

Comparison of Random Forest, Decision Tree, and XGBoost Models in Predicting Student Academic Success

Nurbaeti¹, Neny Sulistyaningsih², Ria Rismayati³

¹ Ilmu Komputer, Fakultas Teknik, Universitas Bumigora, Mataram, 83127, Indonesia

^{2,3} Magister Ilmu Komputer, Program Pascasarjana, Universitas Bumigora, Mataram, 83127, Indonesia

Informasi Artikel

Diterima : 14 June 2025
Revisi : 7 Juli 2025
Publikasi : 31 September 2025

Kata Kunci:

Keberhasilan Akademik Mahasiswa
Machine Learning
Prediksi Pendidikan
Random Forest
Seleksi Fitur

ABSTRAK

Keberhasilan akademik mahasiswa dipengaruhi oleh berbagai faktor akademik dan non-akademik. Machine learning (ML) menawarkan pendekatan efektif dalam memprediksi keberhasilan tersebut melalui analisis data yang kompleks. Namun, sebagian besar penelitian sebelumnya masih terbatas pada prediksi kelulusan dan belum memanfaatkan fitur non-akademik secara optimal, serta jarang menggabungkan teknik seleksi fitur yang berbeda. Penelitian ini bertujuan membandingkan performa tiga algoritma ML Random Forest, Decision Tree, dan XGBoost dalam mengklasifikasikan keberhasilan akademik mahasiswa menggunakan dataset UCI yang mencakup 4424 data dan 37 fitur. Data melalui tahap pembersihan, transformasi label, serta seleksi fitur menggunakan PCA, SelectKBest, dan Variance Threshold. Model dilatih dengan metode holdout (80:20) dan dievaluasi menggunakan akurasi, presisi, recall, dan F1-score. Hasil menunjukkan Random Forest dengan Variance Threshold menghasilkan akurasi tertinggi (0.77) dan F1-score terbaik (0.84) pada kelas mayoritas. XGBoost menyusul dengan akurasi 0.75, sedangkan Decision Tree menunjukkan performa paling rendah. Seluruh model mengalami penurunan performa pada kelas minoritas, menandakan tantangan pada data tidak seimbang. Penelitian ini menunjukkan pentingnya pemilihan algoritma dan seleksi fitur dalam klasifikasi akademik, serta mendorong penggunaan strategi penyeimbangan data untuk mengatasi bias kelas. Temuan ini dapat digunakan oleh institusi pendidikan dalam merancang intervensi berbasis data untuk meningkatkan keberhasilan belajar dan mengurangi angka putus kuliah.

ABSTRACT

Students' academic success is influenced by various academic and non-academic factors. Machine learning (ML) offers an effective approach to predicting academic outcomes by analyzing complex data patterns. However, most previous studies are limited to graduation prediction and rarely incorporate non-academic features or multiple feature selection techniques. This study aims to compare the performance of three ML algorithms Random Forest, Decision Tree, and XGBoost in classifying students' academic success using a dataset from the UCI Machine Learning Repository, consisting of 4424 records and 37 features. The data underwent cleaning, label transformation, and feature selection using PCA, SelectKBest, and Variance Threshold. Models were trained using a holdout method (80% training, 20% testing) and evaluated based on accuracy, precision, recall, and F1-score. The results show that Random Forest with Variance Threshold achieved the highest accuracy (0.77) and F1-score (0.84) on majority classes. XGBoost followed with 0.75 accuracy, while Decision Tree showed the lowest performance. All models struggled to classify the minority class, indicating challenges related to data imbalance. This research highlights the importance of algorithm choice and effective feature selection in academic classification tasks. It also emphasizes the need for data balancing strategies to reduce class bias. The findings can help educational institutions design data-driven interventions to improve learning outcomes and reduce dropout rates.

***Penulis Koresponden**

Email: neny.sulistianingsih@universitasbumigora.ac.id

Cara sitasi IEEE:

Nurbaeti, N. Sulistianingsih, dan R. Rismayanti, "Comparison of Random Forest, Decision Tree, and XGBoost Models in Predicting Students' Academic Success," *Journal of Artificial Intelligence and Software Engineering (J-AISE)*, vol. 5, no. 3, pp. 920-930, September 2025, doi: 10.30811/jaise.v5i3.7138

1. PENDAHULUAN

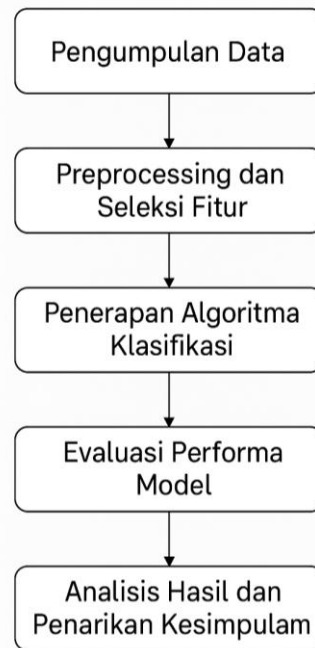
Perkembangan teknologi informasi yang sangat pesat dalam beberapa tahun terakhir telah memberikan dampak besar pada hampir seluruh aspek kehidupan manusia [1]. Kemajuan ini menghasilkan ledakan data dalam jumlah besar yang berasal dari berbagai sektor, seperti industri, ekonomi, ilmu pengetahuan, dan terutama pendidikan [2]. Di sektor pendidikan, transformasi digital ini membuka peluang baru untuk mengevaluasi dan meningkatkan mutu pembelajaran, salah satunya melalui analisis keberhasilan akademik mahasiswa. Keberhasilan akademik sering dijadikan tolok ukur utama dalam menilai kualitas suatu institusi pendidikan [3][4]. Prestasi akademik mahasiswa tidak hanya bergantung pada kapasitas intelektual, melainkan juga dipengaruhi oleh berbagai faktor eksternal dan internal, seperti motivasi belajar, lingkungan keluarga, hingga metode pengajaran yang diterapkan oleh guru [5][6]. Seiring meningkatnya jumlah data yang tersedia di bidang pendidikan, kini terdapat peluang besar untuk memanfaatkan teknologi Machine Learning (ML) guna menganalisis dan memprediksi capaian akademik mahasiswa secara lebih akurat [7].

Berbagai penelitian telah memanfaatkan algoritma ML untuk klasifikasi keberhasilan akademik mahasiswa. Naibaho dan Zahra melaporkan bahwa Random Forest mencapai akurasi tertinggi 99,5%, dibanding XGBoost (99,0%) dan Decision Tree (96,6%) [8]. Astika dan Wibowo juga menunjukkan superioritas Random Forest dan XGBoost dibanding Decision Tree dalam memprediksi performa mahasiswa [9]. Selain itu, studi pada mata kuliah pemrograman lanjut menggunakan Decision Tree mencatat akurasi 96% [10]. Rajendran et al. menggabungkan faktor akademik dan sosial ekonomi, dan Random Forest memperoleh precision 74,94%, recall 75,41%, serta F1-score 75,14% [11]. Asselman et al. menggunakan XGBoost dengan akurasi 78,75% [12], sedangkan Raharjo dan Indriani menemukan Random Forest unggul (96,46%) dibanding XGBoost (87,50%) pada dataset yang lebih kompleks [13]. Machine Learning terbukti efektif dalam mengelola data besar dan kompleks untuk mendukung pengambilan keputusan di bidang pendidikan [14]. Algoritma seperti Random Forest, Decision Tree, dan XGBoost dapat mengidentifikasi mahasiswa berisiko dan merekomendasikan intervensi yang tepat [15]. Dengan pendekatan ini, lembaga pendidikan dapat merancang strategi pembelajaran berbasis data untuk meningkatkan mutu dan prestasi akademik mahasiswa [16], menjadikannya pendekatan menjanjikan dalam peningkatan mutu pendidikan secara menyeluruh [17]. Setiap algoritma memiliki keunggulan tersendiri akurasi, interpretabilitas, dan efisiensi komputasi sehingga penelitian ini bertujuan membandingkan kinerja Random Forest, Decision Tree, dan XGBoost pada data komprehensif yang mencakup status pernikahan, metode pendaftaran, riwayat pendidikan, nilai akademik, kondisi sosial ekonomi, dan faktor pendukung lainnya [18].

Penelitian ini bertujuan untuk membandingkan kinerja tiga algoritma machine learning, yaitu Random Forest, Decision Tree, dan XGBoost, dalam memprediksi keberhasilan akademik mahasiswa berdasarkan data pendidikan yang bersifat komprehensif. Data yang digunakan mencakup berbagai atribut penting, seperti status pernikahan, metode pendaftaran, latar belakang pendidikan sebelumnya, nilai akademik, kondisi sosial ekonomi, serta faktor-faktor lain yang dapat memengaruhi hasil belajar mahasiswa. Melalui analisis perbandingan ini, penelitian diharapkan dapat memberikan kontribusi dalam pengembangan sistem pendukung keputusan berbasis data di bidang pendidikan, serta membantu lembaga pendidikan dalam merancang strategi yang lebih tepat untuk meningkatkan kualitas pembelajaran dan prestasi akademik mahasiswa.

2. METODOLOGI PENELITIAN

Penelitian ini bertujuan untuk mengimplementasikan algoritma Random Forest, Decision Tree, dan XGBoost dalam mengklasifikasi keberhasilan akademik mahasiswa berdasarkan faktor akademik dan non-akademik menggunakan bahasa pemrograman Python. Proses penelitian terbagi menjadi empat tahapan utama, yaitu: (1) Pengumpulan Data, (2) Preprocessing dan Seleksi Fitur, (3) Penerapan Algoritma Klasifikasi, dan (4) Evaluasi Performa Model. Alur penelitian dapat dilihat pada Gambar 1. Metodologi Penelitian.



Gambar 1. Metodologi Penelitian

2.1. Pengumpulan Data

Dataset diperoleh dari UCI Machine Learning Repository dalam format CSV, yang terdiri atas 37 variabel prediktor dan 1 variabel target dengan tiga label kelas: *Graduate*, *Dropout*, dan *Enrolled*. Format CSV dipilih karena kompatibel dengan pustaka Python seperti Pandas dan Scikit-learn, serta mendukung efisiensi dalam pemrosesan data.

2.2. Preprocessing Data

Pada penelitian ini Tahapan ini mencakup pembersihan data (menghapus duplikasi, menangani missing values), transformasi label target menjadi numerik ($\text{Dropout} = 0$, $\text{Graduate} = 1$, $\text{Enrolled} = 2$), dan seleksi fitur untuk meningkatkan efisiensi serta akurasi model. Tiga metode seleksi fitur yang digunakan adalah:

1. PCA (Principal Component Analysis)

Principal Component Analysis (PCA) yaitu metode statistik yang digunakan untuk mengurangi dimensi data (dimensionality reduction) dengan tetap mempertahankan informasi (variasi) yang paling penting bertujuan untuk menyederhanakan dan menghilangkan fitur yang kurang relevan namun tanpa menghilangkan informasi penting dari data set aslinya[18].

$$X_{PCA} = Z \cdot W_k \quad (1)$$

2. SelectKBest (dengan ANOVA F-test)

SelectKBest merupakan algoritme pemilihan fitur yang digunakan untuk meningkatkan akurasi prediksi atau untuk meningkatkan kinerja pada dataset dimensi tinggi. SelectKBest termasuk dalam bagian Univariate Feature selection yang memilih fitur terbaik berdasarkan uji statistik univariat atau uji ANOVA. Uji statistik dapat digunakan untuk memilih fitur yang memiliki hubungan paling kuat dengan variabel output. SelectKBest menghapus semua kecuali fitur yang memiliki skor tertinggi. SelectKBest memilih fitur top K yang memiliki relevansi maksimum dengan variabel target [19].

$$F = \frac{\text{variansi antar kelompok}}{\text{variansi dalam kelompok}} = \frac{\sum_{j=1}^k n_j (\bar{x}_j - \bar{x})^2}{\sum_{j=1}^k \sum_{i=1}^{n_j} (x_{ij} - \bar{x}_j)^2} \quad (2)$$

3. Variance threshold

Variance threshold adalah pendekatan dasar sederhana untuk pemilihan fitur [19]. Variance threshold menghapus semua fitur yang variansinya tidak memenuhi ambang batas tertentu. Secara default, variance threshold menghapus semua fitur varian nol yaitu fitur yang memiliki nilai yang sama di semua sampel. Penelitian ini menggunakan nilai variance threshold 0.1 yang diimplementasikan menggunakan library sklearn [20].

$$\text{Var}(X_i) = \frac{1}{n} \sum_{j=1}^n (x_{ij} - \bar{x}_i)^2 \quad (3)$$

Kedua, dilakukan transformasi pada variabel target dengan mengonversi kategori menjadi nilai numerik: label Graduate diberikan nilai 1, Dropout diberikan nilai 0, dan Enrolled diberikan nilai 2. Transformasi ini bertujuan mempermudah proses pemodelan dengan Tiga algoritma digunakan untuk membangun model klasifikasi yaitu:

1. Random Forest

Random Forest merupakan kumpulan dari beberapa pohon keputusan (decision tree) yang digunakan untuk membuat prediksi dengan memecahkan data menjadi beberapa kategori berdasarkan atribut tertentu dan membuat keputusan berdasarkan perbandingan nilai tertentu [21].

$$\hat{y} = \text{mode}(f_1(x), f_2(x), \dots, f_T(x)) \quad (4)$$

Dengan \hat{y} sebagai prediksi akhir, *mode* merupakan fungsi mayoritas dari semua prediksi pohon, $h_i(x)$ adalah hasil prediksi dari pohon ke i , dan n adalah Jumlah total pohon keputusan dalam ensemble.

2. Decision tree

Decision tree adalah teknik model prediksi yang dapat di gunakan untuk klasifikasi dan prediksi tugas. *Decision tree* menggunakan teknik “Membagi dan menaklukkan” untuk membagi ruang pencarian masalah menjadi himpunan masalah [22].

$$\text{Gini}(D) = 1 - \sum_{i=1}^C p_i^2 \quad (5)$$

Dengan $\text{Gini}(t)$ merupakan nilai Gini impurity pada node t . Simbol C adalah menyatakan jumlah total kelas. Notasi p_i merupakan proporsi data dari kelas ke- i dalam node tersebut. indeks i merupakan 1 menunjukkan bahwa penjumlahan dimulai dari kelas pertama hingga kelas ke- C .

3. XGBoost

XGBoost adalah singkatan dari Extreme Gradient Boosting, yaitu sebuah algoritma machine learning yang berbasis pohon keputusan dan menggunakan pendekatan gradient boosting yang efisien, cepat, dan akurat[23].

$$L^{(t)} = \sum_{i=1}^n l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \Omega(f_t) \quad (6)$$

Dengan persamaan XGBoost di atas menunjukkan fungsi objektif XGBoost pada iterasi ke- t . Bagian pertama adalah jumlah kesalahan prediksi antara nilai sebenarnya y_i dengan prediksi gabungan dari model sebelumnya dan model baru. Bagian kedua, $\Omega(f_t)$, adalah fungsi regularisasi yang memberi penalti pada model agar tidak terlalu kompleks.

Model dibangun menggunakan pustaka *Scikit-learn* dan *XGBoost*. Parameter yang digunakan sebagian besar mengikuti nilai default. *Hyperparameter tuning* tidak dilakukan secara eksplisit, dan hal ini menjadi salah satu keterbatasan penelitian. Pada studi lanjutan, tuning dapat dilakukan menggunakan Grid Search atau Randomized Search untuk mengoptimalkan performa model. Selain itu, validasi dilakukan dengan metode holdout (80% train, 20% test), tanpa teknik *k-fold cross-validation* sehingga kestabilan model belum diuji secara menyeluruh.

2.3. Evaluasi

Evaluasi kinerja model dilakukan menggunakan metrik akurasi, presisi, recall, dan F1-score. Metrik ini digunakan untuk menilai kemampuan algoritma dalam mengklasifikasikan keberhasilan akademik mahasiswa pada tiga kategori target: Dropout (0), Graduate (1), dan Enrolled (2). Proses evaluasi meliputi encoding label, pembagian dataset menjadi training dan testing, pelatihan model menggunakan ketiga algoritma (Random Forest, Decision Tree, dan XGBoost), serta penghitungan metrik evaluasi pada data uji.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (7)$$

Akurasi (Accuracy) merupakan ukuran yang menunjukkan seberapa banyak prediksi yang benar dilakukan oleh model dibandingkan dengan seluruh jumlah data. Akurasi dihitung dengan persamaan di atas di mana TP (True Positive) adalah jumlah data positif yang diklasifikasikan secara benar, TN (True Negative) adalah jumlah data negatif yang diklasifikasikan secara benar, FP (False Positive) adalah jumlah data negatif yang salah diklasifikasikan sebagai positif, dan FN (False Negative) adalah jumlah data positif yang salah diklasifikasikan sebagai negatif. Semakin tinggi nilai akurasi, semakin baik kinerja model dalam mengklasifikasikan data secara umum.

$$Precision = \frac{TP}{TP + FP} \quad (8)$$

Presisi (Precision) adalah ukuran yang menunjukkan proporsi prediksi positif yang benar-benar positif. di mana TP adalah jumlah data positif yang diklasifikasikan secara benar, dan FN adalah jumlah data negatif yang salah diklasifikasikan sebagai positif. Nilai presisi yang tinggi menunjukkan bahwa model jarang membuat kesalahan dalam mengklasifikasikan data sebagai positif.

$$Reccal = \frac{TP}{TP + FN} \quad (9)$$

Recall (juga dikenal sebagai sensitivitas atau True Positive Rate) mengukur seberapa banyak data positif yang berhasil diklasifikasikan dengan benar oleh model dari seluruh data yang benar-benar positif. dengan TP sebagai jumlah data positif yang diklasifikasikan secara benar dan FN sebagai jumlah data positif yang salah diklasifikasikan sebagai negatif. Recall yang tinggi menunjukkan bahwa model memiliki kemampuan yang baik dalam mendeteksi data positif.

$$F1 - score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{10}$$

F1-score merupakan rata-rata harmonik dari presisi dan recall, yang digunakan untuk memberikan gambaran seimbang antara keduanya, terutama ketika terjadi ketidakseimbangan data antar kelas. F1-score dihitung menggunakan rumus . Nilai F1-score yang tinggi menunjukkan bahwa model tidak hanya memiliki presisi yang baik tetapi juga mampu menangkap sebagian besar data positif.

3. HASIL DAN PEMBAHASAN

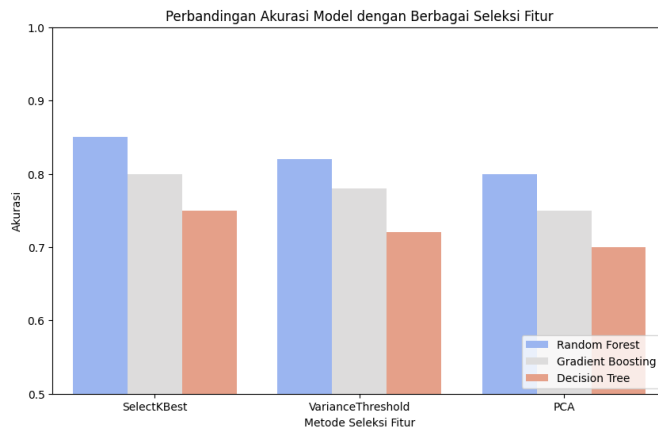
Dalam penelitian ini, dataset yang digunakan diperoleh dari UCI Machine Learning Repository. Dataset tersebut mencakup berbagai variabel yang berhubungan dengan faktor akademik dan non-akademik mahasiswa, seperti Marital status, Application mode, Application order, Course, Previous qualification, Previous qualification (grade), dan target berupa dropout, graduate, atau enrolled. Dataset terdiri dari 4.424 baris data dengan 37 kolom fitur dan 1 kolom target. Data dibagi menjadi 80% untuk training dan 20% untuk testing guna mengevaluasi kemampuan generalisasi model terhadap data baru Pada tahap preprocessing, pertama-tama dilakukan feature selection menggunakan tiga metode utama: Principal Component Analysis (PCA) untuk mereduksi dimensi sambil mempertahankan informasi penting SelectKBest dengan fungsi f_classif untuk memilih fitur berdasarkan skor ANOVA F-value dan Variance Threshold untuk menghapus fitur dengan variansi rendah agar hanya menyertakan atribut yang informatif Kemudian, variabel target dikonversi menjadi nilai numerik: “Graduate” = 1, “Dropout” = 0, dan “Enrolled” = 2, sehingga siap diproses oleh algoritma klasifikasi.

Tabel 2. Hasil seleksi fitur

Feature yang di pilih <i>SelectKBest</i>	Feature yang di pilih <i>Variance threshold</i>	Feature yang di pilih <i>PCA(Principal Component Analysis)</i>
- Tuition fees up to date	- Marital status	- International
- Curricular units 1st sem (approved)	- Application mode	- Nationality
- Curricular units 1st sem (grade)	- Application order	- Admission grade
- Curricular units 2nd sem (approved)	- Course	- Previous qualification (grade)
- Curricular units 2nd sem (grade)	- Previous qualification	- Curricular units 1st sem (grade)
	- Previous qualification (grade)	- Curricular units 2nd sem (grade)
	- Nationality	- Curricular units 1st sem (credited)
	- Admission grade	- Curricular units 1st sem (without evaluations)
	- Displaced	- Curricular units 2nd sem (without evaluations)
	- Educational special needs	- Age at enrollment
	- Debtor	- Course
	- Tuition fees up to date	- Application mode
	- Gender	- Displaced
	- Scholarship holder	- Curricular units 2nd sem (approved)
	- Age at enrollment	
	- International	
	- Curricular units 1st sem (credited)	
	- Curricular units 1st sem (enrolled)	
	- Curricular units 1st sem (evaluations)	
	- Curricular units 1st sem (approved)	
	- Curricular units 1st sem (grade)	
	- Curricular units 1st sem (without evaluations)	
	- Curricular units 2nd sem (credited)	
	- Curricular units 2nd sem (enrolled)	
	- Curricular units 2nd sem (evaluations)	
	- Curricular units 2nd sem (approved)	
	- Curricular units 2nd sem (grade)	
	- Curricular units 2nd sem (without evaluations)	
	- Unemployment rate	
	- Inflation rate	
	- GDP	

Evaluasi model menggunakan metrik akurasi, presisi, recall, dan F1-score untuk ketiga kelas target: 0 (*Dropout*), 1 (*Graduate*), dan 2 (*Enrolled*). Tabel 1 menunjukkan ringkasan performa Random Forest, Decision Tree, dan XGBoost dengan berbagai teknik feature selection. Secara keseluruhan, Variance

Threshold terbukti memberikan kontribusi positif terhadap performa model terutama untuk algoritma kompleks seperti Random Forest dan XGBoost sedangkan Decision Tree paling sensitif terhadap pilihan fitur.



Gambar 2. Perbandingan model dengan berbagai seleksi fitur.

Tabel 2. Hasil evaluasi

Model	Feature Selection	Accuracy	Precision	Recall	F1-Score
Random Forest	PCA	0.730	0	0.846	0.715
			1	0.711	0.940
			2	0.415	0.179
Random Forest	Select K-Best	0.769	0	0.850	0.772
			1	0.769	0.923
			2	0.531	0.338
Random Forest	Variance Threshold	0.773	0	0.848	0.759
			1	0.771	0.945
			2	0.544	0.325
Decision Tree	PCA	0.642	0	0.714	0.633
			1	0.733	0.754
			2	0.303	0.351
Decision Tree	Select K-Best	0.698	0	0.761	0.674
			1	0.778	0.811
			2	0.391	0.437
Decision Tree	Variance Threshold	0.676	0	0.729	0.646
			1	0.763	0.799
			2	0.359	0.397
XGBoost	PCA	0.721	0	0.855	0.690
			1	0.712	0.923
			2	0.386	0.255
XGBoost	Select K-Best	0.759	0	0.850	0.737
			1	0.775	0.916
			2	0.479	0.371
XGBoost	Variance Threshold	0.750	0	0.823	0.737
			1	0.780	0.907
			2	0.448	0.344

Random Forest menunjukkan performa terbaik ketika menggunakan Variance Threshold, dengan Accuracy sebesar 0.77, Precision 0.84, Recall 0.77, dan F1-Score 0.80. Sementara itu, penggunaan PCA dan Select K-Best menghasilkan Accuracy lebih rendah, masing-masing 0.73 dan 0.76.

```

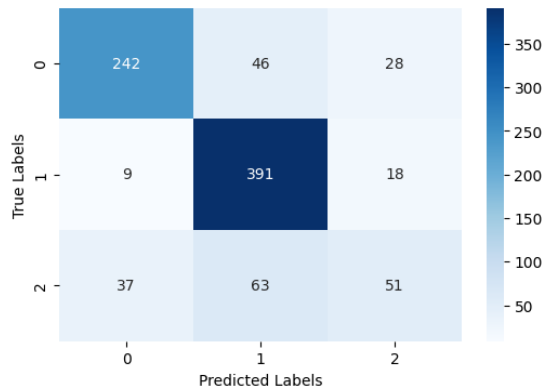
=== Evaluasi Random Forest dengan Fitur Variance Threshold ===
Accuracy Score: 0.773
Classification Report:
      precision    recall  f1-score   support

     0       0.840     0.766     0.801     316
     1       0.782     0.935     0.852     418
     2       0.526     0.338     0.411     151

 accuracy         0.773         885
 macro avg       0.716     0.680     0.688     885
 weighted avg    0.759     0.773     0.759     885

```

Confusion Matrix - Random Forest with Variance Threshold



Gambar 3. Confution matrix random forest variance threshold.

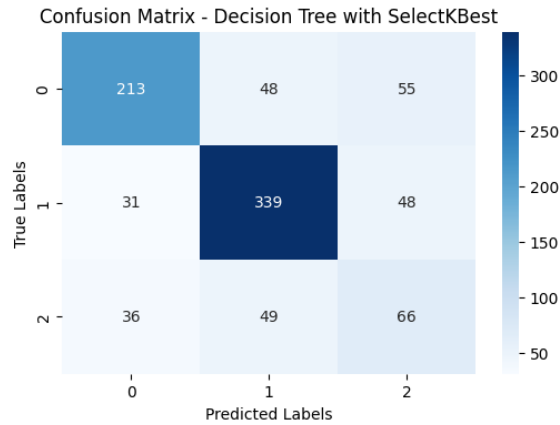
Decision Tree memiliki performa yang lebih rendah dibandingkan dengan *Random Forest* dan XGBoost. Teknik PCA menghasilkan Accuracy 0.64, sedangkan Select K-Best dan Variance Threshold mencapai 0.69 dan 0.67. Hal ini menunjukkan bahwa Decision Tree kurang mampu menangkap pola yang kompleks dibandingkan Random Forest dan XGBoost.

```

=== Evaluasi Decision Tree dengan Fitur SelectKBest ===
Accuracy Score: 0.698
Classification Report:
      precision    recall  f1-score   support

 0       0.761     0.674     0.715     316
 1       0.778     0.811     0.794     418
 2       0.391     0.437     0.412     151

 accuracy         0.698     0.698     0.698     885
 macro avg        0.643     0.641     0.640     885
 weighted avg     0.705     0.698     0.701     885
    
```



Gambar 4. Confusion matrix Decision Tree Select K-Best.

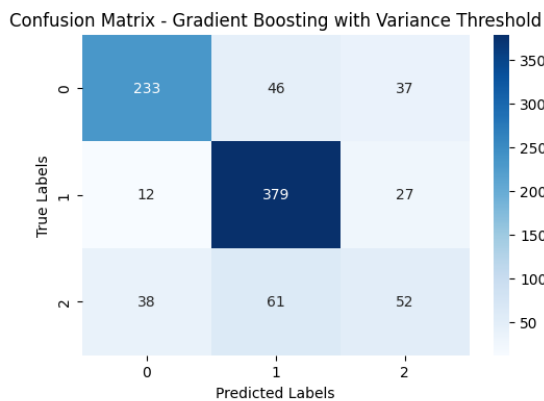
XGBoost memberikan hasil yang lebih baik dibandingkan Decision Tree dan hampir setara dengan Random Forest. Teknik feature selection Variance Threshold menghasilkan Accuracy 0.75, Precision 0.82, Recall 0.74, dan F1-Score 0.78, menunjukkan performa yang cukup stabil. Select K-Best juga memiliki hasil yang sama dengan Accuracy 0.75.

```

=== Evaluasi Gradient Boosting dengan Fitur Variance Threshold ===
Accuracy Score: 0.750
Classification Report:
      precision    recall  f1-score   support

 0       0.823     0.737     0.778     316
 1       0.780     0.907     0.838     418
 2       0.448     0.344     0.390     151

 accuracy         0.750     0.750     0.750     885
 macro avg        0.684     0.663     0.669     885
 weighted avg     0.739     0.750     0.740     885
    
```



Gambar 5. confusion matrix XGBoost variance threshold.

4. KESIMPULAN

Berdasarkan hasil penelitian mengenai klasifikasi keberhasilan akademik mahasiswa menggunakan algoritma machine learning, dapat disimpulkan bahwa Random Forest dan XGBoost menunjukkan performa terbaik, yang dipengaruhi oleh metode seleksi fitur yang digunakan. Random Forest mencapai hasil optimal ketika dikombinasikan dengan Variance Thresholding, sementara XGBoost menunjukkan kinerja terbaik dengan metode PCA dan Select K-Best. Di sisi lain, Decision Tree memiliki performa yang paling rendah dibandingkan kedua algoritma tersebut, baik dari segi akurasi maupun metrik evaluasi lainnya.

Dengan demikian, kombinasi Random Forest dan Variance Thresholding dapat dianggap sebagai pendekatan paling efektif dalam penelitian ini, diikuti oleh XGBoost dengan PCA atau Select K-Best sebagai alternatif yang menjanjikan. Untuk pengembangan lebih lanjut, penelitian ini dapat diperluas dengan menguji algoritma machine learning lainnya atau menggabungkan metode ensemble yang lebih kompleks untuk meningkatkan akurasi prediksi. Selain itu, penerapan model ini dapat diintegrasikan ke dalam sistem pendukung keputusan di lembaga pendidikan guna membantu identifikasi mahasiswa yang memerlukan intervensi sejak dini. Penelitian lanjutan juga dapat mengeksplorasi penggunaan data yang lebih beragam dan dinamis, serta menerapkan teknik feature engineering dan optimasi hyperparameter yang lebih mendalam untuk meningkatkan performa model secara signifikan.

UCAPAN TERIMAKASIH

Penulis mengucapkan terima kasih kepada semua pihak yang telah memberikan dukungan dalam pelaksanaan penelitian ini. Secara khusus, penulis menyampaikan apresiasi kepada Fakultas Teknik, Universitas Bumigora Mataram atas dukungan fasilitas dan sumber daya yang diberikan selama proses penelitian. Penulis juga menyampaikan terima kasih kepada Ibu dan bapak serta teman-teman dan atas bantuan serta dukungan sehingga penulis lebih bersemangat sehingga penelitian ini dapat diselesaikan dengan baik. Tidak lupa, penulis menghargai kontribusi para dosen pembimbing dan rekan sejawat yang telah memberikan masukan dan arahan yang konstruktif selama proses penyusunan jurnal ini.

REFERENSI

- [1] E. F. Agyemang, "ScholarWorks @ UTRGV Predicting Students ' Academic Performance Via Machine Learning Algorithms : An Empirical Review and Practical Application," vol. 15, no. 1, pp. 86–102, 2024.
- [2] Orji Fidelia and vassileva Julita, "c," *Univ. Saskatchewan*, 2022, [Online]. Available: https://www.researchgate.net/publication/364471938_Machine_Learning_Approach_for_Predicting_Students_Academic_Performance_and_Study_Strategies_based_on_their_Motivation
- [3] Y. Chen, J. Sun, J. Wang, L. Zhao, X. Song, and L. Zhai, "Machine Learning-Driven Student Performance Prediction for Enhancing Tiered Instruction," pp. 1–19, 2025, [Online]. Available: <http://arxiv.org/abs/2502.03143>
- [4] N. Sharma, S. Appukutti, U. Garg, J. Mukherjee, and S. Mishra, "Analysis of Student's Academic Performance based on their Time Spent on Extra-Curricular Activities using Machine Learning Techniques," *Int. J. Mod. Educ. Comput. Sci.*, vol. 15, no. 1, pp. 46–57, 2023, doi: 10.5815/ijmecs.2023.01.04.
- [5] G. Airlangga, "A Comparative Analysis of Machine Learning Models for Predicting Student Performance : Evaluating the Impact of Stacking and Traditional Methods," vol. 4, no. 2, pp. 491–499, 2024.
- [6] R. Sharma, S. S. Shrivastava, and A. Sharma, "Predicting Student Performance Using Educational Data Mining and Learning Analytics Technique," *J. Intell. Syst. Internet Things*, vol. 10, no. 2, pp. 24–37, 2023, doi: 10.54216/JISIoT.100203.
- [7] A. I. Gufroni, P. Purwanto, and F. Farikhin, "Academic Performance Prediction Using Supervised Learning Algorithms in University Admission," *Int. J. Informatics Vis.*, vol. 9, no. 1, pp. 184–194, 2025, doi: 10.62527/joiv.9.1.2974.
- [8] D. Andrade-Girón *et al.*, "Predicting Student Dropout based on Machine Learning and Deep Learning: A Systematic Review," *EAI Endorsed Trans. Scalable Inf. Syst.*, vol. 10, no. 5, pp. 1–11, 2023, doi: 10.4108/eetsis.3586.
- [9] R. Rismaya, D. Yuniarto, and D. Setiadi, "Penerapan Algoritma Machine Learning dalam Prediksi Prestasi Akademik Mahasiswa," 2025.
- [10] S. K. Ghosh and F. Janan, "Prediction of student's performance using random forest classifier," *Proc. Int. Conf. Ind. Eng. Oper. Manag.*, pp. 7089–7100, 2021, doi: 10.46254/an11.20211238.
- [11] M. YILDIZ and C. BÖREKÇİ, "Predicting Academic Achievement with Machine Learning Algorithms," *J. Educ. Technol. Online Learn.*, vol. 3, no. 3, pp. 372–392, 2020, doi: 10.31681/jetol.773206.
- [12] U. P. Inyang and E. A. Johnson, "Intelligent Ensemble Learning Framework for Prediction of Students Academic Performance Using Extreme Gradient Boosting and Random Forest Algorithms," vol. 13, no. 3, pp. 1–19, 2025.
- [13] M. R. Borna, H. Saadat, A. T. Hojjati, and E. Akbari, "Analyzing click data with AI: implications for student performance prediction and learning assessment," *Front. Educ.*, vol. 9, no. December, 2024, doi: 10.3389/educ.2024.1421479.
- [14] O. Ojajuni *et al.*, *Predicting Student Academic Performance Using Machine Learning*, vol. 12957 LNCS, no. June 2023. Springer International Publishing, 2021. doi: 10.1007/978-3-030-87013-3_36.

- [15] K. P. Pengkomputeran, I. Media, and T. Mara, "Student Dropout Prediction Using Random Forest and XGBoost Method," vol. 9, no. 1, pp. 147–157, 2025.
- [16] S. Rajendran, S. Chamundeswari, and A. A. Sinha, "Predicting the academic performance of middle- and high-school students using machine learning algorithms," *Soc. Sci. Humanit. Open*, vol. 6, no. 1, 2022, doi: 10.1016/j.ssaho.2022.100357.
- [17] A. Asselman, M. Khaldi, and S. Aammou, "Enhancing the prediction of student performance based on the machine learning XGBoost algorithm," *Interact. Learn. Environ.*, vol. 31, no. 6, pp. 3360–3379, 2023, doi: 10.1080/10494820.2021.1928235.
- [18] H. Kaur, T. Kaur, and R. Garg, "A Prediction Model for Student Academic Performance Using Machine Learning," *Inform.*, vol. 47, no. 1, pp. 97–108, 2023, doi: 10.31449/inf.v47i1.4297.
- [19] L. Amatullah, Y. Widiastiwi, and N. Chamidah, "Penerapan Klasifikasi Random Forest Terhadap Data Gangguan Spektrum Autisme (ASD) Pada Anak-Anak Menggunakan Seleksi Fitur Principal Component Analysis," *Semin. Nas. Mhs. Ilmu Komput. dan Apl.*, vol. 3, no. 2, pp. 356–364, 2022.
- [20] Generosa Lukhayu Pritalia, "Analisis Komparatif Algoritme Machine Learning dan Penanganan Imbalanced Data pada Klasifikasi Kualitas Air Layak Minum," *KONSTELASI Konvergensi Teknol. dan Sist. Inf.*, vol. 2, no. 1, pp. 43–55, 2022, doi: 10.24002/konstelasi.v2i1.5630.
- [21] K. N. Azizah, I. Kurniawan, and F. Nhita, "Model Klasifikasi berbasis Ekspresi Gen Non-Small Cell Lung Carcinoma (NSCLC) pada Wanita Bukan Perokok Menggunakan Metode Ensemble," *Log. J. Penelit. Inform.*, vol. 1, no. 1, p. 1, 2023, doi: 10.25124/logic.v1i1.6389.
- [22] Jan Melvin Ayu Soraya Dachi and Pardomuan Sitompul, "Analisis Perbandingan Algoritma XGBoost dan Algoritma Random Forest Ensemble Learning pada Klasifikasi Keputusan Kredit," *J. Rits. Rumpun Mat. Dan Ilmu Pengetah. Alam*, vol. 2, no. 2, pp. 87–103, 2023, doi: 10.55606/jurrimipa.v2i2.1470.
- [23] Z. Jin, J. Shang, Q. Zhu, C. Ling, W. Xie, and B. Qiang, "RFRSF: Employee Turnover Prediction Based on Random Forests and Survival Analysis," *Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics)*, vol. 12343 LNCS, pp. 503–515, 2020, doi: 10.1007/978-3-030-62008-0_35.
- [24] R. H. Pambudi, B. D. Setiawan, and Indriati, "Penerapan Algoritma C4.5 Untuk Memprediksi Nilai Kelulusan Siswa Sekolah Menengah Berdasarkan Faktor Eksternal," *Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. 7, pp. 2637–2643, 2020, [Online]. Available: <http://j-ptiik.ub.ac.id>
- [25] A. F. B. Sajiwo, B. Rahmat, and A. Junaidi, "Klasifikasi Indeks Standar Pencemaran Udara (Ispu) Menggunakan Algoritma Xgboost Dengan Teknik Imbalanced Data (Smote)," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.4699.