabel serta identifikasi adanya *outlier* menggunakan metode IQR (*Interquartile Range*). Hasil eksplorasi menunjukkan bahwa seluruh fitur berada dalam rentang normal tanpa adanya outlier, sehingga data siap digunakan untuk proses pemodelan.

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

3.2.1 Algoritma CART pada 1 Pohon *Random Forest*

Semua 8 fitur dipertimbangkan sekaligus pada CART untuk setiap *node*, tetapi hanya 1 fitur terbaik yang dipilih untuk melakukan pemisahan (*split*) berdasarkan kriteria *Gini Impurity* (atau *Entropy*). Proses ini berulang secara rekursif hingga satu pohon terbentuk di *Random Forest*. Berikut perhitungan CART untuk satu *node* menggunakan *Gini Impurity*:

$$Gini(t) = 1 - \sum_{i=1}^{c} p_i^2 \quad (1)$$

Dimana c = jumlah kelas dan p_i = proporsi kelas i pada *node* t
Status Keberhasilan dengan kode: 0 = Tidak Lulus (27 mahasiswa), 1 = Lulus Tidak Tepat Waktu (23 mahasiswa), dan 2 = Lulus Tidak Tepat Waktu (91 mahasiswa)

3.2.1.1 Gini *Root*

Gini Root dinyatakan sebagai bentuk penggunaan Indeks *Gini* (*Gini Impurity*) untuk menentukan simpul akar (*root node*) dalam algoritma CART. Dikarenakan pada dataset terdapat 3 kelas maka perhitungan *Gini Root* untuk 3 Kelas:

$p_0 = \frac{27}{141} = 0.19148936$ $p_0 = \frac{23}{141} = 0.16312057$ $p_0 = \frac{91}{141} = 0.64539007$
p(Status 0) = 0.1915 p(Status 1) = 0.1631 p(Status 2) = 0.6454

Hitung $\sum p_i^2$ Sebelum hitung *Gini Impurity*:

$p_0^2 = \left(\frac{27}{141}\right)^2 = \frac{729}{19881}$ $p_1^2 = \left(\frac{23}{141}\right)^2 = \frac{529}{19881}$ $p_2^2 = \left(\frac{91}{141}\right)^2 = \frac{8281}{19881}$

$\sum p_i^2 = \frac{729+529+8281}{19881}$ $\sum p_i^2 = \frac{9539}{19881}$=0.47979357

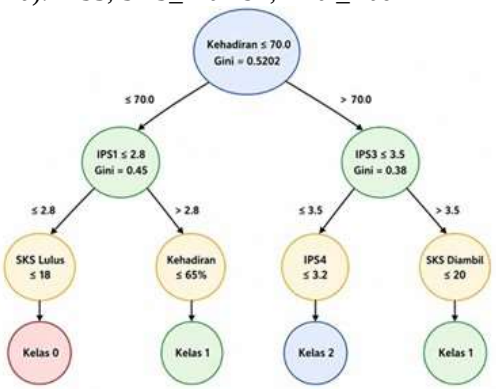
Hitung *Gini Impurity*

Gini = $1 - \sum p_i^2 \rightarrow = 1 - \frac{9531}{19881} \rightarrow = \frac{19881-9531}{19881} \rightarrow = \frac{10341}{19881} \rightarrow$ = 0.5202

3.2.1.3 Pemilihan Fitur Acak untuk *Random Forest*

Berdasarkan dataset yang terdiri dari 141 mahasiswa dengan 9 fitur (setelah menghapus NPM), berikut pemilihan fitur acak untuk 1 pohon.

1. Identifikasi Fitur
   Terdapat 8 fitur numerik (setelah menghapus NPM kategorikal
2. Aturan *Random Forest* untuk Pemilihan Fitur Acak
   Evaluasi sebagian fitur secara acak berdasarkan jumlah fitur yang dipilih, bukan semua fitur.
   Jumlah Fitur yang dipilih per *Node*: $m_{try}=\sqrt{m}$ (klasifikasi) m = 8 → $\sqrt{8} = 2.828 \approx 3$
   Pada *Root node* (Pohon Pertama): → pilih secara acak 3 fitur (tanpa pengembalian): IPS2, Nilai_Hadir, SKS_Lulus
   Pada *Node* berikutnya (Level 2): → pilih 3 fitur acak lagi (bisa berbeda dari *root*)
   Node Kiri (setelah *split* pertama): IPS1, IPS4, MK_Ulang
   Node Kanan (setelah *split* pertama): IPS3, SKS_Diambil, Nilai_Hadir



Gambar 5. Pemilihan Fitur Acak

3.2.1.4 Perhitungan Gini Gain

Sebelum hitung Gini Gain untuk setiap fitur terpilih, maka periksa *root node* yang sudah dipilih 3 fitur acak, yaitu IPS2, Nilai Hadir, dan SKS Lulus. Pemisahan yang dipilih adalah pemisahan yang menghasilkan Gini Gain paling tinggi. Untuk *threshold* ≤ 3.0 dan > 3.0

Tabel 6. *threshold* ≤ 3.0 dan > 3.0

Status	≤ 3.0	>3.0	Jumlah
0	22	5	27
1	12	11	23
2	5	86	91

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

Total	39	102	141

Kelompok Kiri (≤ 3.0) - Total 39 →p(0) = 22/39 = 0.5641, p(1) = 12/39 = 0.3077, p(2) = 5/39 = 0.1282
Kelompok Kanan (> 3.0) - Total 102→p(0) = 5/102 = 0.0490, p(1) = 11/102 = 0.1078, p(2) = 86/102 = 0.8431
Hitung *Gini* untuk Setiap Kelompok

$$Gini = 1 - (p_0^2 + p_1^2 + p_2^2) \quad (2)$$

$Gini_{kiri} = 1 - (0.5641^2 + 0.3077^2 + 0.1282^2) = 0.5707$ dan $Gini_{kanan} = 1 - (0.0490^2 + 0.1078^2 + 0.8431^2) = 0.2752$
Hitung *Gini Split* (*Weighted*)

$$Gini_{split} = \frac{n_{kiri}}{n_{total}} x Gini_{kiri} + \frac{n_{kiri}}{n_{total}} x Gini_{kanan} \quad (3)$$

$Gini_{split} = \frac{39}{141}$ x 0.5707 + $\frac{102}{141}$ x 0.2752, $Gini_{split}$ = 0.1579+0.1991=0.3570
Hitung *Gini Gain*, dimana dari perhitungan sebelumnya *Gini impurity/Gini Root* = 0.5202
$Gini_{Gain} = 0.5202 - 0.3570 = 0.1632$
Pada penelitian ini ditetapkan 5 *threshold* dan penghitungan untuk semua *threshold*.

1. Perhitungan untuk Fitur IPS2
   Data diurutkan berdasarkan IPS2, lalu cari *threshold* terbaik. *Gini Gain* terbaik IPS2 = 0.2702 pada *threshold* 3.00

Tabel 7. *Threshold* IPS2

Threshold IPS2		n 0	n 1	n2	sub	total	Gini Kelompok	Gini Split	Gini Gain
≤ 2.50	Kelompok Kiri	14	0	0	14	141	0.0000	0.3993	0.1209
>2.50	Kelompok Kanan	13	23	91	127		0.4433		
≤ 2.80	Kelompok Kiri	19	9	0	28	141	0.4362	0.3520	0.1682
>2.80	Kelompok Kanan	8	14	91	113		0.3311		
≤ 3.00	Kelompok Kiri	22	12	5	39	141	0.5707	0.3568	0.1633
>3.00	Kelompok Kanan	5	11	86	102		0.2751		
≤ 3.20	Kelompok Kiri	25	14	9	48	141	0.6085	0.3475	0.1727
>3.20	Kelompok Kanan	2	9	82	93		0.2127		
≤ 3.50	Kelompok Kiri	27	17	20	64	141	0.6538	0.3752	0.1450
>3.50	Kelompok Kanan	0	6	71	77		0.1437		

2. Perhitungan untuk Nilai Hadir
   Gini Gain terbaik Nilai Hadir = 0.3002 pada *threshold* 70%.

Tabel 8. *Threshold* Nilai Hadir

Threshold Nilai Hadir		n 0	n 1	n2	sub	total	Gini Kelompok	Gini Split	Gini Gain
<= 60	Kelompok Kiri	15	0	0	15	141	0.0000	0.3896	0.1306
>60	Kelompok Kanan	12	23	91	126		0.4360		
<= 65	Kelompok Kiri	20	7	0	27	141	0.3841	0.3479	0.1723
> 65	Kelompok Kanan	7	16	91	114		0.3393		
<= 70	Kelompok Kiri	25	13	0	38	141	0.4501	0.2745	0.2457
> 70	Kelompok Kanan	2	10	91	103		0.2096		
<= 75	Kelompok Kiri	27	16	6	49	141	0.5748	0.2915	0.2287
>75	Kelompok Kanan	0	7	85	92		0.1406		
<= 80	Kelompok Kiri	27	20	14	61	141	0.6439	0.3195	0.2007
>80	Kelompok Kanan	0	3	77	80		0.0722		

3. Perhitungan untuk Fitur SKS Lulus
   Gini Gain terbaik SKS Lulus = 0.2702 pada *threshold* 19

Tabel 9. *Threshold* SKS Lulus

Threshold Nilai Hadir		n 0	n 1	n2	sub	total	Gini Kelompok	Gini *Split*	Gini Gain
<= 15	Kelompok Kiri	6	2	1	9	141	0.4938	0.4851	0.0351
>15	Kelompok Kanan	21	21	90	132		0.4845		
<= 17	Kelompok Kiri	13	8	4	25	141	0.6016	0.4409	0.0793
> 17	Kelompok Kanan	14	15	87	116		0.4062		
<= 19	Kelompok Kiri	19	9	5	33	141	0.5712	0.3969	0.1233
> 19	Kelompok Kanan	8	14	86	108		0.3436		
<= 21	Kelompok Kiri	25	17	38	80	141	0.6316	0.4597	0.0605
> 21	Kelompok Kanan	2	6	53	61		0.2343		

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

Threshold Nilai Hadir		n 0	n 1	n2	sub	total	Gini Kelompok	Gini *Split*	Gini Gain
<= 23	Kelompok Kiri	27	22	79	128	141	0.5450	0.5079	0.0123
>23	Kelompok Kanan	0	3	77	80		0.1420		

3.2.1.5 Pilih *Split* Terbaik di *Root node*

Root node terbaik di awal proses akan dihitung Indeks *Gini* untuk setiap fitur dalam dataset. *Root node* Yang dipilih adalah fitur dengan *Gini Gain* tertinggi. Gini Gain mengukur seberapa besar penurunan *impurity* setelah dilakukan *split*.

Tabel 10. *Split* Terbaik *Root node*: Nilai Hadir ≤ 70%

Fitur	Gini Gain Terbaik	*Threshold*	Ranking
Kehadiran	0.2457	70%	1 (terbaik)
IPS2	0.1682	2.80	2
SKS Lulus	0.1233	19	3

4.2.1.6 Pembentukan *Node* Anak

Pembentukan *node* anak membagi data dari *parent node* ke dua *child node* menggunakan partisi biner rekursif.
Node Kiri (Kehadiran ≤ 70%): Jumlah data: 38 mahasiswa → status 0 = 25, status 1 = 13, Status 2 tidak ada
Node Kanan (Kehadiran > 70%): Jumlah data: 103 mahasiswa → status 0 = 2, status 1 = 10, status 2 = 91

1. *Node* Kiri (≤ 70%)
   Data *Node* Kiri (38 data): [0 = 25, 1 = 13, 2 = 0] dan Gini *Node* Kiri = 0.4501
   Karena *node* ini tidak murni (masih ada Status 0 dan 1), maka harus melakukan *split* lagi.
   Pemilihan Fitur Acak untuk *Node* Kiri:
   Pilih 3 fitur acak dari 8 fitur→IPS1, MK Ulang, SKS Diambil
   Hasil Perhitungan Gini Gain (setelah dihitung dari data 38): *Split* terbaik di *Node* Kiri: IPS1 ≤ 2.80

   Tabel 11. Gini gain IPS1, MK Ulang, SKS Diambil

Fitur	*Threshold*	Gini Gain
IPS1	≤ 2.80	0.2100
MK Ulang	≤ 2	0.1500
SKS Diambil	≤ 18	0.0900

   a. *Node* Kiri-Kiri (IPS1 ≤ 2.80): jumlah data: 20 mahasiswa dan status 0=18, 1=2, 2=0, Gini = 0.1800. *Node* dapat dikatakan cukup murni → Menjadi Leaf (Prediksi: Status 0 – Tidak Lulus)
   b. *Node* Kiri-Kanan (IPS1 > 2.80): jumlah data: 18 mahasiswa dan status: 0=7, 1=11, 2=0, Gini = 0.4753. *Node* tidak murni → Perlu *split* lagi
2. *Node* Kanan (> 70%)
   Data *Node* Kanan (103 data): [0 = 2, 1 = 10, 2 = 91] dan Gini *Node* Kanan = 0.2096.
   Pemilihan Fitur Acak untuk *Node* Kanan:
   Pilih 3 fitur acak dari 8 fitur → IPS3, IPS4, SKS Lulus
   Hasil Perhitungan *Gini Gain* (setelah dihitung dari data 103)

   Tabel 12. *Gini Gain* IPS3, IPS4, SKS Lulus

Fitur	*Threshold*	Gini Gain
IPS3	≤ 3.50	0.1500
IPS4	≤ 3.20	0.1000
SKS Lulus	≤ 19	0.0500

   a. *Node* Kanan-Kiri (IPS3 ≤ 3.50): jumlah data: 35 mahasiswa dan status: 0=2, 1=8, 2=25, Gini = 0.4000. *Node* tidak murni → Perlu *split* lagi
   b. *Node* Kanan-Kanan (IPS3 > 3.50): jumlah data: 68 mahasiswa dan status: 0=0, 1=2, 2=66, Gini = 0.0576. *Node* cukup murni → Menjadi Leaf (Prediksi: Status 2 – Lulus Tepat Waktu)
3. *Node* Kiri-Kanan (IPS1 > 2.80)
   Data 18 mahasiswa, Status 0=7, 1=11, 2=0, Gini = 0.4753
   Pemilihan Fitur Acak: Nilai Hadir, SKS Lulus, MK Ulang, *Split*: Kehadiran ≤ 65%

   Tabel 13. Nilai Hadir, SKS Lulus, MK Ulang

Fitur	*Threshold*	*Gini Gain*
Kehadiran	≤ 65%	0.1800
SKS Lulus	≤ 17	0.1000
Jumlah MK Mengulang	≤ 2	0.0800

   a. Kiri-Kanan-Kiri (≤ 65%): jumlah 8 data dan status [0 = 6,1 = 2] → Leaf (Prediksi: Status 0)

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

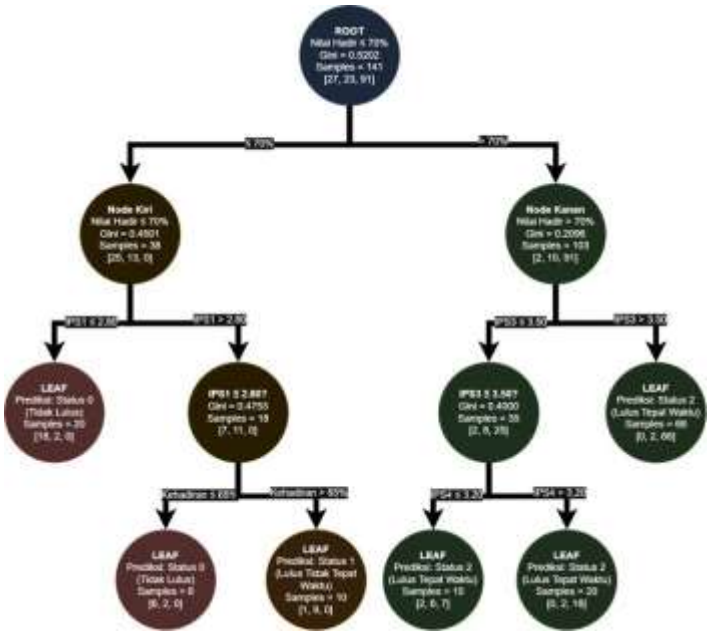
b. Kiri-Kanan-Kanan (> 65%): jumlah 10 data, [0 = 1,1 = 9] → Leaf (Prediksi: Status 1)

4. Lanjutan *Node* Kanan-Kiri (IPS3 ≤ 3.50)
Data: 35 mahasiswa, Status 0=2, 1=8, 2=25, Gini = 0.4000
Pemilihan Fitur Acak: IPS4, SKS Diambil, Nilai Hadir. *Split*: IPS4 ≤ 3.20

Tabel 14. Gini Gain

Fitur	*Threshold*	Gini Gain
IPS4	≤ 3.20	0.1500
SKS Diambil	≤ 20	0.0800
Kehadiran	≤ 75%	0.0700

a. Kiri-Kanan-Kiri (≤ 65%): jumlah 8 data, [0:6,1:2] → Leaf (Prediksi: Status 0)
b. Kiri-Kanan-Kanan (> 65%): jumlah 10 data, [0:1,1:9] → Leaf (Prediksi: Status 1)



Gambar 6. pohon CART pertama untuk *Random Forest* Final

Pohon CART pertama untuk *Random Forest* menghasilkan 6 aturan klasifikasi dengan distribusi data yang cukup baik dan simpan pohon pertama dengan nama tree_1.

Tabel 15. Interpretasi Pohon

Aturan	Prediksi	Jumlah
Nilai Hadir ≤ 70% dan IPS1 ≤ 2.80	Tidak Lulus	20
Nilai Hadir ≤ 70% dan IPS1 > 2.80 dan Kehadiran ≤ 65%	Tidak Lulus	8
Nilai Hadir ≤ 70% dan IPS1 > 2.80 dan Kehadiran > 65%	Lulus Tidak Tepat waktu	10
Nilai Hadir > 70% dan IPS3 ≤ 3.50	Lulus Tepat waktu	35
Nilai Hadir > 70% dan IPS3 > 3.50	Lulus Tepat waktu	68

3.2.2 Pembentukan ke-2 sampai ke-100 *Random Forest*

Semua pohon menggunakan perhitungan dan cara yang sama dan kelas dengan suara terbanyak menjadi prediksi akhir, tetapi pembentukkan pohon memiliki 2 perbedaan utama:

Perbedaan 1: *Bootstrap Sampling* (Data Acak)

Bootstrap sampling adalah proses pengambilan sampel data secara acak dengan pengembalian dari dataset asli.

Perbedaan 2: Pemilihan Fitur Acak di Setiap *Node*

Pada setiap pembentukan *node* dalam pohon, *Random Forest* tidak mengevaluasi seluruh fitur yang tersedia, melainkan hanya memilih sejumlah kecil fitur secara acak (biasanya sebanyak akar kuadrat dari total fitur, atau sekitar 3 dari 8 fitur). Dari subset fitur acak inilah, model kemudian memilih satu fitur dengan *Gini Gain* tertinggi untuk dijadikan sebagai *split* pada *node* tersebut.

Tabel 16. Pemilihan Fitur Acak di Setiap *Node*

Pohon	Fitur di *Root node* (3 fitur acak)
Pohon 1	IPS2, Nilai Hadir, SKS Lulus → Nilai Hadir terpilih
Pohon 2	IPS1, IPS3, SKS Diambil → IPS1 terpilih

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

Pohon 3	IPS4, Kehadiran, MK Ulang → Kehadiran terpilih
...	... (kombinasi acak lainnya)
Pohon 100	IPS2, IPS4, SKS Lulus → IPS2 terpilih

3.2.3 Uji Prediksi

Pengujian terhadap pohon keputusan pertama dilakukan terhadap pohon pertama yang telah selesai disiapkan pada pembentukkan pohon sebelumnya dengan menggunakan dua data uji baru yang memiliki karakteristik berbeda. Untuk menghasilkan prediksi yang lebih *robust* dibutuhkan 100 pohon yang masing-masing memberikan suara (*voting*) terhadap kelas yang diprediksi.

Tabel 17. Data Uji

NPM	IPS1	IPS2	IPS3	IPS4	Nilai Hadir	SKS Diambil	SKS Lulus	MK ulang
100142	2.25	3.00	3.80	3.88	73.5	19	17	2
100143	3.95	4	2.92	3.01	90.8	21	20	4
100144	2.30	2.00	3.45	3.52	58.3	18	14	4
100145	2.25	3.00	3.80	3.88	73.5	23	22	1
100146	3.95	4	2.92	3.01	90.8	21	17	2
100147	3.28	3.35	3.42	3.38	80.4	22	21	1
100148	3.88	2.75	2.82	2.91	67.7	19	15	3
100149	3.95	4	3.88	3.92	91.2	22	21	1
100150	2.15	2.08	1.95	2.02	52.1	22	19	2
100151	3.28	3.35	3.42	3.38	80.4	19	17	2

Penelusuran Pohon untuk mahasiswa NPM 100142:

1. *Root*: Nilai Hadir ≤ 70 % → TIDAK (73.5 > 70)→ Ke KANAN (*Node* Kanan)
2. *Node* Kanan: IPS3 ≤ 3.50 → TIDAK (3.80 > 3.50)→ Ke KANAN (*Leaf* Kanan-Kanan)
3. *Leaf*: Prediksi Status 2 (Lulus Tepat Waktu)

Penelusuran Pohon untuk mahasiswa NPM 100143:

1. *Root*: Nilai Hadir 90.5 ≤ 70% → TIDAK (90.5 > 70) → Ke KANAN (*Node* Kanan)
2. *Node* Kanan: IPS4 3.01 ≤ 3.50 → YA→ Ke KIRI (*Leaf* Kanan-Kiri-Kiri)
3. *Leaf*: Prediksi Status 2 (Lulus Tepat Waktu)

Data uji pertama (NPM 100142) dengan nilai hadir 73.5% dan IPS1 2.25, mengikuti jalur ke kanan karena nilai hadirnya melewati *threshold* 70%, kemudian masuk ke leaf kanan-kanan karena IPS3 (3.80) lebih besar dari 3.50, sehingga diprediksi sebagai Status 2 (Lulus Tepat Waktu). Data uji kedua (NPM 100143) dengan nilai hadir 90.8% dan IPS3 2.92, juga bergerak ke kanan pada *root*, tetapi kemudian belok ke kiri karena IPS3 memenuhi syarat ≤ 3.50, lalu masuk ke leaf kiri-kiri karena IPS4 (3.01) ≤ 3.20, yang juga menghasilkan prediksi Status 2 (Lulus Tepat Waktu). Untuk 8 data uji berikutnya melakukan penelusuran pohon dengan cara yang sama, hasil dapat dilihat pada tabel di bawah

Tabel 18. Hasil Data Uji

NPM	Nilai Hadir	IPS1	IPS3	IPS4	Jalur Pohon	Prediksi 1 Pohon
100142	73.5	2.25	3.80	3.88	*Root* → Kanan → Kanan-Kanan	Status 2
100143	90.8	3.95	2.92	3.01	*Root* → Kanan → Kanan-Kiri → Kiri	Status 2
100144	58.3	2.30	3.45	3.52	*Root* → Kiri → Kiri-Kiri	Status 0
100145	73.5	2.25	3.80	3.88	*Root* → Kanan → Kanan-Kanan	Status 2
100146	90.8	3.95	2.92	3.01	*Root* → Kanan → Kanan-Kiri → Kiri	Status 2
100147	80.4	3.28	3.42	3.38	*Root* → Kanan → Kanan-Kiri → Kanan	Status 2
100148	67.7	3.88	2.82	2.91	*Root* → Kiri → Kiri-Kanan → Kanan	Status 1
100149	91.2	3.95	3.88	3.92	*Root* → Kanan → Kanan-Kanan	Status 2
100150	52.1	2.15	1.95	2.02	*Root* → Kiri → Kiri-Kiri	Status 0
100151	80.4	3.28	3.42	3.38	*Root* → Kanan → Kanan-Kiri → Kanan	Status 2

3.2.3.1 Hasil *Voting* 100 Pohon CART *Random Forest*

Hasil *voting* dari 100 pohon *Random Forest* menunjukkan bahwa mekanisme *ensemble* mampu menghasilkan prediksi yang lebih stabil dibandingkan satu pohon tunggal, dengan tingkat keyakinan rata-rata mencapai 79,4% pada 10 data uji. Terdapat NPM 100145 menunjukkan hasil *voting* selisih hanya 3 suara antara Status 1 dan Status 2.

Tabel 19. Rekapitulasi Voting 100 Pohon Random Forest

NPM	Status 0	Status 1	Status 2	Vote	Prediksi Akhir
100142	3	29	68	68	Status 2

JURNAL SISTEM INFORMASI TGD
Volume 5, Nomor 4, Juli 2026, Hal 913-925
P-ISSN : 2828-1004 ; E-ISSN : 2828-2566
https://ojs.trigunadharma.ac.id/index.php/jsi

100143	0	10	90	90	Status 2
100144	71	16	13	71	Status 0
100145	5	49	46	49	Status 1
100146	3	12	85	85	Status 2
100147	14	11	75	75	Status 2
100148	5	85	10	85	Status 1
100149	1	9	90	90	Status 2
100150	93	7	0	93	Status 0
100151	0	12	88	88	Status 2

3.2.3.2 Analisis Tingkat Keyakinan Prediksi

Tingkat keyakinan (*confidence*) prediksi dihitung berdasarkan persentase suara terbanyak terhadap total 100 pohon. NPM 100145 menunjukkan tingkat keyakinan terendah yaitu 49%, dengan selisih suara yang sangat tipis antara Status 1 (49 suara) dan Status 2 (46 suara). Temuan ini menjelaskan mengapa pada pengujian sebelumnya, CART (1 pohon) memprediksi Status 2, sedangkan *Random Forest* (100 pohon) memprediksi Status 1.

3.4.2 Hasil Akurasi, Presisi, Recall, dan F1-Score

Perhitungan metrik evaluasi dilakukan untuk mengukur kinerja kedua model secara lebih komprehensif.

Tabel 20. Perbandingan Metrik Evaluasi

Metrik	*Random Forest* (100 Pohon)	CART (1 Pohon)
Akurasi	90%	100%
Presisi Makro	0.83	1.00
Recall Makro	0.95	1.00
F1-Score Makro	0.86	1.00

4.5 Analisis *Feature importance*

Analisis *feature importance* dilakukan untuk mengetahui kontribusi masing-masing variabel. Nilai *importance* pada seluruh pohon dalam *Random Forest*. Urutan peringkat Kepentingan Fitur dimulai Jumlah MK Mengulang = 0.28, SKS Lulus = 0.21, SKS Diambil = 0.16, Kehadiran= 0.11, IPS4 = 0.09, IPS3 = 0.07, IPS2 = 0.05, IPS1= 0.03

4. KESIMPULAN

Berdasarkan hasil penelitian dapat disimpulkan bahwa penerapan algoritma CART sebagai *base estimator* dalam *Random Forest* telah berhasil diimplementasikan menggunakan 8 variabel akademik CART dan *Random forest*. Algoritma mampu mengklasifikasikan mahasiswa ke dalam tiga status yaitu Tidak Lulus, Lulus Tidak Tepat Waktu, dan Lulus Tepat Waktu. Pohon CART pertama yang dibangun menghasilkan struktur pohon dengan kedalaman 3 level, di mana *split* terbaik pada *root node* adalah Nilai Hadir ≤ 70% dengan *Gini Gain* sebesar 0,2457 yang menunjukkan bahwa kehadiran merupakan faktor pemisah awal paling efektif. Hasil pengujian pada 10 data uji menunjukkan bahwa CART (1 pohon) menghasilkan akurasi 100% tanpa kesalahan klasifikasi, sedangkan *Random Forest* (100 pohon) menghasilkan akurasi 90%, namun sebenarnya *Random Forest* lebih konservatif dan stabil melalui mekanisme *voting*. Analisis *feature importance* mengungkapkan bahwa Jumlah MK Mengulang merupakan faktor paling dominan dengan nilai importance 0,28, diikuti oleh SKS Lulus (0,21) dan SKS Diambil (0,16), sementara keempat variabel IPS memberikan kontribusi yang relatif kecil dengan IPS1 memiliki pengaruh paling rendah (0,03), yang mengindikasikan bahwa tren peningkatan nilai lebih diperhatikan daripada nilai awal. Sistem prediksi dini yang dikembangkan dengan antarmuka PySimpleGUI telah berhasil mengintegrasikan kedua model dan menyediakan fitur input data, prediksi dengan probabilitas, visualisasi faktor risiko, *dashboard monitoring*, dan ekspor laporan.

UCAPAN TERIMAKASIH

Terima kasih disampaikan kepada pihak-pihak yang telah mendukung terlaksananya penelitian ini.

DAFTAR PUSTAKA

[1] M. Nachouki and M. A. Naaj, “Predicting Student Performance to Improve Academic Advising Using the Random Forest Algorithm,” *Int. J. Distance Educ. Technol.*, vol. 20, no. 1, Feb. 2022, doi: 10.4018/IJDET.296702.

[2] H. Andrianof, A. P. Gusman, and O. A. Putra, “Implementasi Algoritma Random Forest untuk Prediksi Kelulusan Mahasiswa Berdasarkan Data Akademik: Studi Kasus di Perguruan Tinggi Indonesia,” *J. Sains Inform. Terap. E-ISSN*, vol. 4, no. 1, pp. 24–28, 2025.

[3] S. Linawati, S. Nurdiani, K. Handayani, and Latifah, “Prediksi Prestasi Akademik Mahasiswa Menggunakan Algoritma Random Forest dan C4.5,” *J. KHATULISTIWA Inform.*, vol. VIII, no. 1, pp. 47–52, Jun. 2020, [Online]. Available: www.bsi.ac.id

[4] B. Carballo-Mendívil, A. Arellano-González, N. J. Ríos-Vázquez, and M. del P. Lizardi-Duarte, "Predicting Student Dropout from Day One: XGBoost-Based Early Warning System Using Pre-Enrollment Data," *Appl. Sci.*, vol. 15, no. 16, Aug. 2025, doi: 10.3390/app15169202.

[5] J. Kuswanto and L. Hakim, "Penerapan Algoritma Random Forest untuk memprediksi Performa Akademik Mahasiswa," *Decod. J. Pendidik. Teknol. Inf.*, vol. 5, no. 1, pp. 262–270, 2025, doi: 10.51454/decode.v5i1.1103.

[6] F. J. Hamu, D. Wea, and N. Setiyaningtiyas, "Faktor-Faktor yang Memperngaruhi Kinerja Akademik Mahasiswa :Analisis Structural Equation Model," *J. Paedagogy J. Penelit. dan Pengemb. Pendidik.*, vol. 10, no. 1, pp. 175–186, Jan. 2023, [Online]. Available: https://e-journal.undikma.ac.id/index.php/pedagogy/index

[7] A. Muganga, Y. K. Mekonen, M. A. Adarkwah, O. A. Oladipo, C. N. Nweze, and S. Bibi, "Is everywhere I go home? Reflections on the acculturation journey of African international students in China," *Int. J. Intercult. Relations*, vol. 105, no. April 2024, p. 102136, 2025, doi: 10.1016/j.ijintrel.2024.102136.

[8] G. Testiana, "Perancangan Model Prediksi Kelulusan Mahasiswa Tepat Waktu pada UIN Raden Fatah," 2018.

[9] H. Santoso, L. Hakim, A. Afiyati, and B. Jokonowo, "Implementation of the Random Forest algorithm to predict rice needs in DKI Jakarta," *J. Inform. dan Teknol. Inf.*, vol. 22, no. 1, pp. 20–29, 2025, doi: 10.31515/telematika.v22i2.12850.

[10] T. Berhane, T. Melese, and A. M. Seid, "Performance Evaluation of Hybrid Machine Learning Algorithms for Online Lending Credit Risk Prediction," *Appl. Artif. Intell.*, vol. 38, no. 1, 2024, doi: 10.1080/08839514.2024.2358661.

[11] N. Nurussakinah, M. Faisal, and I. B. Santoso, "Algoritma Random Forest dan Synthetic Minority Oversampling Technique (SMOTE) untuk Deteksi Diabetes," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 10, no. 2, pp. 221–234, 2025, doi: 10.14421/jiska.2025.10.2.221-234.

[12] J. A. Ali, M. K. Abdi, T. A. Ali, A. H. Muse, and M. A. Cumar, "Geographic and school-level disparities as primary predictors of numeracy skills: A supervised machine learning approach of Somaliland's national learning assessment," *Soc. Sci. Humanit. Open*, vol. 12, Jan. 2025, doi: 10.1016/j.ssaho.2025.102305.

[13] M. Schonlau and R. Y. Zou, "The random forest algorithm for statistical learning," *Stata J.*, vol. 20, no. 1, pp. 3–29, 2020, doi: 10.1177/1536867X20909688.

[14] N. H. Alfajr, G. Garno, and D. Yusup, "Studi Komparasi Algoritma Random Forest Classifier Dan Support Vector Machine Dalam Prediksi Penyakit Jantung," *J. Inform. dan Tek. Elektro Terap.*, vol. 13, no. 3, pp. 22–30, 2025, doi: 10.23960/jitet.v13i3.6569.

[15] S. A. A. Balabied and H. F. Eid, "Utilizing random forest algorithm for early detection of academic underperformance in open learning environments," *PeerJ Comput. Sci.*, vol. 9, 2023, doi: 10.7717/peerj-cs.1708.

[16] E. Kalita *et al.*, "Educational data mining: a 10-year review," Dec. 01, 2025, *Springer Science and Business Media B.V.* doi: 10.1007/s10791-025-09589-z.

[17] K. Roy and D. M. Farid, "An Adaptive Feature Selection Algorithm for Student Performance Prediction," *IEEE Access*, vol. 12, pp. 75577–75598, 2024, doi: 10.1109/ACCESS.2024.3406252.

[18] M. Aruna *et al.*, "Enhancing safety in surface mine blasting operations with IoT based ground vibration monitoring and prediction system integrated with machine learning," *Sci. Rep.*, vol. 15, no. 1, Dec. 2025, doi: 10.1038/s41598-025-86827-w.

[19] J. He, J. Baileyt, B. I. P. Rubinstein, and R. Zhang, "Identifying at-risk students in massive open online courses," *Proc. Natl. Conf. Artif. Intell.*, vol. 3, pp. 1749–1755, 2015, doi: 10.1609/aaai.v29i1.9471.

[20] R. Syahranita, S. Suhartono, and S. Zaman, "Prediksi Kategori Kelulusan Mahasiswa Menggunakan Metode Regresi Logistik Multinomial," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 8, no. 2, pp. 102–111, 2023, doi: 10.14421/jiska.2023.8.2.102-111.

[21] M. Khalil, F. Vadiee, R. Shakya, and Q. Liu, *Creating Artificial Students that Never Existed: Leveraging Large Language Models and CTGANs for Synthetic Data Generation*, vol. 1, no. 1. Association for Computing Machinery, 2025. doi: 10.1145/3706468.3706523.