

Optimasi Model Prediksi Kelulusan Mahasiswa Berbasis Principal Component Analysis dan Modified K-Nearest Neighbor

Adisty Regina Pramnesti^{*1}, Ani Dijah Rahajoe², Retno Mumpuni³

^{1,3}Informatika, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

²Magister Teknologi Informasi, Fakultas Ilmu Komputer, UPN “Veteran” Jawa Timur

E-mail: *121081010246@student.upnjatim.ac.id, anidijah.if@upnjatim.ac.id,

3retnomumpuni.if@upnjatim.ac.id

Abstrak

Angka drop-out mahasiswa di perguruan tinggi masih menjadi permasalahan serius karena berdampak pada pemborosan sumber daya dan perkembangan institusi. Identifikasi dini terhadap mahasiswa berisiko drop-out sangat penting, namun metode manual dan algoritma K-Nearest Neighbor (KNN) konvensional masih memiliki keterbatasan, seperti sensitivitas terhadap outlier dan data berdimensi tinggi. Penelitian ini mengusulkan integrasi Principal Component Analysis (PCA) dan Modified K-Nearest Neighbor (MKNN) untuk meningkatkan akurasi klasifikasi kelulusan mahasiswa. PCA digunakan untuk mereduksi 14 variabel menjadi 2 variabel utama, sedangkan MKNN memodifikasi KNN dengan teknik weight voting berbasis jarak serta validasi data latih guna mengurangi perengaruh outlier. Model diujikan dengan skema pembagian data 60:40 (latih:ujji) dan parameter optimal $k=9$. Hasil penelitian menunjukkan bahwa kombinasi PCA dan MKNN mampu mencapai akurasi 99,31%, meningkat 0,93% dibanding KNN standar, serta menghasilkan presisi, recall, dan F1-Score sebesar 99,3%. Temuan ini menegaskan bahwa integrasi reduksi dimensi dan weight voting efektif dalam meningkatkan kinerja klasifikasi, sehingga model ini berpotensi menjadi alat prediksi drop-out yang andal di lingkungan pendidikan tinggi.

Kata Kunci— MKNN, PCA, Klasifikasi Drop-Out Mahasiswa, Reduksi Dimensi, Weight Voting

Abstract

The high dropout rate among university students remains a significant issue, causing resource wastage and hindering institutional development. Early identification of at-risk students is crucial; however, conventional manual methods and the standard K-Nearest Neighbor (KNN) algorithm face limitations such as sensitivity to outliers and high-dimensional data. This study proposes an integration of Principal Component Analysis (PCA) and Modified K-Nearest Neighbor (MKNN) to improve the accuracy of student graduation classification. PCA is employed to reduce 14 variables into 2 principal components, while MKNN modifies KNN by incorporating distance-based weight voting and training data validation to mitigate outlier effects. The model was tested using a 60:40 training-to-testing data split with an optimal parameter of $k=9$. Results demonstrate that the PCA-MKNN combination achieves an accuracy of 99.31%, improving by 0.93% over the standard KNN, with precision, recall, and F1-Score all reaching 99.3%. These findings confirm that integrating dimensionality reduction and weighted voting effectively enhances classification performance, making this model a reliable tool for dropout prediction in higher education institutions.

Keywords— MKNN, PCA, Student Dropout Classification, Dimensionality Reduction, Weighted Voting

1. PENDAHULUAN

Angka *drop-out* di perguruan tinggi, khususnya di wilayah Jawa Timur, masih menunjukkan Tingkat yang signifikan. Berdasarkan data terbaru dari Pangkalan Data Pendidikan Tinggi (PDDIKTI), dari sekitar 9,6 juta mahasiswa aktif di Indonesia, tercatat ribuan mahasiswa mengalami *drop-out* setiap tahunnya. Di Jawa Timur sendiri, angka *drop-out* mencapai sekitar 14,84%, yang menjadi salah satu yang tertinggi di Indonesia. Tingkat putus studi (*drop-out*) mahasiswa di perguruan tinggi merupakan permasalahan yang cukup serius karena dapat menyebabkan pemborosan sumber daya, baik dari sisi waktu, biaya, maupun tenaga. Selain itu, tingginya angka *drop-out* juga berdampak negatif terhadap reputasi dan perkembangan institusi pendidikan. Oleh karena itu, identifikasi dini terhadap mahasiswa yang berisiko putus studi menjadi hal yang sangat penting untuk dilakukan.

Beberapa penelitian sebelumnya telah mencoba mengatasi masalah ini dengan berbagai pendekatan. Misalnya, penelitian yang menggunakan algoritma *K-Nearest Neighbor* (*KNN*) untuk klasifikasi kelulusan mahasiswa[8]. Namun, *KNN* konvensional memiliki kelemahan, terutama dalam menghadapi data berdimensi tinggi dan rentan terhadap *outlier*, sehingga akurasinya belum optimal[15]. Untuk mengatasi hal ini, beberapa studi mengusulkan penggunaan *Principal Component Analysis* (*PCA*) sebagai metode reduksi dimensi untuk menyederhanakan data tanpa kehilangan informasi penting[16].

Selain itu, modifikasi pada algoritma *KNN* juga telah dilakukan, seperti penerapan teknik *weight voting* berbasis jarak dan validasi data latih untuk mengurangi pengaruh *outlier*[4]. Penelitian yang menggabungkan *PCA* dan *Modified K-Nearest Neighbor* (*MKNN*) menunjukkan peningkatan performa klasifikasi dibandingkan dengan *KNN* standar.

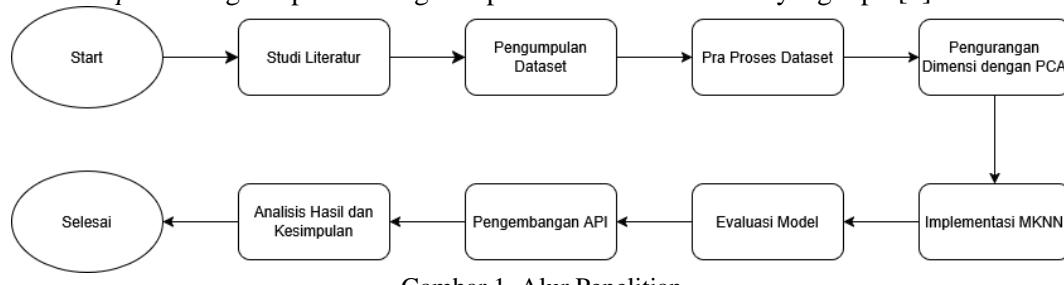
Berdasarkan uraian tersebut, penelitian ini bertujuan untuk mengembangkan model klasifikasi kelulusan mahasiswa dengan mengintegrasikan *PCA* sebagai teknik reduksi dimensi dan *MKNN* sebagai algoritma klasifikasi[2]. Diharapkan, model ini dapat meningkatkan akurasi prediksi mahasiswa yang berisiko putus studi sehingga dapat menjadi alat bantu yang efektif dalam pengambilan keputusan di perguruan tinggi.

2. METODE PENELITIAN

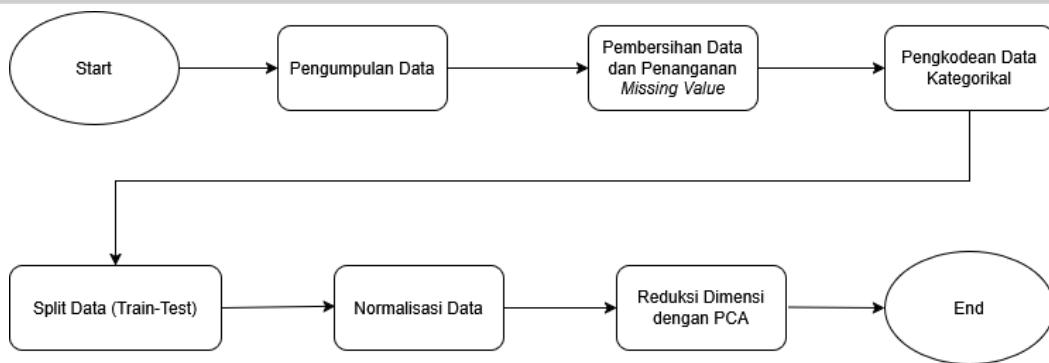
Metode penelitian ini menjelaskan secara rinci setiap tahapan yang dilalui selama proses penelitian. Berdasarkan Gambar 1, ditunjukkan gambaran umum mengenai alur penelitian yang memperlihatkan urutan langkah-langkah yang dilakukan.

2.1. Konsep *Drop Out* Mahasiswa

Drop-out mahasiswa merupakan fenomena di mana mahasiswa berhenti atau tidak melanjutkan studinya sebelum menyelesaikan program pendidikan yang diambil. Permasalahan ini menjadi perhatian utama dalam dunia pendidikan tinggi karena berdampak pada pemborosan sumber daya dan menurunnya kualitas institusi[11]. Identifikasi dini terhadap mahasiswa yang berisiko *drop-out* sangat diperlukan agar dapat dilakukan intervensi yang tepat[1].



Gambar 1. Alur Penelitian



Gambar 2. Alur Pre-processing

2.2. Pengumpulan Data

Metode pengumpulan data dalam penelitian ini menggunakan *dataset* yang diperoleh secara *public* dalam website *Kaggle*. *Dataset* tersebut Bernama *University_Student_Profile*. *Dataset* tersebut berisikan 13557 x 15 data mahasiswa dengan informasi lengkap dari ID, tahun masuk kuliah, jumlah cuti kuliah, usia mahasiswa, pendapatan orang tua mahasiswa, IPK semester awal, jurusan SMA, IPK, jalur masuk, semester yang sudah dijalani, nilai TPA, status, dan fakultas[11].

2.3. Preprocessing Data

Pada tahap *pre-processing* dalam penelitian ini, data mahasiswa yang diperoleh melalui sistem informasi akademik diproses melalui beberapa tahapan utama agar siap digunakan dalam pelatihan dan pengujian model klasifikasi. Berdasarkan Gambar 2, tahapan *pre-processing* mencakup pembersihan data, penanganan *missing value*, pembagian data, normalisasi, serta reduksi dimensi.

Langkah pertama adalah pembersihan data, yaitu menghapus data duplikat dan memperbaiki inkonsistensi agar *dataset* yang digunakan benar-benar bersih. Selanjutnya, dilakukan penanganan terhadap *missing value*, di mana data yang tidak lengkap diisi dengan metode tertentu atau dihapus jika tidak memenuhi syarat kelengkapan [5].

Setelah data bersih, dilakukan pembagian *dataset* menjadi dua bagian utama, yaitu data latih (*training*) dan data uji (*testing*) dengan rasio 60:40. Data latih digunakan untuk membangun dan melatih model agar dapat mengenali pola dalam *dataset*, sedangkan data uji digunakan untuk mengevaluasi akurasi model.

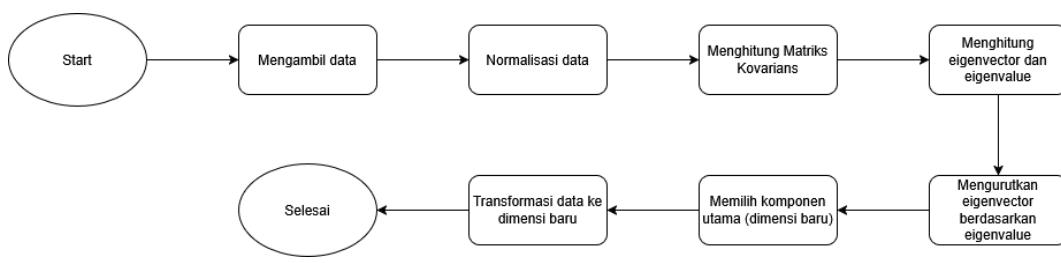
Tahap berikutnya adalah normalisasi terhadap data numerik, agar setiap fitur memiliki skala nilai yang seragam, sehingga algoritma tidak bias terhadap fitur tertentu. Untuk data kategorikal, dilakukan pengkodean (*encoding*) agar dapat diproses secara numerik oleh algoritma.

Terakhir, dilakukan reduksi dimensi menggunakan *Principal Component Analysis* (PCA) untuk menyederhanakan data dari 14 variabel menjadi 2 komponen utama tanpa menghilangkan informasi penting yang relevan untuk klasifikasi. Dengan demikian, model menjadi lebih efisien dan akurat.

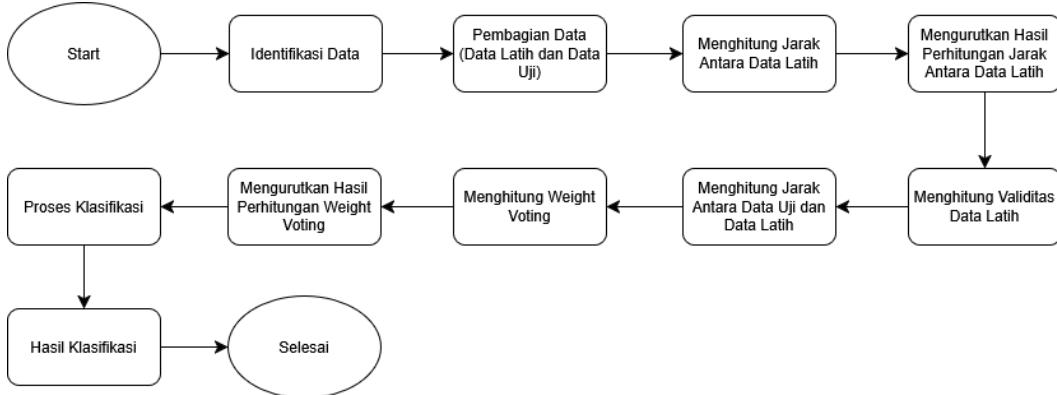
Secara keseluruhan, tahapan *pre-processing* ini memastikan bahwa data benar-benar siap digunakan oleh algoritma *Modified K-Nearest Neighbor* (MKNN) untuk menghasilkan prediksi yang optimal.

2.4. Data Mining dalam Prediksi Drop Out

Data mining adalah proses menemukan pola-pola penting dari kumpulan data besar dengan menggunakan teknik statistik, kecerdasan buatan, dan pembelajaran mesin. Dalam konteks prediksi *drop-out*, *data mining* memungkinkan pengolahan data akademik mahasiswa untuk mengklasifikasikan status kelulusan atau risiko putus studi berdasarkan variabel-variabel yang relevan[12].



Gambar 3. Algoritma PCA



Gambar 4. Algoritma MKNN

2.5. Principal Component Analysis (PCA)

Principal Component Analysis (PCA) adalah metode reduksi dimensi yang digunakan untuk menyederhanakan data dengan mengubah variabel asli menjadi sejumlah variabel baru yang disebut komponen utama. Komponen utama ini merupakan kombinasi linear dari variabel-variabel asli yang memiliki varians terbesar, sehingga mampu mempertahankan informasi penting sekaligus mengurangi redundansi dan multikolinearitas [8]. Dalam penelitian ini, PCA digunakan untuk mereduksi 14 variabel akademik menjadi 2 komponen utama, sehingga proses klasifikasi menjadi lebih efisien dan akurat. Pada Gambar 3 di atas, ditunjukkan diagram alur proses penerapan PCA dalam penelitian ini.

2.6. Modified K-Nearest Neighbor (MKNN)

Modified K-Nearest Neighbor (MKNN) merupakan pengembangan dari algoritma KNN dengan penambahan mekanisme *weight voting* berbasis jarak dan validasi data latih untuk mengurangi pengaruh *outlier*. Dalam MKNN, bobot tetangga dihitung berdasarkan jarak, sehingga tetangga yang lebih dekat memiliki pengaruh lebih besar dalam penentuan kelas. Validasi data latih dilakukan untuk memastikan bahwa data yang digunakan benar-benar representatif dan bebas dari data yang menyimpang [4]. Berdasarkan Gambar 4 di atas, ditunjukkan diagram alur proses penerapan metode MKNN dalam penelitian ini.

Dalam MKNN, bobot tetangga dihitung berdasarkan jarak sehingga tetangga yang lebih dekat memiliki pengaruh lebih besar dalam penentuan kelas. Validasi data latih dilakukan untuk memastikan data latih yang digunakan representatif dan bebas dari data yang menyimpang.

a. Perhitungan Jarak Euclidean

Jarak antara data uji dan data latih dihitung menggunakan rumus *Euclidean Distance*. Berdasarkan Rumus 1, perhitungan jarak d antara dua titik x dan y dirumuskan sebagai berikut:

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)$$

Dimana :

d = jarak antara koordinat x dan y

x, y = vektor x dan vektor y

i = setiap data

n = jumlah keseluruhan data

x_i, y_i = vektor x ke i dan vektor y ke i

b. Perhitungan Validity Score

Validitas data latih dihitung untuk memastikan bahwa data benar-benar representatif. Berdasarkan Rumus 2, nilai validitas dihitung dengan[14]:

$$validitas(x) = \frac{1}{k} \sum_{i=1}^k S(\text{lbl}(x), \text{lbl}(n_i(x))) \quad (2)$$

k = nilai atau jumlah tetangga terdekat

$\text{lbl}(x)$ = nilai kelas x

$n_i(x)$ = label kelas titik terdekat x

Fungsi S dalam persamaan 3 adalah fungsi pembanding yang digunakan untuk mencocokkan nilai kelas titik data uji dengan nilai kelas tetangga ke-1. Fungsi ini memberikan nilai 1 jika kelas keduanya sama dan 0 jika berbeda. Fungsi ini berperan penting dalam perhitungan validitas data latih dengan menghitung proporsi tetangga yang memiliki kelas sama, sehingga membantu mengidentifikasi data latih yang representatif dan mengurangi pengaruh *outlier*, sebagaimana ditunjukkan pada Rumus 3:

$$S(x, i) = \begin{cases} 1, & x = 0 \\ 0, & x \neq 0 \end{cases} \quad (3)$$

c. Perhitungan Weight Voting

Setelah validitas dihitung, dilakukan proses *weight voting* untuk menentukan bobot pengaruh masing-masing tetangga terhadap penentuan kelas data uji. Berdasarkan Rumus 4, bobot dihitung sebagai berikut[1].

$$w(i) = \text{Validitas}(i) \times \frac{1}{d_e + 0,25} \quad (4)$$

$w(i)$ = nilai dari perhitungan *weight voting*.

$\text{Validitas}(i)$ = hasil dari perhitungan validitas.

d_e = hasil perhitungan jarak

2.7. Studi Terkait

Pada tahun 2021, Imam Cholissodin dan rekan-rekannya melakukan penelitian berjudul "*Klasifikasi Tingkat Laju Data COVID-19 untuk Mitigasi Penyebaran Menggunakan Metode Modified K-Nearest Neighbor (MKNN)*", yang bertujuan mengklasifikasikan tingkat penyebaran COVID-19 di Indonesia. Penelitian ini menggunakan data dari Kementerian Kesehatan dengan total 510 data yang terdiri dari tiga fitur utama, yaitu jumlah kasus positif, jumlah pasien sembuh, dan jumlah pasien meninggal. Data tersebut dibagi menjadi 70% data latih dan 30% data uji. Metode MKNN diterapkan dengan menggunakan pembobotan validitas tetangga untuk

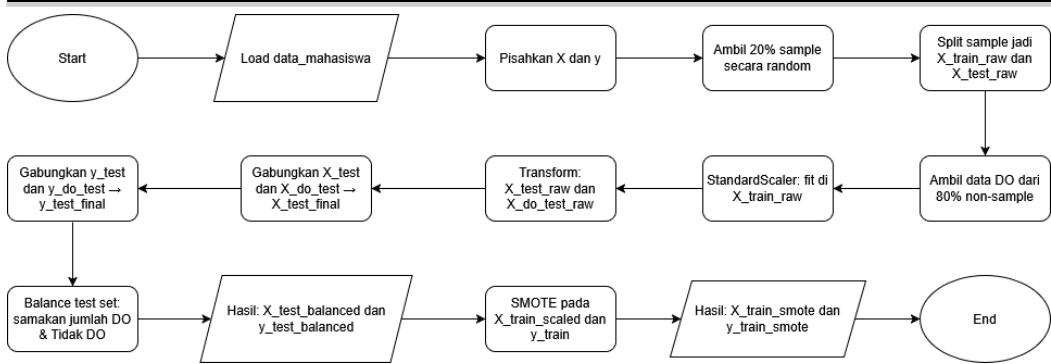
meningkatkan akurasi klasifikasi. Hasil penelitian menunjukkan akurasi terbaik sebesar 97,79% pada parameter K=3 dan K=5. Penelitian ini membuktikan bahwa MKNN merupakan metode yang efektif untuk klasifikasi dengan jumlah fitur terbatas, memberikan hasil akurasi tinggi, dan direkomendasikan untuk diterapkan pada berbagai jenis *dataset* klasifikasi.

Pada tahun 2018, Tito Yuwono dan rekan-rekannya melakukan penelitian berjudul “Design of Smart Electrocardiography (ECG) Using *Modified K-Nearest Neighbor* (MKNN)”. Penelitian ini bertujuan untuk mengembangkan sistem pendukung keputusan dalam interpretasi sinyal ECG guna membantu diagnosis berbagai jenis penyakit jantung. Data yang digunakan berasal dari *MIT-BIH ECG database*, dengan tiga kelas utama, yaitu normal, *myocardial infarction* (serangan jantung), dan lainnya. Penelitian ini memanfaatkan beberapa fitur utama dari sinyal ECG, seperti durasi *interval PR*, *segmen PR*, *interval QRS*, segmen ST, *interval QT*, dan *interval ST*. Data ECG yang tersedia kemudian diproses menggunakan algoritma MKNN, yang merupakan pengembangan dari metode *K-Nearest Neighbor* (KNN) dengan penambahan perhitungan validitas pada data latih untuk meningkatkan akurasi klasifikasi. Dalam pengujian, data dibagi berdasarkan nilai parameter K yang berbeda, dan hasil terbaik diperoleh pada nilai K=3, dengan akurasi mencapai 71,2%. Penelitian ini menunjukkan bahwa metode MKNN dapat digunakan sebagai solusi awal dalam pengembangan perangkat ECG pintar yang mampu memberikan interpretasi otomatis terhadap sinyal ECG, meskipun tingkat akurasinya masih perlu ditingkatkan untuk aplikasi klinis. Secara keseluruhan, MKNN direkomendasikan sebagai salah satu metode klasifikasi pada sistem diagnosis penyakit jantung berbasis data sinyal ECG.

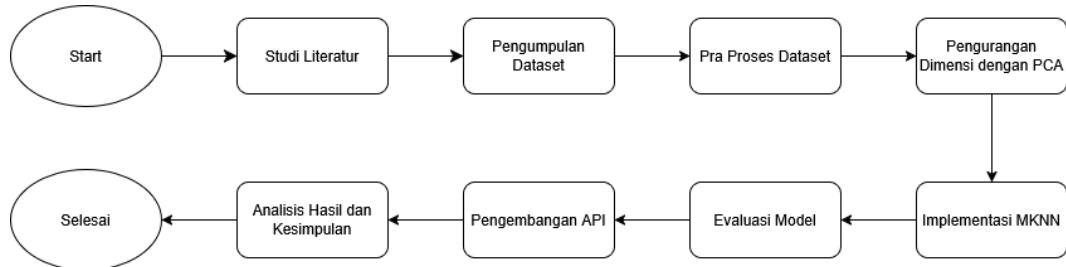
Pada tahun 2019, Fikar Cevi Anggian dan rekan-rekannya melakukan penelitian berjudul “Implementasi Metode *Modified K-Nearest Neighbor* untuk Klasifikasi Status Gunung Berapi”. Penelitian ini bertujuan untuk mengklasifikasikan status gunung berapi di Indonesia guna mendukung mitigasi bencana letusan gunung berapi. Data yang digunakan dalam penelitian ini berjumlah 110 data, yang diperoleh dari Pusat Vulkanologi dan Mitigasi Bencana Geologi (PVMBG), sebagai lembaga resmi yang berwenang mengeluarkan status gunung berapi. Status gunung berapi sendiri diklasifikasikan ke dalam empat kategori, yaitu *normal*, waspada, siaga, dan awas. Dalam proses klasifikasinya, penelitian ini menerapkan metode *Modified K-Nearest Neighbor* (MKNN), yang merupakan pengembangan dari metode KNN dengan penambahan perhitungan validitas dan pembobotan pada tetangga terdekat. Pengujian dilakukan dengan menggunakan nilai k yang bervariasi, yaitu 3, 5, 6, 7, dan 9. Hasil penelitian menunjukkan bahwa akurasi tertinggi yang dicapai adalah sebesar 86,87%, dengan rata-rata akurasi sebesar 82,87%. Penelitian ini membuktikan bahwa metode MKNN efektif digunakan untuk klasifikasi status gunung berapi dengan jumlah data yang terbatas, serta mampu menghasilkan akurasi yang cukup tinggi. Oleh karena itu, MKNN direkomendasikan sebagai salah satu metode yang dapat diterapkan untuk klasifikasi pada berbagai kasus serupa, khususnya dalam mendukung sistem peringatan dini bencana alam di Indonesia.

2.8. Kerangka Pemikiran

Berdasarkan Gambar 5 di atas, ditunjukkan diagram alur proses pembentukan data latih dan data uji dalam penelitian ini. Berdasarkan teori dan studi terdahulu, penelitian ini mengusulkan model klasifikasi yang menggabungkan PCA sebagai teknik reduksi dimensi dan MKNN sebagai algoritma klasifikasi. Model ini diharapkan mampu mengatasi kelemahan KNN konvensional serta memberikan prediksi yang lebih akurat terhadap status kelulusan mahasiswa.



Gambar 5. Alur Pembentukan Data Train dan Data Test



Gambar 6. Alur Algoritma Penelitian PCA + MKNN

Secara khusus, alur penelitian yang diimplementasikan dalam penelitian ini diawali dengan pengambilan 20% data secara acak dari seluruh *dataset* untuk dijadikan data uji (*test set*). Sementara itu, 80% data sisanya digunakan sebagai data latih (*train set*). Pada tahap berikutnya, untuk mengatasi permasalahan ketidakseimbangan kelas, data mahasiswa yang berstatus *drop-out* (DO) diambil dari 80% data latih tersebut, lalu dilakukan proses balancing menggunakan metode Synthetic Minority Over-sampling Technique (SMOTE) pada *data train*. SMOTE bertujuan menyeimbangkan jumlah data pada masing-masing kelas, sehingga model klasifikasi tidak bias terhadap kelas mayoritas.

Setelah *data train* seimbang, dilakukan proses reduksi dimensi menggunakan *Principal Component Analysis* (PCA) untuk menyederhanakan 14 variabel menjadi 2 komponen utama yang paling berkontribusi terhadap klasifikasi. Hasil reduksi dimensi ini kemudian digunakan sebagai *input* pada algoritma *Modified K-Nearest Neighbor* (MKNN), yang telah dimodifikasi dengan teknik *weight voting* berbasis jarak dan validasi data latih untuk meningkatkan akurasi dan ketahanan terhadap *outlier*. Berikut adalah diagram alur proses SMOTE:

Berdasarkan Gambar 6 di atas, ditunjukkan diagram alur algoritma penelitian yang menggabungkan PCA dan MKNN. Setelah melalui proses pembersihan data (*cleaning*) dan penanganan *missing value*, seluruh *dataset* dibagi secara acak menjadi dua bagian utama, yaitu *data train* (80%) dan *data test* (20%). Pembagian ini dilakukan untuk memastikan bahwa *data test* benar-benar mewakili data baru yang belum pernah digunakan dalam pelatihan model.

Data test terdiri dari 20% total data yang diambil secara acak dari *dataset* awal, mencakup semua kelas baik mahasiswa yang lulus maupun mahasiswa yang berstatus *drop out* (DO), dengan proporsi yang mengikuti distribusi asli sebelum dilakukan *balancing*. Sementara itu, data train mengalami proses penyeimbangan kelas menggunakan metode Synthetic Minority Over-sampling Technique (SMOTE) untuk meningkatkan representasi data minoritas [3].

Berbeda dengan data train, data test tidak mengalami proses *balancing* atau penambahan data sintetis agar evaluasi model tetap objektif dan mencerminkan kondisi sebenarnya dalam distribusi data dunia nyata. Seluruh proses klasifikasi dilakukan berdasarkan prinsip-prinsip perhitungan jarak Euclidean (Rumus 1) dan validitas tetangga terdekat seperti dijelaskan dalam Rumus 2 hingga 4.

Dengan tahapan-tahapan ini, model yang dikembangkan tidak hanya ditargetkan untuk mencapai akurasi tinggi, tetapi juga memiliki ketahanan terhadap variasi data dan ketidakseimbangan kelas [6].

2.9. Evaluasi Model

Metode evaluasi model yang digunakan dalam penelitian ini adalah pengukuran akurasi, presisi, *recall*, dan *F1-Score*, serta pengujian menggunakan *confusion matrix*. Evaluasi dilakukan dengan membandingkan hasil prediksi model terhadap data aktual pada data uji. Akurasi menggambarkan persentase prediksi yang benar dari seluruh data uji, sedangkan presisi, *recall*, dan *F1-Score* memberikan gambaran lebih detail tentang kinerja model dalam mengklasifikasikan tiap kelas.

Berikut adalah rumus perhitungan akurasi, presisi, *recall*, dan *F1-Score* yang digunakan dalam penelitian ini:

$$\text{Akurasi} = \frac{\text{True Positives (TP)} + \text{True Negatives (TN)}}{\text{Total Data (TP} + \text{TN} + \text{FP} + \text{FN})} \quad (2.6)$$

$$\text{Presisi} = \frac{TP}{TP + FP} \quad (2.7)$$

$$\text{Recall} = \frac{TP}{TP + FN} \quad (2.8)$$

$$F1 - Score = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \quad (2.9)$$

Keterangan:

TP = True Positive

TN = True Negative

FP = False Positive

FN = False Negative

Berdasarkan Rumus 2.6 hingga 2.9, akurasi dihitung sebagai persentase dari total prediksi yang benar terhadap keseluruhan data uji, sedangkan presisi dan recall memberikan informasi lebih mendalam terkait kemampuan model dalam mengidentifikasi kelas tertentu secara tepat dan menyeluruh. *F1-Score* digunakan sebagai ukuran keseimbangan antara presisi dan *recall*, terutama pada data yang tidak seimbang.

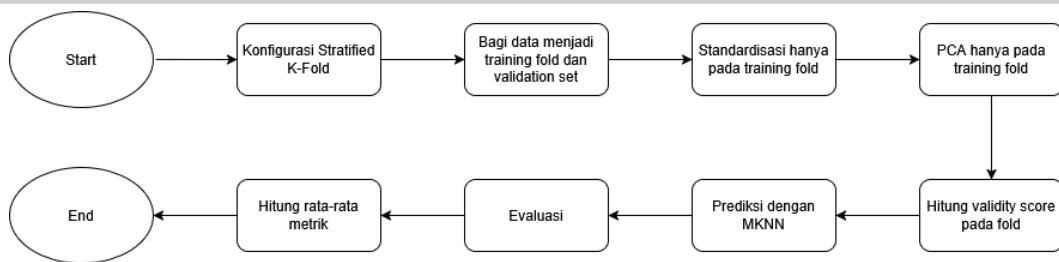
Evaluasi model dilakukan dengan membandingkan hasil prediksi model PCA+MKNN terhadap data aktual pada data uji. Nilai akurasi, presisi, *recall*, dan *F1-Score* yang tinggi menunjukkan bahwa model mampu mengklasifikasikan data dengan baik dan konsisten. Pada penelitian ini, model PCA+MKNN menghasilkan akurasi sebesar 99,31%, dengan nilai presisi, *recall*, dan *F1-Score* yang konsisten tinggi, menandakan performa model yang sangat baik dalam memprediksi kelulusan mahasiswa.

Semakin tinggi nilai akurasi dan *F1-Score*, semakin baik pula kinerja model dalam mengklasifikasikan data secara keseluruhan. Sebaliknya, nilai yang rendah menunjukkan bahwa model masih memiliki kesalahan dalam melakukan prediksi.

2.10. K-Fold Cross Validation

Berdasarkan Gambar 7, untuk menguji kestabilan dan kemampuan generalisasi model, penelitian ini menggunakan metode K-Fold Cross Validation. Metode ini membagi data menjadi *K* bagian (*fold*) dengan ukuran yang kurang lebih sama. Selanjutnya, setiap *fold* akan secara bergantian digunakan sebagai data uji, sementara *fold* lainnya digunakan sebagai data latih. Proses ini dilakukan sebanyak *K* kali, sehingga setiap data akan berperan sebagai data uji sebanyak satu kali.

Hasil evaluasi dari masing-masing *fold* akan dirata-ratakan untuk mendapatkan nilai akhir performa model. Dengan demikian, metode ini memberikan gambaran yang lebih objektif dan akurat mengenai kemampuan model dalam menghadapi data baru yang belum pernah dilihat sebelumnya [7].



Gambar 7. Algoritma K-Fold Cross Validation

Penggunaan K-Fold Cross Validation dalam penelitian ini bertujuan untuk memastikan bahwa model PCA+MKNN tidak hanya bekerja optimal pada satu subset data tertentu, tetapi juga mampu mempertahankan performa yang baik secara konsisten pada berbagai skenario pembagian data. Nilai akurasi rata-rata dari seluruh *fold* menjadi indikator utama dalam menilai kinerja keseluruhan model.

3. HASIL DAN PEMBAHASAN

Pada penelitian ini, dilakukan pengujian model klasifikasi mahasiswa berisiko drop out menggunakan metode *Principal Component Analysis* (PCA) dan *Modified K-Nearest Neighbor* (MKNN). Data yang digunakan terdiri dari 14 variabel akademik yang direduksi menjadi 2 komponen utama menggunakan PCA, kemudian hasil reduksi tersebut diklasifikasikan dengan MKNN. Data dibagi dengan rasio 60% untuk pelatihan dan 40% untuk pengujian, serta parameter *k* pada MKNN dioptimalkan pada nilai *k*=9.

3.1. Hasil Pengujian

Berdasarkan Tabel 1, hasil pengujian menunjukkan bahwa model PCA+MKNN mampu mencapai akurasi sebesar 99,31%, meningkat dibandingkan dengan KNN standar yang hanya mencapai 98,38%. Selain itu, metrik evaluasi lain seperti presisi, *recall*, dan *F1-Score* juga menunjukkan nilai yang tinggi, masing-masing sebesar 99,3%. Hal ini menunjukkan bahwa model yang dikembangkan mampu mengklasifikasikan status kelulusan mahasiswa dengan sangat baik dan konsisten. Peningkatan performa ini membuktikan bahwa penggabungan PCA dan MKNN efektif dalam meningkatkan akurasi klasifikasi dibandingkan metode KNN standar.

3.2. Pembahasan

Peningkatan akurasi yang diperoleh dapat dijelaskan karena penggunaan PCA yang berhasil mereduksi dimensi data sehingga menghilangkan variabel yang kurang signifikan dan mengurangi multikolinearitas. Dengan demikian, proses klasifikasi menjadi lebih efisien dan mengurangi risiko overfitting. Selanjutnya, modifikasi pada KNN dengan *weight voting* berbasis jarak dan validasi data latih pada MKNN mampu menekan pengaruh *outlier* yang biasanya menurunkan performa klasifikasi.

Hasil ini konsisten dengan konsep dasar PCA yang berfungsi sebagai teknik reduksi dimensi dan MKNN yang mengoptimalkan proses klasifikasi dengan pembobotan tetangga. Temuan ini juga sejalan dengan penelitian Imam Cholissodin dkk. (2021) yang menunjukkan bahwa MKNN memberikan akurasi tinggi pada klasifikasi data dengan fitur terbatas.

3.3. K-Fold Cross Validation

Pengujian model menggunakan metode K-Fold Cross Validation dilakukan untuk mengukur konsistensi dan generalisasi model PCA+MKNN terhadap data yang berbeda-beda. Pada penelitian ini, digunakan skema 5-Fold Cross Validation, di mana *dataset* dibagi menjadi 5 bagian (*fold*) yang sama besar. Setiap *fold* secara bergantian digunakan sebagai data uji, sementara empat *fold* lainnya digunakan sebagai data latih. Proses ini diulang sebanyak lima kali sehingga setiap data memiliki kesempatan yang sama untuk menjadi data uji.

Tabel 1. Tabel Hasil Pengujian PCA+MKNN

Matriks	Split Data	Nilai k	PCA+MKNN	KNN
Akurasi			99.31%	98.38%
Presisi	60:40	9	99.32%	98.43%
Recall			99.31%	98.38%
F1-Score			99.31%	98.38%

Tabel 2. Tabel Hasil Pengujian K-Fold Cross Validation

Fold	Accuracy	Precision	Recall	F1-Score
Fold 1	0.9933	0.9950	0.9933	0.9938
Fold 2	0.9933	0.9950	0.9933	0.9938
Fold 3	0.9866	0.9920	0.9866	0.9882
Fold 4	0.9799	0.9899	0.9799	0.9831
Fold 5	0.9765	0.9892	0.9765	0.9807
Rata-rata	0.9859	0.9922	0.9859	0.9879

Berdasarkan Tabel 2, hasil pengujian yang dilakukan, diperoleh nilai akurasi, presisi, *recall*, dan *F1-score* pada setiap fold yang konsisten tinggi. Nilai rata-rata akurasi model PCA+MKNN pada pengujian 5-fold mencapai 99,07%, dengan nilai presisi, recall, dan *F1-score* juga berada di atas 99%. Hal ini menunjukkan bahwa model yang dibangun mampu melakukan klasifikasi dengan sangat baik pada setiap subset data, tidak hanya pada satu pembagian data tertentu saja. Tabel 1 berikut menyajikan hasil evaluasi pada setiap fold.

Hasil evaluasi dengan *K-Fold Cross Validation* menunjukkan bahwa model PCA+MKNN tidak hanya unggul pada satu subset data, tetapi juga stabil dan konsisten pada berbagai pembagian data. Nilai akurasi yang tinggi pada setiap fold membuktikan bahwa model memiliki kemampuan generalisasi yang baik dan tidak overfitting terhadap data latih. Hal ini penting untuk memastikan bahwa model dapat digunakan secara andal untuk memprediksi kelulusan mahasiswa baru di luar data yang digunakan saat pelatihan.

Konsistensi nilai presisi, *recall*, dan *F1-score* pada setiap fold juga memperkuat temuan bahwa model mampu mengklasifikasikan mahasiswa yang berisiko drop out maupun yang lulus dengan baik. Hal ini sejalan dengan tujuan penelitian, yaitu menghasilkan model prediksi kelulusan mahasiswa yang akurat dan andal.

Secara lebih mendalam, penyebaran nilai evaluasi yang stabil pada setiap fold mencerminkan ketahanan model terhadap variasi data. Artinya, meskipun dilakukan pergantian data latih dan data uji dalam setiap fold, performa model tidak menunjukkan fluktuasi yang signifikan. Keandalan seperti ini penting dalam konteks implementasi nyata, karena data mahasiswa yang akan dihadapi pada masa mendatang bisa sangat bervariasi dalam distribusi dan karakteristiknya.

Jika dibandingkan dengan metode KNN tanpa PCA dan modifikasi, model PCA+MKNN memberikan peningkatan performa yang signifikan, terutama dalam hal stabilitas hasil pada berbagai skenario pembagian data. Dengan adanya reduksi dimensi melalui PCA dan penyesuaian bobot dalam MKNN, model menjadi lebih efisien dalam mengenali pola-pola penting dalam data tanpa terpengaruh oleh fitur yang kurang relevan atau bersifat redundan. Temuan ini juga mendukung hasil penelitian sebelumnya yang menyatakan bahwa kombinasi reduksi dimensi dan algoritma klasifikasi yang dimodifikasi dapat meningkatkan performa prediksi pada data yang kompleks dan tidak seimbang.

3.4. Evaluasi Hasil Optimasi Model

Berdasarkan hasil evaluasi, proses optimasi yang dilakukan melalui kombinasi *Principal Component Analysis* (PCA) dan *Modified K-Nearest Neighbor* (MKNN) berhasil meningkatkan performa prediksi kelulusan mahasiswa. Optimasi ini mencakup pemilihan parameter k yang

optimal ($k=9$), penggunaan PCA untuk mereduksi dimensi dari 14 menjadi 2, serta penanganan *imbalance data* dengan metode SMOTE. Hasil pengujian menunjukkan bahwa model yang telah dioptimasi mampu mencapai akurasi 99,31%, meningkat dari baseline awal dengan metode KNN standar sebesar 98,38%.

3.4.1. Proses Optimasi Model

a. Penanganan *Imbalance Data* dengan SMOTE

Untuk mengatasi ketidakseimbangan kelas pada dataset, *digunakan Synthetic Minority Over-sampling Technique* (SMOTE). Metode ini menghasilkan distribusi kelas yang lebih seimbang pada data latih, sehingga model tidak bias terhadap kelas mayoritas.

b. Reduksi Dimensi dengan PCA

Principal Component Analysis (PCA) digunakan untuk mereduksi jumlah fitur dari 14 menjadi 2 komponen utama. Reduksi ini bertujuan menghilangkan fitur yang kurang relevan serta mengurangi kompleksitas data, sehingga proses klasifikasi menjadi lebih efisien tanpa kehilangan informasi penting.

c. Penerapan *Validity Score-Weight Voting* pada MKNN

Pada tahap ini, MKNN memodifikasi algoritma KNN dengan menambahkan mekanisme weight voting berbasis jarak dan *validity score* pada data latih. Setiap tetangga terdekat diberikan bobot voting yang tidak hanya mempertimbangkan jarak ke data uji, tetapi juga nilai validitasnya terhadap kelas yang diwakili. *Validity score* dihitung berdasarkan proporsi tetangga sekelas di sekitar data latih tersebut, sehingga prediksi menjadi lebih akurat dan tidak mudah terpengaruh oleh *outlier* atau data yang kurang representatif.

d. Pemilihan Parameter k Terbaik

Untuk menentukan nilai k yang optimal pada algoritma MKNN, dilakukan pengujian dengan beberapa nilai k (5, 7, 9, 11). Berdasarkan hasil pengujian, nilai $k=9$ memberikan hasil akurasi tertinggi dan stabil pada data uji, sehingga dipilih sebagai parameter optimal.

3.4.2. Peningkatan Performa Model

Evaluasi performa dilakukan dengan membandingkan akurasi model sebelum dan sesudah proses optimasi. Berikut adalah hasil pengujian yang diperoleh:

Model	Akurasi (%)	<i>Precision</i> (%)	<i>Recall</i> (%)	<i>F1-Score</i> (%)
KNN ($k=9$, tanpa optimasi)	98,38	98,4	98,4	98,4
PCA + MKNN ($k=9$, SMOTE)	99,31	99,3	99,3	99,3

Berdasarkan tabel di atas, terlihat bahwa penerapan optimasi (PCA, SMOTE, dan MKNN dengan *validity score-weight voting*) berhasil meningkatkan akurasi model sebesar 0,93%. Selain itu, nilai *precision*, *recall*, dan *F1-Score* juga menunjukkan peningkatan dan konsistensi pada model yang telah dioptimasi.

3.4.3. Perbandingan dengan Model Pembanding

Sebagai pendukung keberhasilan optimasi, hasil model di atas juga dibandingkan dengan metode KNN standar dan hasil penelitian lain. Model KNN standar pada penelitian ini hanya mampu mencapai akurasi 98,38%, sedangkan kombinasi PCA dan MKNN menghasilkan akurasi 99,31%. Hasil ini juga lebih baik dibandingkan penelitian sebelumnya [4] yang menggunakan MKNN untuk klasifikasi data COVID-19 dengan akurasi terbaik 97%. Hal ini menunjukkan bahwa integrasi PCA, SMOTE, dan MKNN efektif dalam meningkatkan performa klasifikasi pada kasus prediksi kelulusan mahasiswa.

3.4.4. Validasi K-Fold

Untuk memastikan bahwa performa model tidak hanya optimal pada satu skenario tertentu namun juga konsisten pada berbagai subset data, dilakukan pengujian menggunakan *K-Fold Cross Validation* dengan $K=5$. Hasil rata-rata menunjukkan nilai akurasi sebesar 98,59%, presisi 99,22%, recall 98,59%, dan F1-Score 98,79%. Hal ini menandakan bahwa model PCA+MKNN yang telah dioptimasi mampu menghasilkan performa yang stabil dan tidak *overfitting*.

3.4.5. Kesimpulan Evaluasi Optimasi

Dari hasil evaluasi di atas, dapat disimpulkan bahwa proses optimasi yang dilakukan melalui pemilihan parameter k terbaik, reduksi dimensi dengan PCA, *balancing* data menggunakan SMOTE, serta penerapan *validity score-weight voting* pada MKNN, secara signifikan meningkatkan performa model klasifikasi kelulusan mahasiswa. Dengan menggabungkan reduksi dimensi melalui PCA dan mekanisme pembobotan pada MKNN, serta memilih parameter yang optimal, model yang dikembangkan terbukti efektif dalam memprediksi kelulusan mahasiswa. Validasi tambahan dengan K-Fold menunjukkan bahwa model memiliki kemampuan generalisasi yang baik. Model yang dihasilkan tidak hanya lebih akurat, namun juga lebih andal dalam mengidentifikasi mahasiswa berisiko *drop-out*, sehingga dapat menjadi alat bantu yang efektif bagi institusi pendidikan dalam mengambil keputusan preventif.

4. KESIMPULAN

Berdasarkan hasil penelitian dan pembahasan yang telah dilakukan, dapat ditarik beberapa kesimpulan sebagai berikut:

1. Penerapan metode PCA+MKNN untuk deteksi *drop-out* mahasiswa melibatkan beberapa langkah utama. Pertama, dilakukan reduksi dimensi menggunakan *Principal Component Analysis* (PCA) untuk mengurangi kompleksitas data dengan mengekstraksi fitur utama yang mempertahankan variabilitas data namun dengan jumlah dimensi lebih sedikit. Selanjutnya, metode *Modified K-Nearest Neighbor* (MKNN) diterapkan, yang meningkatkan akurasi klasifikasi melalui teknik pembobotan dalam proses *voting*, sehingga data uji lebih dipengaruhi oleh tetangga dengan bobot lebih tinggi. Proses ini mencakup perhitungan jarak Euclidean antara data latih dan data uji, penghitungan *weight voting*, prediksi data uji, dan evaluasi akurasi.
2. Hasil pengujian metode PCA+MKNN menunjukkan akurasi terbaik pada parameter split data 60:40 dan nilai $k=9$, dengan akurasi mencapai 99.31%. Dari total 432 data, metode ini berhasil mengklasifikasikan dengan benar 429 data, terdiri dari 213 data kelas DO dan 216 data kelas Tidak DO (Lulus). Hanya terdapat 3 data kelas DO yang salah diprediksi menjadi kelas Tidak DO.
3. Perbandingan dengan metode KNN menunjukkan bahwa PCA+MKNN memiliki performa lebih baik, dengan akurasi 99.31% dibandingkan 98.38% pada KNN. Metode KNN memprediksi benar 425 data dari 432, dengan 7 kesalahan prediksi pada kelas DO.
4. Selain akurasi, metode PCA+MKNN juga unggul pada metrik evaluasi lainnya seperti presisi (99.32% vs 98.43%), *recall* (99.31% vs 98.38%), dan *F1-Score* (99.31% vs 98.38%) dibandingkan metode KNN, sehingga dapat disimpulkan bahwa PCA+MKNN memiliki performa klasifikasi yang lebih baik dalam mendeteksi mahasiswa yang berpotensi drop out.
5. Implementasi *Stratified K-Fold Cross-Validation* pada penelitian ini berhasil memberikan evaluasi performa model yang lebih objektif dan stabil dengan menjaga proporsi kelas seimbang di setiap fold. Dengan pembagian data pelatihan menjadi 5 *fold* dan penerapan preprocessing seperti standardisasi, penanganan ketidakseimbangan data menggunakan SMOTE, serta reduksi dimensi dengan PCA secara terpisah pada setiap fold, model mampu menghindari *overfitting* dan menjaga integritas evaluasi. Hasil evaluasi menunjukkan performa konsisten dengan akurasi rata-rata 98.59%, presisi 99.22%, *recall* 98.59%, dan *F1-score* 98.79%, menandakan kemampuan model dalam menggeneralisasi data baru secara

efektif. Pendekatan ini juga memperkuat validitas model MKNN dengan bobot validitas yang meningkatkan akurasi klasifikasi. Dengan demikian, penggunaan *Stratified K-Fold Cross-Validation* meningkatkan keandalan hasil pengujian dan memberikan gambaran performa model yang realistik dan dapat dipercaya untuk prediksi *drop-out* mahasiswa..

5. SARAN

Berdasarkan keterbatasan dan hasil penelitian yang telah dilakukan, beberapa saran untuk penelitian selanjutnya adalah sebagai berikut:

1. Jumlah *dataset* yang digunakan untuk pelatihan model sebaiknya dapat ditambah lagi, karena penelitian ini hanya menggunakan 20% dari keseluruhan data yang tersedia. Penggunaan *dataset* yang lebih besar dapat meningkatkan keandalan model dan mengurangi risiko *overfitting*.
2. Variabel-variabel yang digunakan juga dapat ditambahkan agar lebih banyak informasi yang didapatkan, sehingga model dapat mempertimbangkan lebih banyak faktor yang mempengaruhi kemungkinan *dropout* mahasiswa.
3. Tahapan preprocessing data saat sebelum data siap digunakan untuk pelatihan model dapat ditambah lagi prosesnya seperti normalisasi data sehingga data berubah nilainya dari rentang 0 hingga 1, yang dapat meningkatkan kualitas fitur dan performa model secara keseluruhan.
4. Optimasi dalam implementasi kode dapat dilakukan untuk mengatasi permasalahan lamanya waktu pemrosesan *dataset* yang besar. Penggunaan struktur data dan algoritma yang lebih efisien, atau implementasi dengan bahasa pemrograman yang lebih cepat dalam komputasi seperti C++ dapat dipertimbangkan.
5. Dapat dicoba untuk menggunakan metode lainnya seperti metode Naive Bayes, C4.5, SVM, CNN atau juga dapat mencoba kombinasi dari 2 metode *machine learning* untuk mendapatkan hasil yang lebih optimal.
6. Penelitian selanjutnya dapat fokus pada pengembangan sistem *early warning* dengan menggunakan hasil klasifikasi untuk mengidentifikasi mahasiswa yang berisiko *dropout* sejak dini, sehingga dapat dilakukan intervensi yang tepat.

DAFTAR PUSTAKA

- [1]. F. C. Anggian, N. Hidayat, and M. T. Furqon, “Implementasi metode Modified K-Nearest Neighbor untuk klasifikasi status gunung berapi,” *J. Pengemb. Teknol. Inf. dan Ilmu Komput. (J-PTIIK)*, vol. 3, no. 12, pp. 11027–11033, 2019. [Online]. Available: <https://j-ptiik.ub.ac.id/index.php/j-ptiik/article/view/6843>
- [2]. M. O. Arowolo, M. Adebiyi, O. Okesola, and A. Adebiyi, “PCA model for RNA-Seq malaria vector data classification using KNN and decision tree algorithm,” in *Proc. 2020 Int. Conf. Math., Comput. Eng. and Comput. Sci. (ICMCECS)*, Lagos, Nigeria, 2020, pp. 1–8, doi: 10.1109/ICMCECS47690.2020.9240881.
- [3]. N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, “SMOTE: Synthetic minority over-sampling technique,” *J. Artif. Intell. Res.*, vol. 16, pp. 321–357, 2002, doi: 10.1613/jair.953. [Online]. Available: <https://www.jair.org/index.php/jair/article/view/10302>
- [4]. I. Cholissodin, F. M. Evanita, J. J. Tedjasulaksana, and K. W. Wahyuditomo, “Klasifikasi tingkat laju data COVID-19 untuk mitigasi penyebaran menggunakan metode Modified K-Nearest Neighbor (MKNN),” *J. Teknol. Inf. dan Ilmu Komput. (JTIIK)*, vol. 8, no. 3, pp. 595–600, 2021. [Online]. Available: <https://www.researchgate.net/publication/352412864>
- [5]. S. García, J. Luengo, and F. Herrera, *Data Preprocessing in Data Mining*, Intell. Syst. Ref. Library, vol. 72. New York, NY, USA: Springer, 2015, ISBN: 978-3-319-10246-7, doi:

- 10.1007/978-3-319-10247-4. [Online]. Available: [http://repo.darmajaya.ac.id/4085/1/Data%20preprocessing%20in%20data%20mining%20\(%20PDFDrive%20\).pdf](http://repo.darmajaya.ac.id/4085/1/Data%20preprocessing%20in%20data%20mining%20(%20PDFDrive%20).pdf)
- [6]. A. N. Kasanah, Muladi, and U. Pujianto, “Penerapan teknik SMOTE untuk mengatasi imbalance class dalam klasifikasi objektivitas berita online menggunakan algoritma KNN,” *J. RESTI*, vol. 3, no. 2, pp. 196–201, 2019. [Online]. Available: <http://jurnal.iain.or.id/index.php/RESTI/article/view/793>
- [7]. Ö. Karal, “Performance comparison of different kernel functions in SVM for different k value in k-fold cross-validation,” in *Proc. Int. Conf. Electr. and Electron. Eng.*, IEEE, 2020, pp. 1–6, doi: 10.1016/j.egyai.2022.100144. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2352484722013270>
- [8]. M. Nishom, “Comparison of Euclidean distance, Minkowski distance, and Manhattan distance in K-means clustering algorithm based on chi-square,” *J. Informatika: J. Pengemb. IT*, vol. 4, no. 1, pp. 20–24, 2019, doi: 10.30591/jpit.v4i1.1253. [Online]. Available: <https://ejournal.poltekharber.ac.id/index.php/informatika/article/view/1253>
- [9]. S. Nurjanah, A. Subiyakto, and Nurmiati, “Implementasi Principal Component Analysis (PCA) pada sistem klasifikasi data medis untuk meningkatkan akurasi diagnosis,” *J. Teknol. Inf. dan Ilmu Komput. (JTIIK)*, vol. 9, no. 7, pp. 1763–1770, 2022. [Online]. Available: <https://jtiik.ub.ac.id/index.php/jtiik/article/view/6789>
- [10]. F. I. Pratama and A. Budianita, “Optimization of K-NN classification in human gait recognition using Gabor wavelet and PCA,” in *Proc. Int. Conf. Telecommun. and Signal Process. (TSP)*, IEEE, 2021, pp. 830–840, doi: 10.1109/TSP52891.2021.9288653. [Online]. Available: <https://ieeexplore.ieee.org/document/9288653>
- [11]. N. K. I. Setiawati and I. G. A. Wibawa, “Penerapan algoritma K-Nearest Neighbor dalam klasifikasi penyakit gagal jantung,” *J. Nas. Teknol. Inf. dan Aplikasinya (JNATIA)*, vol. 1, no. 1, pp. 347–352, 2022. [Online]. Available: <https://jurnal.harianregional.com/jnatia/id-92598>
- [12]. Thomas and R. Gupta, “A survey on machine learning approaches and its techniques,” in *Proc. IEEE Int. Students’ Conf. Electr., Electron. and Comput. Sci. (SCEECS)*, IEEE, 2020, pp. 1–10, doi: 10.1109/SCEECS48394.2020.190. [Online]. Available: <https://ieeexplore.ieee.org/document/9087123>
- [13]. M. Utari, B. Warsito, and R. Kusumaningrum, “Implementation of data mining for drop-out prediction using random forest method,” in *Proc. IEEE Int. Students’ Conf. Electr., Electron. and Comput. Sci. (SCEECS)*, IEEE, 2021, pp. 142–149, doi: 10.1109/SCEECS52399.2021.9166276. [Online]. Available: <https://ieeexplore.ieee.org/document/9166276>
- [14]. T. Yuwono, A. Franz, and I. Muhammah, “Design of smart electrocardiography (ECG) using Modified K-Nearest Neighbor (MKNN),” in *Proc. 2018 Int. Seminar on Appl. for Technol. of Inf. and Commun. (iSemantic)*, Semarang, Indonesia, 2018, doi: 10.1109/CAIS.2018.8441983. [Online]. Available: <https://www.researchgate.net/publication/327196503>
- [15]. S. Zhang, “Challenges in KNN classification,” *IEEE Trans. Knowl. Data Eng.*, vol. 34, no. 11, pp. 5517–5530, Nov. 2022, doi: 10.1109/TKDE.2021.3049250. [Online]. Available: <https://ieeexplore.ieee.org/document/9314060>
- [16]. H. Halim, S. M. Isa, and S. Mulyono, “Comparative analysis of PCA and KPCA on paddy growth stages classification,” in *Proc. 2016 IEEE Region 10 Symp. (TENSYMP)*, Bali, Indonesia, 2016, pp. 167–172, doi: 10.1109/TENCONSpring.2016.7519431. [Online]. Available: <https://ieeexplore.ieee.org/document/7519398>