# OPTIMASI K-NEAREST NEIGHBORS MENGGUNAKAN FUZZY C-MEANS PADA KETEPATAN WAKTU KELULUSAN MAHASISWA

**SKRIPSI**

**Oleh:**
**LAILATUL FADILAH**
**NIM. 19650032**

**PROGRAM STUDI TEKNIK INFORMATIKA**
**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM**
**MALANG**
**2023**

**OPTIMASI K-NEAREST NEIGHBORS MENGGUNAKAN FUZZY C-MEANS PADA KETEPATAN WAKTU KELULUSAN MAHASISWA**

**SKRIPSI**

**Oleh:**
**LAILATUL FADILAH**
**NIM. 19650032**

**Diajukan kepada:**
**Fakultas Sains dan Teknologi**
**Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang**
**Untuk Memenuhi Salah Satu Persyaratan Dalam**
**Memperoleh Gelar Sarjana Komputer (S.Kom)**

**PROGRAM STUDI TEKNIK INFORMATIKA**
**FAKULTAS SAINS DAN TEKNOLOGI**
**UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM MALANG**
**2023**

## HALAMAN PERSETUJUAN

## OPTIMASI K-NEAREST NEIGHBORS MENGGUNAKAN FUZZY C-MEANS PADA KETEPATAN WAKTU KELULUSAN MAHASISWA

**SKRIPSI**

**Oleh:**
**LAILATUL FADILAH**
**NIM. 19650032**

Telah Diperiksa dan Disetujui untuk Diuji
Tanggal: 13 Juni 2023

| Pembimbing I | Pembimbing II |
|---|---|
| Fajar Rohman Hariri, M.Kom | Dr. M. Ainul Yaqin, M.Kom |
| NIP. 19890515 201801 1 001 | NIP. 19761013 200604 1 004 |

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Fachrul Kurniawan, M.MT .,IPM
NIP. 19771020 200912 1 001

## HALAMAN PENGESAHAN

## OPTIMASI K-NEAREST NEIGHBORS MENGGUNAKAN FUZZY C-MEANS PADA KETEPATAN WAKTU KELULUSAN MAHASISWA

## SKRIPSI

**Oleh:**
**LAILATUL FADILAH**
**NIM. 19650032**

Telah Dipertahankan di Depan Dewan Penguji Skripsi
Dan Dinyatakan Diterima Sebagai Salah Satu Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal: 21 Juni 2023

**Susunan Dewan Penguji**

| | | |
|---|---|---|
| Ketua Penguji | : Dr. Totok Chamidy, M.Kom<br>NIP. 19691222 200604 1 001 | ( ) |
| Anggota Penguji I | : Roro Inda Melani. M.T. M.Sc<br>NIP. 19780925 200501 2 008 | ( ) |
| Anggota Penguji II | : Fajar Rohman Hariri, M.Kom<br>NIP. 19890515 201801 1 001 | ( ) |
| Anggota Penguji III | : Dr. M. Ainul Yaqin, M.Kom<br>NIP. 19761013 200604 1 004 | ( ) |

Mengetahui,
Ketua Program Studi Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang

Dr. Fachrul Kurniawan, M.MT ,IPM
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini:

| | |
|---|---|
| Nama | : Lailatul Fadilah |
| NIM | : 19650032 |
| Fakultas/Program Studi | : Sains dan Teknologi/Teknik Informatika |
| Judul Skripsi | : Optimasi K-Nearest Neighbors menggunakan Fuzzy C-Means pada Ketepatan Waktu Kelulusan Mahasiswa |

Menyatakan dengan sebenarnya bahwa skripsi yang saya tulis ini benar benar merupakan hasil karya saya sendiri, bukan merupakan pengambilan daya, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka. Apabila di kemudian hari terbukti atau dapat dibuktikan skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan saya tersebut.

Malang, 22 Juni 2023
Yang membuat pernyataan,


Lailatul Fadilah
NIM. 19650032

## HALAMAN MOTTO

“Stick on your why and finish what you started”

## HALAMAN PERSEMBAHAN

**Puji syukur kehadirat Allah SWT, shalawat serta salam kepada Rasulullah SAW.**

Skripsi ini saya persembahkan untuk Kedua Orang Tua saya,
Bapak Achmad Musta'in dan Ibu Fitrianingsih, Keluarga,
Seluruh Dosen, Sahabat, Teman-Teman Seperjuangan,
Seluruh Pihak yang Terlibat, dan Diri Saya Sendiri

Terima kasih

# KATA PENGANTAR

*Bismillahirrahmanirrahiin*, *Alhamdulillahirabbil'aalamiin*, segala puji syukur penulis limpahkan atas kehadirat Allah SWT yang telah melimpahkan rahmat dan karunia-Nya, sehingga peneliti mampu menyelesaikan skripsi dengan baik. Salawat serta serta salam senantiasa tercurahkan kepada Baginda Nabi Muhammad SAW yang telah memberikan panutan bagi seluruh umat manusia. Semoga kita termasuk dalam golongan yang mendapatkan pertolongan Nabi Muhammad SAW di hari akhir.

Penulis menyadari bahwa penyusunan skripsi ini tak luput dari dukungan dan bantuan dari berbagai pihak. Oleh karena itu, dengan segala kerendahan hati, penulis menyampaikan terimakasih kepada:

1. Prof. Dr. M. Zainuddiin, MA selaku rektor Universitas Islam Negeri Maulana Malik Ibrahin Malang.
2. Dr. Sri Hariani, M.Si selaku dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang.
3. Dr. Fachrul Kurniawan, M.MT selaku ketua Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang.
4. Fajar Rohman Hariri, M.Kom selaku dosen pembimbing I yang telah bersedia meluangkan waktunya untuk membimbing dan memberikan arahan dalam penyusunan skripsi ini.
5. Dr. M. Ainul Yaqin, M.Kom selaku Dosen Pembimbing II yang sudah membantu memberikan arahan, bimbingan, dan dorongan selama penulisan skripsi ini.

6. Dr. Totok Chamidy, M.Kom sebagai ketua penguji dan Roro Inda Melani, M.T, M.Sc sebagai anggota penguji I yang telah meluangkan waktu dan memberikan arahan yang membangun untuk skripsi ini.
7. Seluruh dosen dan staf Program Studi Teknik Informatika yang telah memberikan ilmu dan pengalaman yang berharga.
8. Seluruh staf Bagian Administrasi Akademik Universitas Islam Negeri Maulana Malik Ibrahim Malang terutama saudara M.Abror yang telah membantu memberikan data sesuai prosedur yang berlaku.
9. Kedua orang tua, bapak Achmad Musta'in dan ibu Fitrianingsih, yang telah memberikan banyak dukungan moril dan materiil serta doa selama proses studi dan penyusunan skripsi ini.
10. Seluruh keluarga besar Program Studi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang, terutama saudari Adisa Dwi Wanti dan Salma Zulfatul Latifah yang telah memberikan semangat, bantuan, dan doa sepanjang masa studi dan selama penyusunan skripsi ini.
11. Serta seluruh pihak yang pernah membantu dalam proses penulisan skripsi ini, baik secara langsung maupun tidak langsung yang tidak bisa disebutkan satu per satu.

Penulis menyadari bahwa dalam penyusunan skripsi ini masih jauh dari kesempurnaan, baik dari segi keilmuan dan kepenulisan. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun. Semoga penulisan skripsi ini dapat bermanfaat bagi banyak pihak.

Malang, 12 Juni 2023

Penulis

**DAFTAR ISI**

## DAFTAR GAMBAR

## DAFTAR TABEL

# ABSTRAK

Fadilah, Lailatul, 2023. **Optimasi K-Nearest Neighbors menggunakan Fuzzy C-Means pada Ketepatan Waktu Kelulusan Mahasiswa**. Skripsi. Program Studi Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Fajar Rohman Hariri, M.Kom (II) Dr. M. Ainul Yaqin, M.Kom.

*Kata Kunci : prediksi, ketepatan waktu kelulusan, fuzzy c-means, k-nearest neighbors, data mining*

Ketepatan waktu kelulusan mahasiswa perlu diprediksi karena berpengaruh pada keberhasilan evaluasi akademik. Evaluasi akademik memiliki tantangan dalam pelaksanaanya seperti adanya metode yang beragam, pendekatan evaluasi yang berbeda-beda, dan kesulitan mengukur kriteria abstrak. Prediksi ketepatan waktu kelulusan mahasiswa sebagai solusi yang ditawarkan pada penelitian ini untuk mengoptimalkan evaluasi akademik. Objek penelitian ini yaitu mahasiswa Program Studi Teknik Informatika UIN Malang. Dalam kurun waktu 2014 sampai 2018, rerata tingkat kelulusan tepat waktu mahasiswa hanya 29%. Angka tersebut tergolong rendah jika dibandingkan dengan standar persentase penilaian akreditasi oleh BAN-PT. Tujuan penelitian ini yaitu membuat model prediksi yang dapat melakukan klastering data dan prediksi ketepatan waktu kelulusan mahasiswa. Penelitian ini menggunakan 2 metode; *fuzzy c-means* (algoritma klastering data) dan *k-nearest neighbors* (algoritma prediksi data). Implementasi algoritma memperhitungkan 6 atribut yaitu indeks prestasi semester 1-4, jenis kelamin, jenis pembiayaan. Dataset didapat dari data akademik mahasiswa tahun 2014 sampai 2018. Dataset diolah dengan 4 tahap *preprocessing;* data cleaning, data integration, data transformation, data reduction. Uji coba dilakukan dengan komposisi data *training* dan *testing* 60:40 dan 70:30. Dalam pengimplementasian algoritma *fuzzy c-mean* dilakukan skenario kustomisasi nilai variabel $c$=2, $c$=3, $c$=4. Dalam pengimplementasian algoritma *k-nearest neighbors* dilakukan skenario kustomisasi nilai variabel $k$=1, $k$=3, $k$=5, $k$=7. Hasil uji coba menunjukkan bahwa dari 16 model prediksi yang dihasilkan, model prediksi 11 memiliki akurasi terbaik yaitu 74.7% dengan komposisi data *training* dan data *testing* 60:40, kustomisasi nilai $c=3$, kustomisasi nilai $k=5$. Model terbaik yang ditemukan dapat digunakan untuk memprediksi persentase tingkat kelulusan tepat waktu mahasiswa di Program Studi Teknik Informatika. Persentase tersebut dapat digunakan sebagai acuan program studi untuk memperbarui, menambah, atau memperbaiki kebijakan akademik dalam rangka mengoptimalkan evaluasi akademik.

# ABSTRACT

Fadilah, Lailatul, 2023. **Optimization of K-Nearest Neighbors using Fuzzy C-Means on Student Graduation Timeliness**. Undergraduate Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University Maulana Malik Ibrahim Malang. Supervisor: (I) Fajar Rohman Hariri, M.Kom (II) Dr. M. Ainul Yaqin, M.Kom.

The timeliness of student graduation needs to be predicted because it affects the success of academic evaluations. Academic evaluation has challenges in its implementation such as the existence of various methods, different evaluation approaches, and difficulty measuring abstract criteria. Predicting student graduation timeliness as a solution offered in this study to optimize academic evaluation. The objects of this research are students of the UIN Malang Informatics Engineering Study Program. From 2014 to 2018, the average on-time graduation rate for students was only 29%. This figure is low when compared to the standard percentage for accreditation assessment by BAN-PT. The purpose of this study is to create a predictive model that can cluster data and predict the timeliness of student graduation. This research uses 2 methods; fuzzy c-means (data clustering algorithm) and k-nearest neighbors (data prediction algorithm). Algorithm implementation takes into account 6 attributes, namely grade point average 1-4, gender, type of financing. The dataset was obtained from student academic data from 2014 to 2018. The dataset was processed in 4 stages of preprocessing; data cleaning, data integration, data transformation, data reduction. The trials were carried out with the composition of training and testing data 60:40 and 70:30. In implementing the fuzzy c-mean algorithm, scenarios for customizing the values of variables c=2, c=3, c=4 are carried out. In implementing the k-nearest neighbors algorithm, scenarios for customizing the variable values k=1, k=3, k=5, k=7 are carried out. The results show that of the 16 prediction models produced, prediction model 11 has the best accuracy, namely 74.7% with a composition of training data and testing data of 60:40, customizing the value of c = 3, customizing the value of k = 5. The best model found can be used to predict the percentage of on-time graduation rate of students in the Informatics Engineering Study Program. This percentage can be used as a reference for study programs to update, add, or improve academic policies in order to optimize academic evaluation.

*Keywords : prediction, graduation timeliness prediction, fuzzy c-means, k-nearest neighbors, data mining*

الفضيلة، ليلة. 2023. تحسين K-Nearest Neighbours باستخدام Fuzzy C-Means في توقيت تخرج الطلاب. رسالة بكالوريوس. قسم تكنولوجيا المعلومات، كلية العلوم والتكنولوجيا، جامعة مولانا مالك إبراهيم مالانغ الإسلامية الدولية، مشرف: (I) فجر رحمان هاريري، م.كوم. (II) م. عين يقين م.كوم.

كلمات مفتاحية: التنبؤ ، توقيت التخرج ، Fuzzy C-Means ، K-Nearest Neighbours ، التنقيب في البيانات.

يجب توقع توقيت تخرج الطلاب لأنه يؤثر على نجاح التقييمات الأكاديمية. يواجه التقييم الأكاديمي تحديات في تنفيذه مثل وجود طرق مختلفة ، ومقاربات تقييم مختلفة ، وصعوبة قياس المعايير المجردة. التنبؤ بمواعيد تخرج الطلاب كحل مقدم في هذه الدراسة لتحسين التقييم الأكاديمي. أهداف هذا البحث هم طلاب برنامج دراسة هندسة المعلوماتية UIN Malang. من عام 2014 إلى عام 2018 ،كان متوسط معدل التخرج في الوقت المحدد للطلاب 29 ٪ فقط. هذا الرقم منخفض بالمقارنة مع النسبة المئوية القياسية لتقييم الاعتماد من قبل BAN-PT. الغرض من هذه الدراسة هو إنشاء نموذج تنبؤي يمكنه تجميع البيانات والتنبؤ بمواعيد تخرج الطلاب. يستخدم هذا البحث طريقتين ؛ Fuzzy C-Means (خوارزمية تجميع البيانات) و K-Nearest Neighbours (خوارزمية توقع البيانات). يأخذ تطبيق الخوارزمية في الاعتبار 6 سمات وهي المعدل التراكمي في الفصول من 1 إلى 4 ، الجنس ، نوع التمويل. تم الحصول على مجموعة البيانات من البيانات الأكاديمية للطلاب من 2014 إلى 2018. تتم معالجة مجموعة البيانات بأربع مراحل من المعالجة المسبقة ؛ تنظيف البيانات ، تكامل البيانات ، تحويل البيانات ، تقليل البيانات. تم إجراء التجارب باستخدام بيانات التدريب والاختبار 60:40 و 70:30. عند تنفيذ خوارزمية Fuzzy C-Mean ، يتم تنفيذ سيناريوهات لتخصيص قيم المتغيرات c = 2 ، c = 3 ، c = 4. عند تنفيذ خوارزمية K-Nearest Neighbours ، يتم تنفيذ سيناريوهات لتخصيص قيم المتغيرات k = 1 ، k = 3 ، k = 5 ، k = 7. أظهرت نتائج الاختبار أنه من بين 16 نموذج تنبؤ تم إنتاجه ،كان لنموذج التنبؤ 11 أفضل دقة بنسبة 74.68٪ مع تركيبة بيانات التدريب وبيانات الاختبار 60:40 ، تخصيص قيمة C = 3 ، تخصيص قيمة k = 5. يمكن استخدام أفضل نموذج تم العثور عليه للتنبؤ بنسبة التخرج في الوقت المحدد للطلاب في برنامج دراسة هندسة المعلوماتية. يمكن استخدام هذه النسبة المئوية كمرجع لبرامج الدراسة لتحديث أو إضافة أو تحسين السياسات الأكاديمية من أجل تحسين التقييم الأكاديمي

# BAB I

# PENDAHULUAN

## 1.1 Latar Belakang

Salah satu hal yang diperhatikan institusi perguruan tinggi sebagai pertahanan dan peningkatan mutu akademik adalah ketepatan waktu lulus mahasiswa. Persentase ketepatan waktu lulus mahasiswa masuk dalam indikator akuntabilitas akademik dan non akademik perguruan tinggi (BAN-PT, 2022). Semakin besar persentase mahasiswa yang lulus tepat waktu akan semakin membantu penilaian akreditasi program studi. Selain itu, ketepatan waktu lulus juga dapat membebaskan mahasiswa dari tanggungan membayar biaya kuliah serta memperbesar peluang mereka untuk lebih cepat mendapatkan pekerjaan. Kriteria ketepatan waktu lulus mahasiswa diatur dalam panduan akademik program studi. Umumnya, mahasiswa dikatakan lulus tepat waktu jika berhasil menyelesaikan kuliah dalam 8 semester dan paling lama 9 semester (Kamil & Cholil, 2020).

Prediksi adalah suatu perhitungan yang melibatkan beberapa variabel di kumpulan data untuk meramalkan nilai yang belum diketahui atau masa depan dari variabel lain (Vijiyarani & Sudha, 2013). Prediksi ketepatan waktu lulus mahasiswa berarti menentukan perkiraan besarnya peluang seorang mahasiswa dapat lulus tepat waktu berdasarkan perhitungan variabel-variabel tertentu. Informasi yang dihasilkan dari proses prediksi, dapat dimanfaatkan untuk hal lain salah satunya pengambilan keputusan.

Ketepatan waktu kelulusan mahasiswa perlu diprediksi karena berpengaruh pada keberhasilan mahasiswa dan lingkungan program studi salah

satunya keberhasilan dalam hal evaluasi akademik. Evaluasi akademik berkaitan dengan penilaian terhadap prestasi mahasiswa maupun prestasi program studi. Perguruan tinggi dan program studi harus menonjolkan subjektivitas mahasiswa dalam evaluasi akademik serta melibatkan kriteria-kriteria tertentu sehingga evaluasi akademik dapat menghasilkan model baru (Yi, 2010). Akan tetapi, evaluasi akademik memiliki tantangan-tantangan dalam pelaksanaannya seperti adanya penilaian bias dari seorang dosen, metode evaluasi yang beragam, adanya kesulitan mengukur kriteria abstrak, serta pendekatan evaluasi yang berbeda-beda. Evaluasi akademik yang kurang optimal juga berdampak pada reputasi. Reputasi berkaitan dengan citra profesionalisme mahasiswa, citra kemampuan mahasiswa dalam bidang akademik, peluang karir, serta reputasi program studi. Prediksi ketepatan waktu kelulusan mahasiswa dapat memberikan gambaran keseluruhan dari kinerja mahasiswa serta kemampuan mereka dalam mengelola waktu dan tugas perkuliahan. Oleh karena itu, prediksi ketepatan waktu mahasiswa dapat membantu mahasiswa dan program studi untuk mengoptimalkan evaluasi akademik.

Adapun dalam urgensi keagamaan, prediksi ketepatan waktu kelulusan mahasiswa perlu dilakukan karena merupakan salah satu bentuk usaha menghargai waktu. Hal tersebut disinggung dalam salah satu hadis Rasulullah SAW berikut.

نِعْمَتَانِ مَغْبُوْنٌ فِيْهِمَا كَثِيْرٌ مِنَ النَّاسِ: الصِّحَّةُ وَالفَرَاغُ

*"Dua nikmat yang banyak manusia tertipu di dalam keduanya, yaitu nikmat sehat dan waktu luang."* (HR Bukhari, Tirmidzi dan Ibnu Majah).

Abdul Fattah bin Muhammad dalam Qimatuz Zaman ‘Indal ‘Ulama memberikan penjelasan bahwa yang dimaksud dengan terminologi ‘tertipu’ dalam

hadis tersebut yaitu ‘merugi’. Banyak manusia, khususnya mahasiswa, akan merugi dalam agama jika kurang memanfaatkan dan mensyukuri nikmat sehat dan waktu luang. Maka dari itu menyelesaikan kuliah dengan lulus tepat waktu juga merupakan tanggung jawab dari segi agama. Sehingga perlu diutamakan dibandingkan kegiatan lain yang tidak bermanfaat.

Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang mengalami kenaikan jumlah mahasiswa tiap tahun dalam kurun waktu 2014 sampai 2018 (Pddikti.kemdikbud.go.id., 2021). Berdasarkan data Pangkalan Data Perguruan Tinggi Kementerian Pendidikan, pada tahun 2014 terdapat 115 mahasiswa dengan jumlah kelulusan tepat waktu 16% . Pada tahun 2015 jumlah mahasiswa mengalami kenaikan yaitu 133 mahasiswa, dengan jumlah kelulusan tepat waktu 20%. Pada tahun 2016 jumlah mahasiswa meningkat sebanyak 131 dengan jumlah kelulusan tepat waktu 40%. Pada tahun 2017, jumlah mahasiswa angkatan tersebut 125 orang dengan tingkat kelulusan tepat waktu 26%. Pada tahun 2018, jumlah mahasiswa 127 dengan tingkat kelulusan tepat waktu 43%. Sehingga dalam kurun waktu 5 tahun, rata-rata jumlah mahasiswa yang mendapat predikat Lulus Tepat Waktu yaitu 29%.

Dari fenomena tersebut, dapat dilihat bahwa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang dalam 5 tahun mengalami pertambahan jumlah mahasiswa yang tidak diimbangi dengan jumlah kelulusan tepat waktu. Hal ini berpengaruh pada evaluasi akademik, utamanya berkaitan dengan reputasi program studi. Hal tersebut ditunjukkan oleh fakta bahwa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang ini belum

pernah memperoleh nilai akreditasi A atau Unggul. Pertambahan jumlah mahasiswa juga berdampak pada penumpukan data akademik mahasiswa. Data berjumlah besar dapat memberikan informasi baru yang berguna sebagai pendukung evaluasi akademik jika diolah menggunakan metode tertentu.

Oleh karena itu diperlukan suatu model yang dapat memprediksi ketepatan waktu kelulusan mahasiswa untuk membantu meningkatkan evaluasi akademik dalam program studi. Model prediksi tersebut dapat memberikan informasi sejak dini tentang tingkat ketepatan waktu kelulusan mahasiswa. Informasi tersebut bisa digunakan pihak program studi untuk mengevaluasi kebijakan guna meningkatkan kualitas akademik mahasiswa dan program studi.

Tujuan penelitian ini yaitu membuat model prediksi yang dapat melakukan klasterisasi data dan memprediksi ketepatan waktu kelulusan mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang menggunakan metode *Fuzzy C-Means* dan *k-Nearest Neighbors*. Adapun objek penelitian berupa data akademik mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang tahun 2014-2018 yang berjumlah 631 data. Atribut data yang diperhitungkan yaitu indeks prestasi semester, jenis kelamin, jenis pembiayaan, serta data target berupa label Lulus Tepat Waktu dan Lulus Tidak Tepat Waktu.

Sistem prediksi biasanya dilakukan pada kumpulan data berskala besar, sehingga diperlukan metode *data mining* (Dany, 2014). *Data mining* adalah proses ekstraksi dari informasi implisit yang sebelumnya belum memiliki kegunaan (Larose, 2014). *Data mining* memiliki beberapa fungsi dalam menggali dan

menambang knowledge dari data antara lain fungsi estimasi (*estimation*), fungsi klasifikasi (*classification*), fungsi prediksi (*prediction*), fungsi asosiasi (*association*), fungsi pengelompokkan (*classification*) dan fungsi deskripsi (*description*). Ada berbagai metode *data mining* beberapa di antaranya metode *k-Nearest Neighbors*, *Decision Tree*, *K-Means*, dan *Naive Bayes*. Metode-metode tersebut masuk ke dalam kategori fungsi klasifikasi, tetapi knowledge yang ditemukan dari hasil klasifikasi tersebut dapat digunakan untuk prediksi (Maghari, 2018).

Penelitian sejenis sudah pernah dilakukan pada penelitian terdahulu, antara lain penelitian oleh Zainuddin pada tahun 2019 yaitu membandingkan algoritma *Naive Bayes*, *Decision Tree* (C4.5), *k-Nearest Neighbor*, dan *Neural Network* berbasis *particle swarm optimization* untuk prediksi ketepatan waktu kelulusan mahasiswa. Penelitian tersebut menghasilkan algoritma dengan performa terbaik adalah *k-Nearest Neighbors* dengan penambahan fitur PSO yaitu memiliki akurasi sebesar 74.08% (Zainuddin, 2019). Dalam penelitian lain oleh (Maghari, 2018), menggunakan enam modifikasi dari metode *k-Nearest Neighbors* yaitu Fine KNN, Medium KNN, Coarse KNN, Cosine KNN, Cubic KNN, dan Weighted KNN untuk memprediksi kinerja siswa menggunakan data nilai dua mata pelajaran. Penelitian tersebut menyimpulkan semua metode memiliki akurasi lebih dari 80% yang artinya metode *k-Nearest Neighbors* dapat memberikan akurasi yang tinggi jika dimodifikasi (Maghari, 2018). Penelitian lain menggabungkan dua metode *Fuzzy C-Means* dan *k-Nearest Neighbors* menggunakan data registrasi mahasiswa oleh (Nabila et al., 2021). Pada penelitian ini, algoritma FCM digunakan untuk memberi

label *cluster* data kemudian proses dilanjutkan dengan penentuan kelas menggunakan KNN. Penelitian ini menghasilkan akurasi 71% dengan skenario k=1 dan pengujian 10-*fold cross validation* (Nabila et al., 2021). Penelitian-penelitian sebelumnya berpendapat bahwa metode *data mining* yang terbaik salah satunya ialah metode *k-Nearest Neighbors* yang dimodifikasi. Metode tersebut memiliki keunggulan yaitu tangguh terhadap data latih yang *noise* dan sangat efektif jika data latihnya berskala besar (Amalia, 2018).

Penelitian ini menggunakan gabungan algoritma *Fuzzy C-Means* dan metode *k-Nearest Neighbors* untuk membuat model prediksi. *Fuzzy C-Means* berfungsi untuk melakukan klasterisasi data sedangkan *k-Nearest Neighbors* berfungsi untuk memprediksi kelas data. Klasterisasi data dilakukan terlebih dahulu untuk mengurangi data *noise* dan mengandung *outliers* karena data yang digunakan dalam penelitian ini merupakan jenis data yang rentan mengalami *human entry error*. Klasterisasi data menghasilkan dataset yang terbagi menjadi beberapa *cluster*, kemudian prediksi data dilakukan dengan menguji data *testing* dengan data *training* dalam *cluster* terdekat saja. Pengujian data *testing* tidak dilakukan terhadap seluruh data karena telah dilakukan klasterisasi sebelumnya.

Penelitian dilakukan dalam 16 skenario uji coba dimana masing-masing model prediksi akan diberikan perlakuan yang berbeda-beda. Adapun perlakuan yang dimaksud antara lain kustomisasi nilai *c* dalam penerapan algoritma *Fuzzy C-Means*, kustomisasi nilai *k* dalam penerapan metode *k-Nearest Neighbors*, serta pembagian dataset menjadi data training (60%) dan data testing (40%). Keenambelas model prediksi akan dibandingkan berdasarkan nilai *accuracy*,

*precision*, dan *recall*. Model prediksi dengan nilai performa paling baik kemudian akan diuji ketepatannya menggunakan metode *confusion matrix*. Penelitian ini menghasilkan sebuah model prediksi dengan performa paling optimal yang memiliki spesifikasi nilai *c* dan nilai *k* tertentu.

Manfaat yang diharapkan yaitu penelitian ini dapat memberikan informasi yang mendukung evaluasi kebijakan sistem akademik yang berkaitan tentang peningkatan kualitas program studi.

## 1.2 Pernyataan Masalah

Berdasarkan latar belakang yang dijelaskan, dirumuskan pernyataan masalah bagaimana meningkatkan evaluasi akademik di Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang ?

## 1.3 Tujuan Penelitian

Tujuan penelitian ini yaitu membuat model prediksi yang dapat melakukan klasterisasi data dan memprediksi ketepatan waktu kelulusan mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang menggunakan metode *Fuzzy C-Means* dan *k-Nearest Neighbors*.

## 1.4 Batasan Masalah

Untuk menghindari penyimpangan tafsir akan permasalahan dalam penelitian ini, maka berikut batasan-batasan masalah yang ditentukan:

1. Ruang lingkup objek penelitian adalah data akademik mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang tahun 2014-2018.

2. Kriteria kelulusan “Lulus Tepat Waktu” didasarkan pada standar kelulusan Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang, yaitu 4 tahun. Lebih dari itu akan dianggap “Lulus Tidak Tepat Waktu”.

## 1.5 Manfaat Penelitian

Adapun manfaat yang diharapkan dari penelitian ini adalah sebagai berikut:

1. Bagi bidang akademik, hasil penelitian ini diharapkan dapat bermanfaat untuk memberikan informasi yang mendukung evaluasi kebijakan sistem akademik untuk peningkatan akreditasi program studi, terutama yang berkaitan dengan ketepatan waktu kelulusan mahasiswa.
2. Bagi bidang keilmuan, penelitian ini diharapkan dapat memperkaya referensi untuk pengembangan ilmu pada penelitian-penelitian berikutnya.
3. Bagi peneliti, penelitian ini merupakan pengembangan dan pengimplementasian ilmu yang telah didapat dalam masa studi.

# BAB II

# STUDI PUSTAKA

Pada bagian studi pustaka akan membahas penelitian-penelitian terdahulu yang berkaitan dengan topik serupa sebagai referensi dan perbandingan, serta landasan teori yang mendukung penelitian saat ini.

## 2.1 Penelitian Terdahulu

Penelitian dengan isu terkait pernah dilakukan oleh (Rohmawan, 2018), membahas tentang prediksi kelulusan mahasiswa tepat waktu menggunakan metode *Decision Tree* dan *Artificial Neural Network*. Penelitian tersebut dilakukan terhadap data *training* berupa data induk dan data akademik mahasiswa di universitas A angkatan 2006-2010. Sedangkan data *testing* diambil data mahasiswa angkatan 2011-2012. Penelitian tersebut menggunakan 9 atribut. Tujuan penelitian untuk membandingkan dua metode yang berbeda berdasarkan hasil akurasinya. Implementasi metode *Decision Tree* menghasilkan akurasi sebesar 74.5% dimana prediksi mahasiswa yang lulus tepat waktu sebanyak 56.16% dan prediksi yang telat sebanyak 91.25%. Implementasi metode *Artificial Neural Network* menghasilkan akurasi 79.74% dengan prediksi mahasiswa yang lulus tepat waktu sebanyak 63.49% dan yang telat 91.11%. Sehingga disimpulkan dari penelitian tersebut bahwa dari kedua metode, *Artificial Neural* memiliki tingkat akurasi yang lebih tinggi karena penelitian menggunakan data berlabel.

Penelitian lain dilakukan oleh (Maghari, 2018) tentang prediksi performa siswa menggunakan metode *k-Nearest Neighbors* yang dimodifikasi. Penelitian

bertujuan untuk memprediksi performa siswa sekolah menengah di Jalur Gaza dalam dua mata pelajaran dan menemukan performa metode terbaik. Dataset yang digunakan berisi 13 atribut; 2 atribut identitas siswa dan 11 atribut berisi nilai mata pelajaran, rerata, dan status nilai. Penelitian ini membagi dataset secara random menjadi dua bagian; 70% digunakan untuk data latih dan 30% untuk data uji. Model prediksi didasarkan pada status nilai; *Good, Very Good, Excellent, Fair, Pass*. Metode yang dibandingkan antara lain *Fine KNN CLassifier*, *Medium KNN Classifier*, *Coarse KNN Classifier*, *Cosine KNN Classifier*, *Cubic KNN Classifier*, *Weighted KNN Classifier*. Dari lima metode KNN-termodifikasi, performa terbaik ditunjukkan oleh metode *Weighted k-Nearest Neighbors* dengan akurasi 94% dan waktu 0.97885 sekon.

Penelitian tentang prediksi kelulusan tepat waktu mahasiswa dilakukan oleh (Zainuddin, 2019) dengan pengimplementasian 4 algoritma berbasis *Particle Swarm Optimization*. Adapun metode yang dibandingkan adalah *Naive Bayes*, *Decision Tree (C4.5)*, *K-Nearest Neighbors*, *Neural Network.* Objek yang digunakan adalah data Alumni Mahasiswa Jurusan Teknik Informatika dan Sistem Komputer Angkatan 2007–2011 STMIK ASIA Malang sebanyak 845 data. Atribut yang digunakan ada 12 dimana atribut 1-11 berisi data latih (jenis kelamin, umur, IP semester 1-7, status pekerjaan, status pernikahan), sedangkan atribut ke-12 sebagai atribut target dengan label ‘Lulus Tepat Waktu’ dan ‘Lulus Tidak Tepat Waktu’. *Preprocessing* dilakukan dengan dua tahap; *data validation* dan *data discretization*. Fase pemodelan dilakukan dua kali yaitu penerapan 4 metode secara normal dan penerapan 4 metode dengan penambahan fitur PSO. Penelitian tersebut

menghasilkan kesimpulan metode yang memiliki performa terbaik adalah *k-Nearest Neighbors* dengan penambahan fitur PSO yaitu menghasilkan akurasi 74.08% pada k-optimum=19.

Penelitian terdahulu yang mengimplementasikan algoritma *Fuzzy C-Means* berbasis PSO dilakukan oleh (Jamhur, 2020) untuk memprediksi predikat kelulusan mahasiswa. Objek penelitiannya berupa data mahasiswa STIKOM Binaniaga yang berjumlah 100 sampel. Sedangkan atribut yang diperhitungkan ada 2 antara lain NIM, IP Semester 1-4. Label data didasarkan pada predikat yudisium kelulusan mahasiswa yaitu "Memuaskan", "Sangat Memuaskan", "Dengan Pujian". Penelitian dilakukan dengan *preprocessing* data, penetapan parameter yang dibutuhkan (jumlah *cluster*, maksimum iterasi, dan kriteria penghentian proses), inisialisasi kelengkapan *swarm* (Matriks x, Matriks v, Matriks pbest, Matriks gbest), dan pengujian. Penelitian menghasilkan informasi bahwa dari 30 sampel data yang diuji, 20 mahasiswa mendapat predikat kelulusan "Sangat Memuaskan", 9 mahasiswa mendapat "Memuaskan", dan 1 mahasiswa mendapat "Dengan Pujian". Akurasi yang dihasilkan dari pengimplementasian metode FCM-PSO adalah 86% dengan waktu olah 2.5 sekon.

Penelitian serupa menggunakan metode *Fuzzy C-Means* dan *k-Nearest Neighbors* dilakukan oleh (Nabila et al., 2021). Penelitian tersebut bertujuan untuk membuat sistem yang dapat memprediksi ketepatan waktu lulus mahasiswa guna menyeleksi mahasiswa baru. Objek penelitiannya adalah data penerimaan mahasiswa UINSA tahun 2014-2015 dengan atribut yang diperhitungkan antara lain jalur penerimaan, asal pendidikan SMA/Sederajat, nem SMA/Sederajat, kode

prodi dan atribut target yaitu label kelulusan; “Lulus” dan “Tidak Lulus Tepat Waktu”. Tahap *preprocessing* dilakukan sebanyak tiga tahap kemudian dilakukan pelatihan model. Penelitian ini melakukan pengujian menggunakan teknik validasi *10-fold cross validation*. Performa metode FCM-KNN pada penelitian ini diperoleh akurasi 71% dengan hasil data “Lulus” 72 dan data “Tidak Lulus Tepat Waktu” sebanyak 1139.

Tabel 2.1 di bawah ini menunjukkan perbedaan masing-masing penelitian dilihat dari metode yang digunakan, tahap *data preprocessing*, serta atribut data yang diperhitungkan dalam prediksi. Tahap *data preprocessing* diberi kode 1-4 dimana kode 1 untuk proses data *cleaning*, kode 2 untuk proses data *integration*, kode 3 untuk proses data *transformation*, dan kode 4 untuk proses data *reduction*.

Tabel 2.1 Perbandingan Penelitian Terdahulu

| **Penelitian** | **Metode** | ***Preprocessing*** | | | | **Atribut Data** | **Hasil** |
|---|---|---|---|---|---|---|---|
| | | **1** | **2** | **3** | **4** | | |
| (Rohmawan, 2018) | 1) *Decision Tree*<br>2) *Artificial Neural Network* | - | ✓ | ✓ | ✓ | 1) Jenis kelamin<br>2) Asal sekolah<br>3) Jalur masuk<br>4) Nilai ujian nasional<br>5) Gaji orang tua<br>6) Indeks prestasi semester 1-4 | Model prediksi ANN dengan akurasi 79% |
| (Maghari, 2018) | 1) *Fine* KNN<br>2) *Medium* KNN<br>3) *Coarse* KNN<br>4) *Cosine* KNN<br>5) *Cubic* KNN<br>6) *Weighted* KNN | - | - | - | - | 1) Nilai 11 mata pelajaran | Model prediksi *Weighted KNN* dengan akurasi 94% |
| (Zainuddin, 2019) | 1) *Naive Bayes*<br>2) *Decision* | ✓ | - | - | ✓ | 1) Jenis kelamin<br>2) Umur | Model prediksi KNN-PSO |

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| | *Tree*<br>3) KNN<br>4) *Neural Network* | | | | | 3) Indeks prestasi semester<br>4) Status pekerjaan<br>5) Status pernikahan | dengan akurasi 74% |
| (Jamhur, 2020) | 1) *Fuzzy C-Means* | - | - | - | - | 1) Indeks prestasi semester 1-4 | Model prediksi FCM-PSO dengan akurasi 86% |
| (Nabila et al., 2021) | 1) *Fuzzy C-Means*<br>2) KNN | ✓ | - | ✓ | - | 1) Jalur penerimaan<br>2) Program studi<br>3) Asal sekolah<br>4) Nilai NEM | Model prediksi FCM-KNN dengan akurasi 71% |

Penelitian kali ini berbeda dengan penelitian terdahulu dilihat dari segi pengimplementasian metode, tahapan *preprocessing*, objek penelitian serta pemilihan atribut data. Penelitian ini mengimplementasikan dua metode yang digabungkan secara *sequence* yaitu *Fuzzy c-Means* dan *k-Nearest Neighbors*. Algoritma *Fuzzy c-Means* digunakan untuk klasterisasi data mahasiswa menjadi 3 *cluster* untuk mengurangi data *noise* dan bias, sedangkan metode *k-Nearest Neighbors* digunakan untuk prediksi status ketepatan waktu kelulusan mahasiswa yang didasarkan pada 2 atribut target; Lulus Tepat Waktu dan Lulus Tidak Tepat Waktu. Penelitian ini melakukan *data preprocessing* dalam 4 tahap; *cleaning*, *integration*, *transformation*, dan *reduction*. Objek penelitian ini menggunakan data akademik mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang tahun 2014-2018. Atribut data yang diperhitungkan antara lain indeks prestasi semester 1 sampai 5, jenis kelamin, jenis pembiayaan, serta label kelulusan (Lulus Tepat Waktu dan Lulus Tidak Tepat Waktu).

## 2.2 Landasan Teori

### 2.2.1 Ketepatan Waktu Kelulusan Mahasiswa

Ketepatan waktu kelulusan mahasiswa merupakan salah satu dari 10 indikator evaluasi pemantauan Peringkat Akreditasi Perguruan Tinggi (PEPA-PT) oleh BAN-PT (BAN-PT, 2022). Poin kesembilan dari daftar indikator evaluasi pemantauan mengatakan bahwa persentase kelulusan tepat waktu program D1, D2, D3 Sarjana Terapan, dan Sarjana yaitu minimal 37.5% untuk perguruan tinggi akademik dan 47.5% untuk perguruan tinggi vokasi. Pemenuhan indikator tersebut berpengaruh pada status peringkat akreditasi sebuah program studi. Adapun peringkat akreditasi yang diakui yaitu status Unggul (A), Baik Sekali (B), dan Baik (C).

Ketepatan waktu kelulusan mahasiswa diatur oleh program studi dalam pedoman akademik. Memperpendek masa tunggu lulusan masuk merupakan salah satu strategi untuk meningkatkan kualitas akademik di lingkungan program studi. Secara umum, seorang mahasiswa dikatakan lulus tepat waktu jika berhasil menyelesaikan kuliah dalam 8 semester dan paling lama 9 semester (Kamil & Cholil, 2020).

### 2.2.2 Data Preprocessing

Semua jenis data yang akan digunakan dalam *data mining* perlu diolah terlebih dahulu dengan *preprocessing* (Soni et al., 2011). *Preprocessing* adalah tahapan pengolahan data dari yang awalnya tidak terstruktur menjadi data yang siap dianalisis (Deviyanto & Wahyudi, 2018). Adapun proses dari *preprocessing* antara lain sebagai berikut (Shehab et al., 2021).

*1) Data cleaning*

Data *cleaning* merupakan proses pembersihan data yang tidak konsisten sehingga menyisakan data-data yang memiliki *value* dengan format seragam (Malley et al., 2019). Data di dunia nyata memiliki kemungkinan untuk bersifat tidak terstruktur, tidak lengkap, tidak valid, tidak konsisten, dan mengandung *error* atau *outliers*. Hal tersebut bisa terjadi karena adanya permasalahan teknis ketika mengentry data.

Data yang hilang (*missing data*) dapat diatasi dengan beberapa cara antara lain mengabaikan record data yang memiliki *value* tidak lengkap, mengisi *value* yang hilang secara manual, dan mengisi *value* menggunakan nilai kebanyakan dalam data keseluruhan (Son, 2006).

Data *noise* merupakan penyebutan untuk data yang memiliki random error atau varian dalam variabel yang diamati. Dalam proses *cleaning*, hal yang bisa dilakukan terhadap data *noise* adalah melakukan klasterisasi, mengimplementasikan metode *data learning*, dan *binning method* (Malley et al., 2019).

*2) Data integration*

*Data integration* merupakan proses menggabungkan data-data yang berasal dari berbagai sumber atau database. Masalah yang biasa muncul dalam proses penggabungan data yaitu kemungkinan perbedaan standar antara dua atau lebih sumber data. Sehingga yang bisa dilakukan untuk mengatasi hal tersebut antara lain menghapus, menambah, atau memodifikasi variabel tertentu yang dapat dirujuk oleh sumber data yang berbeda-beda (Malley et al., 2019).

*3) Data transformation*

*Data transformation* bertujuan untuk menyeragamkan format, skala, atau unit data sesuai kebutuhan penelitian (Malley et al., 2019). Proses ini bisa dilakukan dengan beberapa metode tergantung dengan kebutuhan penelitian; normalisasi, agregasi, dan generalisasi.

Normalisasi yaitu metode penskalaan data untuk variabel numerik berkisar antara nilai tertentu. Misalnya menggunakan rentang nilai 0 sampai 10 untuk menggambarkan skor variabel tertentu. Agregasi yaitu menggabungkan dua atau lebih *value* dari atribut yang sama menjadi satu nilai. Generalisasi misalnya mengubah properti atribut data.

*4) Data reduction*

*Data reduction* bertujuan untuk menyediakan versi kumpulan data yang lebih efektif dengan ukuran yang mudah dianalisis. Analisis yang kompleks pada kumpulan data yang besar akan memakan waktu lama. *Reduction* dilakukan dengan pengurangan data dengan kriteria tertentu, misalnya data yang mengandung duplikasi (Malley et al., 2019).

### 2.2.3 Algoritma Fuzzy c-Means

*Fuzzy C-Means* diperkenalkan oleh Jim Bezdek di tahun 1981 dan masuk ke dalam model pengelompokan *fuzzy*. FCM merupakan salah satu algoritma klasterisasi data berdasarkan derajat keanggotaan (Jamhur, 2020). Setiap data dapat menjadi anggota dari dari semua *cluster* yang terbentuk dengan tingkat derajat keanggotaan 0 dan 1. *Cluster* adalah kumpulan data yang memiliki sifat serupa dan

memiliki perbedaan jika dibandingkan dengan kumpulan data lain (Butarbutar et al., 2017).

Algoritma FCM memiliki kelebihan dan kekurangan. Algoritma FCM atau *fuzzy* jenis apapun tidak memiliki kemampuan "*learning*" tetapi dapat menjelaskan penalaran yang dilakukan berdasarkan aturan yang dimilikinya (Wahid & Girsang, 2020). Sedangkan kelebihan algoritma FCM terletak pada tahap penempatan nilai *cluster* yang tepat dibanding algoritma lain, dimana perbaikan pusat *cluster* dilakukan berulang hingga menemukan lokasi yang tepat. Oleh karena itu algoritma jenis ini sering digunakan pada tahap awal *data mining*. Hasil *cluster* yang terbentuk dapat digunakan menjadi dataset baru untuk metode selanjutnya.

Adapun tahapan algoritma FCM adalah sebagai berikut (Bezdek, 1981).

a. Tahap 1 : Input data dalam *cluster* berupa matriks.

Tabel 2.2 Contoh Data Sampel

| Data | Atribut | |
|---|---|---|
| | $j_1$ | $j_2$ |
| $i_1$ | $x_{ij}$ | $x_{ij}$ |
| $i_2$ | $x_{ij}$ | $x_{ij}$ |
| $i_n$ | $x_{ij}$ | $x_{ij}$ |

Contoh pada tabel 2.2 di atas diasumsikan sebagai data sampel berupa matriks berukuran *i* x *j*.

b. Tahap 2 : Inisialisasi parameter.

Ada beberapa parameter yang perlu diinisialisasi untuk proses perhitungan lebih lanjut dalam algoritma FCM. Adapun parameter yang dibutuhkan antara lain sebagai berikut (Jamhur, 2020).

I Parameter 1 : Jumlah *cluster* (*c*).

II Parameter 2 : Derajat pembobot (*w*).

III Parameter 3 : Iterasi maksimum (*MaxIter*).

IV Parameter 4 : Error terkecil ($\varepsilon$).

V Parameter 5 : Fungsi objektif ($P_0$).

VI Parameter 6 : Iterasi awal ($t$).

Parameter jumlah *cluster* (c) memiliki syarat pengisian nilai lebih dari 1 dan kurang dari jumlah data sampel ($1 < c < n$). Parameter derajat pembobot (w) memiliki syarat pengisian nilai lebih dari 1 ($w > 1$). Parameter iterasi maksimum (MaxIter) memiliki syarat pengisian nilai lebih dari 1 ($MaxIter > 1$) dan parameter error ($\varepsilon > 0$).

c. Tahap 3 : Inisialisasi bilangan acak sebagai nilai derajat keanggotaan data dalam *cluster*.

Inisialisasi bilangan acak berfungsi untuk mengisi nilai derajat keanggotaan data untuk masing-masing *cluster*. Jumlah *cluster* sesuai dengan inisialisasi parameter ($c$), jika $c = 3$ artinya untuk setiap data ada 3 variabel yang perlu diisi dengan bilangan acak. Inisialisasi dilakukan dengan mengisi bilangan acak antara 0 dan 1 dengan asumsi nilai 0 berarti sepenuhnya bukan anggota dari *cluster* tersebut dan nilai 1 berarti anggota sepenuhnya dari *cluster* tersebut.

Adapun contoh proses inisialisasi ditunjukkan pada tabel 2.3 berikut.

Tabel 2.3 Contoh Inisialisasi Bilangan Acak

| **No** | **Derajat Keanggotaan** | | | **Total** |
|---|---|---|---|---|
| | $\mu_{i1}$ | $\mu_{i2}$ | $\mu_{i3}$ | |
| 1 | 0.29 | 0.50 | 0.21 | 1 |
| 2 | 0.74 | 0.11 | 0.14 | 1 |
| … | … | … | … | 1 |
| *n* | *dst.* | *dst.* | *dst.* | 1 |

Bilangan acak dimasukkan ke dalam variabel $\mu$□□ (sebagai nilai *centroid* di *cluster* k). Inisialisasi bilangan acak dilakukan dengan syarat setiap data memiliki jumlah nilai derajat keanggotaan sama dengan 1

($\mu_1$+$\mu_2$+$\mu_3$+…+ $\mu_n$ = 1).

d. Tahap 4 : Hitung nilai pusat *cluster*.

Perhitungan pusat *cluster* diawali dengan memangkatkan derajat keanggotaan yang telah ditemukan di tahap sebelumnya dengan nilai *w*. Misalnya, jika w = 2 maka ( $\mu_{ik}$ )$^2$. Kemudian, mengalikan nilai data dengan nilai derajat keanggotaan yang telah dipangkatkan menggunakan persamaan (II.1).

$$(\mu_{ik})^w \,.\, x_{ij} \tag{II.1}$$

Dimana,
$\mu_{ik}$ : nilai derajat keanggotaan pada data ke-*i* dan *cluster* ke-*k*
*w* : parameter derajat pembobot
$x_{ij}$ : nilai data baris ke i kolom ke j.

Adapun perhitungan pusat *cluster* dilakukan dengan persamaan (II.2) (Bezdek, 1981).

$$v_{kj} = \frac{\sum_{i=1}^{n} \left((\mu_{ik})^w \,.\, x_{ij}\right)}{\sum_{i=1}^{n} (\mu_{ik})^w} \tag{II.2}$$

Dimana,
$v_{kj}$ : nilai *k* representasi dari pusat *cluster* dan *j* representasi dari fitur
$\mu_{ik}$ : nilai derajat keanggotaan pada data ke-*i* dan *cluster* ke-*k*
*n* : jumlah data
*w* : parameter derajat pembobot
$x_{ij}$ : nilai data ke-*i* dan dengan atribut ke-*j*.

e. Tahap 5 : Hitung fungsi objektif.

Fungsi objektif digunakan untuk mengetahui apakah error yang dihasilkan lebih kecil dari yang diharapkan. Perhitungan fungsi objektif membutuhkan elemen

nilai data, nilai pusat *cluster*, dan nilai perpangkatan dari derajat keanggotaan yang telah dihitung di tahap sebelumnya. Fungsi objektif dihitung menggunakan persamaan (II.3).

$$P_t = \sum_{i=1}^{n} \quad \sum_{k=1}^{c} \quad \left( \left[ \sum_{j=1}^{m} \quad (x_{ij} - v_{kj})^2 \right] (\mu_{ik})^w \right) \qquad \text{(II.3)}$$

Dimana,
$x_{ij}$ : nilai data baris ke *i* kolom ke *j*
$v_{kj}$ : nilai pusat *cluster*
$(\mu_{ik})^w$ : nilai pangkat derajat keanggotaan.

f. Tahap 6 : Kondisi pemberhentian iterasi.

Iterasi proses perhitungan dihentikan jika selisih fungsi objektif kurang dari error terkecil ($|P_t - P_{t-1}| < \varepsilon$) dan atau jumlah iterasi yang berjalan lebih dari jumlah iterasi maksimum ($t >$ MaxIter). Jika salah satu dan atau dua kondisi tersebut belum terpenuhi, maka ulang tahap 4 sampai 7.

g. Tahap 7 : Mencari matriks partisi dan derajat keanggotaan baru.

Matriks partisi dihitung untuk mendapatkan kelompok derajat keanggotaan baru di iterasi berikutnya. Adapun matriks partisi dapat dihitung menggunakan persamaan (II.4) berikut (Li et al., 2013).

$$L_{ik} = \left[ \sum_{j=1}^{n} \quad \left(x_{ij} - v_{kj}\right)^2 \right]^{\frac{-1}{w-1}} \qquad \text{(II.4)}$$

Dimana,
$L_{ik}$ : nilai *i* representasi dari matriks partisi dan *k* representasi dari *cluster*.
$x_{ij}$ : nilai data baris ke *i* kolom ke *j*
$v_{kj}$ : nilai pusat *cluster*
$w$ : parameter derajat pembobot.

Sedangkan, derajat keanggotaan baru suatu data dapat dihitung dengan membagi nilai elemen matriks partisi data dengan jumlah elemen matriks partisi

data tersebut dalam semua *cluster*. Adapun derajat keanggotaan baru dihitung menggunakan persamaan (II.5) (Li et al., 2013)

$$\mu_{ik}(\text{baru}) = \frac{L_{ik}}{\sum_{k=1}^{c} L_{ik}}$$

$$\mu_{ik}(\text{baru}) = \frac{\left[\sum_{j=1}^{n} \left(X_{ij} - V_{kj}\right)^2\right]^{\frac{-1}{w-1}}}{\sum_{k=1}^{c} \left[\sum_{j=1}^{n} \left(X_{ij} - V_{kj}\right)^2\right]^{\frac{-1}{w-1}}} \quad \text{(II.5)}$$

h. Tahap 8 : Pengelompokan data sebagai anggota *cluster*.

Dalam algoritma FCM, sebuah data akan masuk sebagai anggota suatu *cluster* jika memiliki nilai jarak partisi matriks ($\mu_{\square\square}$) maksimum terhadap nilai pusat *cluster*-nya (Butarbutar et al., 2017).

Adapun output tahapan algoritma FCM yaitu kumpulan data yang sudah terbagi menjadi beberapa *cluster* dan nilai pusat masing-masing *cluster* pada iterasi terakhir.

### 2.2.4 Algoritma k-Nearest Neighbors

*K-Nearest Neighbors* merupakan salah satu metode *data mining* yang masuk ke dalam fungsi klasifikasi. KNN merupakan algoritma *supervised learning*. Metode ini mengklasifikasi data individu berdasarkan tetangga (*neighbors*) terdekat (Nabila et al., 2021). Tetangga terdekat berkontribusi lebih pada proses perhitungan daripada tetangga yang jauh (Kusiak et al., 2012).

Prediksi dalam KNN didasarkan pada asumsi naif dan entitas dengan jarak yang sama. Dalam proses prediksi, metode KNN terdiri atas pengklasifi (*classifiers*) dan regresi (*regression*) (Adithiyaa et al., 2020).

KNN sebagai *classifiers* menghasilkan pola klasifikasi yang digunakan untuk mempelajari data yang diproses dan untuk memprediksi suatu kasus. Pengklasifikasian dalam KNN dilakukan dengan pemilihan nilai *k* dan metrik jarak *euclidean.* Pemilihan nilai *k* sangat mempengaruhi kinerja KNN. Nilai *k* yang terlalu kecil membuat proses klasifikasi terpengaruh data *noise* dan jika terlalu besar batasan klasifikasi akan menjadi kabur. Adapun pemilihan nilai *k* yang baik dapat dilakukan misalnya menggunakan *k-fold cross validation* (Banjarsari et al., 2016). Sedangkan KNN sebagai *regression* memanfaatkan data yang sudah diolah untuk memprediksi data yang akan datang. Regresi berkaitan dengan prediksi hasil variabel yang sudah melalui tahap *preprocessing* dengan set variabel independen yang diberikan (Adithiyaa et al., 2020).

Metode KNN memiliki kelebihan antara lain komputasi pelatihan cepat, sederhana, tahan terhadap data yang *noise*, dan sangat efektif jika data *train* besar. Tetapi metode ini memiliki kelemahan dalam waktu proses klasifikasi karena lebih lambat dari metode lainnya. Oleh karena itu, KNN membutuhkan *backup* untuk mengakselerasi performansi klasterisasi data (Priandini et al., 2017). Adapun salah satu yang dapat dilakukan yaitu menggabungkan dengan metode FCM.

Adapun tahapan algoritma KNN dalam metode FCM-KNN adalah sebagai berikut (Sun et al., 2009).

a. Tahap 1 : Input dataset dan nilai pusat *cluster.*

Dataset yang diinputkan terdiri atas data *training* dan data *testing*. Data *training* didapatkan dari hasil proses klasterisasi data menggunakan algoritma FCM. Sehingga, data yang diinputkan dalam algoritma KNN memiliki elemen

atribut, *cluster* dan label. Selain data *training* dan data *testing* elemen yang perlu diinputkan yaitu nilai pusat masing-masing *cluster* yang juga didapat dari proses FCM.

b. Tahap 2 : Identifikasi pusat *cluster* terdekat

Tahap selanjutnya yaitu mencari *cluster* terdekat dengan data *testing*. Pencarian *cluster* terdekat dilakukan dengan menghitung jarak *euclidean* data *testing* dan nilai pusat *cluster*. Jarak *euclidean* bisa dihitung menggunakan persamaan (II.6) berikut.

$$\sqrt[d_i]{\sum_{i=1}^{n} (x_i - c_i)^2} \quad \text{(II.6)}$$

Dimana,
$x_i$ : data *testing* pada atribut ke-$i$
$c_i$ : pusat *cluster* pada atribut ke-$i$
$n$ : dimensi data.

Pencarian pusat *cluster* terdekat bertujuan untuk memilih *cluster* yang anggotanya akan dibandingkan dengan data *testing*. Masing-masing data *testing* hanya akan dibandingkan dengan anggota-anggota dalam *cluster* terpilih karena dataset sudah di klasterisasi sebelumnya.

c. Tahap 3 : Hitung jarak data dengan data tetangga (*neighbors*) dalam *cluster* terpilih.

Perhitungan nilai jarak antara data *testing* dengan data-data pada *cluster* terpilih dilakukan menggunakan persamaan *euclidean* (II.4). Kemudian data diurutkan dari nilai jarak yang terkecil.

d. Tahap 4 : Penetapan nilai *k* dan prediksi kelas data *testing*.

Perhitungan nilai *k*-tetangga terdekat hanya dilakukan terhadap anggota *cluster* terpilih sebab telah dilakukan proses klasterisasi sebelumnya. Hal ini akan mempercepat proses komputasi karena nilai *k* tidak perlu dibandingkan dengan seluruh data dalam dataset (Nabila et al., 2021). Pemilihan kelas data menggunakan konsep modus pada data *k*-tetangga.

Adapun output dari tahapan metode KNN yaitu prediksi label atau kelas data *testing* berdasarkan skenario nilai *k-neighbors* yang ditetapkan.

## 2.3 Integrasi Islam

Pengimplementasian optimasi *k-Nearest Neighbors* menggunakan *Fuzzy c-Means* pada ketepatan waktu kelulusan mahasiswa menunjukkan kemajuan penerapan bidang teknologi dan pengetahuan. Penerapan teknologi dan ilmu pengetahuan disinggung dalam Islam melalui beberapa dalil al-Qur'an berikut.

وَعِنْدَهٗ مَفَاتِحُ الْغَيْبِ لَا يَعْلَمُهَآ اِلَّا هُوَّ ۗ وَيَعْلَمُ مَا فِى الْبَرِّ وَالْبَحْرِّ ۗ وَمَا تَسْقُطُ مِنْ وَّرَقَةٍ اِلَّا يَعْلَمُهَا وَلَا حَبَّةٍ فِيْ ظُلُمٰتِ الْاَرْضِ وَلَا رَطْبٍ وَّلَا يَابِسٍ اِلَّا فِيْ كِتٰبٍ مُّبِيْنٍ (59)

*"Dan kunci-kunci semua yang gaib ada pada-Nya; tidak ada yang mengetahuinya selain Dia. Dia mengetahui apa yang ada di darat dan di lautan. Tidak ada sehelai daun pun yang gugur yang tidak diketahui-Nya, tidak ada sebutir biji pun dalam kegelapan bumi dan tidak pula sesuatu yang basah atau yang kering, yang tidak tertulis dalam kitab yang nyata."* (QS. al-An'am: 59)

Berdasarkan Tafsir Tahlili yang bersumber dari website resmi Kementerian Agama, yang dimaksud dengan *"yang gaib"* dalam QS. al-An'am ayat 59 yaitu sesuatu yang tidak diketahui hakikat sebenarnya, sekalipun manusia telah diberi pengetahuan yang banyak oleh Allah, tetapi pengetahuan tersebut sedikit jika dibandingkan pengetahuan-Nya (quran.kemenag.go.id., 2022). Salah satu

pengetahuan yang diturunkan Allah berupa ilmu pengetahuan dan teknologi yang dapat digunakan untuk mempelajari, memahami, dan memprediksi beberapa hal. Dalam konteks prediksi ketepatan waktu kelulusan mahasiswa menggunakan metode Fuzzy C-Means dan K-Nearest Neighbors, pengetahuan dan teknologi dapat membantu dalam memprediksi hasil akademik mahasiswa..

اَمَّنْ هُوَ قَانِتٌ اٰنَاۤءَ الَّيْلِ سَاجِدًا وَّقَاۤىِٕمًا يَّحْذَرُ الْاٰخِرَةَ وَيَرْجُوْا رَحْمَةَ رَبِّهٖۗ قُلْ هَلْ يَسْتَوِى الَّذِيْنَ يَعْلَمُوْنَ وَالَّذِيْنَ لَا يَعْلَمُوْنَۗ اِنَّمَا يَتَذَكَّرُ اُولُوا الْاَلْبَابِ (9)

*"(Apakah kamu orang musyrik yang lebih beruntung) ataukah orang yang beribadah pada waktu malam dengan sujud dan berdiri, karena takut kepada (azab) akhirat dan mengharapkan rahmat Tuhannya ? Katakanlah, "Apakah sama orang-orang yang tidak mengetahui ?" Sebenarnya hanya orang yang berakal sehat yang dapat menerima pelajaran."* (QS. al-Zumar: 9)

Melalui pendekatan Tafsir Ijmali, surat al-Zumar ayat 9 berorientasi pada tujuan adanya perubahan ke arah yang lebih baik dari pribadi seorang manusia yang menuntut ilmu (Wahidah, 2019). Dalam konteks prediksi ketepatan waktu kelulusan mahasiswa, ayat ini berhubungan dengan pentingnya para mahasiswa untuk menjadi orang yang berakal dan mengambil pelajaran dari pengalaman serta informasi yang tersedia. Prediksi ketepatan waktu kelulusan mahasiswa melibatkan analisis data historis, pemodelan, dan algoritma yang kompleks. Mahasiswa yang berakal akan memanfaatkan prediksi tersebut untuk mengoptimalkan kinerja akademik mereka dan meningkatkan kemungkinan kelulusan tepat waktu.

اِنَّ فِيْ خَلْقِ السَّمٰوٰتِ وَالْاَرْضِ وَاخْتِلَافِ الَّيْلِ وَالنَّهَارِ وَالْفُلْكِ الَّتِيْ تَجْرِيْ فِى الْبَحْرِ بِمَا يَنْفَعُ النَّاسَ وَمَآ اَنْزَلَ اللّٰهُ مِنَ السَّمَاۤءِ مِنْ مَّاۤءٍ فَاَحْيَا بِهِ الْاَرْضَ بَعْدَ مَوْتِهَا وَبَثَّ فِيْهَا مِنْ كُلِّ دَاۤبَّةٍۖ وَّتَصْرِيْفِ الرِّيٰحِ وَالسَّحَابِ الْمُسَخَّرِ بَيْنَ السَّمَاۤءِ وَالْاَرْضِ لَاٰيٰتٍ لِّقَوْمٍ يَّعْقِلُوْنَ (164)

*"Sesungguhnya pada penciptaan langit dan bumi, pergantian malam dan siang, kapal yang berlayar di laut dengan (muatan) yang bermanfaat bagi manusia, apa yang diturunkan Allah dari langit berupa air, lalu dengan itu dihidupkan-Nya bumi setelah mati (kering), dan Dia tebarkan di dalamnya bermacam-macam binatang, dan perkisaran angin dan awan yang dikendalikan antara langit dan bumi, (semua itu) sungguh, merupakan tanda-tanda (kebesaran Allah) bagi orang-orang yang mengerti."* (QS. al-Baqarah: 164)

Dalam telaah konsep ontologi berdasarkan pendekatan filsafat ilmu, maksud dari kalimat *"apa yang Allah turunkan dari langit berupa air"* yaitu bahwa Allah melakukan seluruh penciptaan baik di bumi dan langit serta menurunkan berkah sehingga makhluk bisa hidup (Fauziah, 2020). Berkah tersebut dapat berupa apapun di alam semesta yang dapat diamati dan dipelajari. Dalam prediksi ketepatan waktu kelulusan mahasiswa, penggunaan metode dan teknologi seperti analisis data, algoritma prediksi merupakan pengimplementasian teknologi yang juga merupakan berkah Allah. Sehingga, prediksi tidak dapat memberi hasil mutlak dan tidak dapat menggantikan kebijaksanaan dan kehendak Allah. Hasil penelitian juga perlu digunakan untuk membantu evaluasi akademik bagi mahasiswa dan program studi dengan penuh tanggung jawab sebagai bentuk menghargai kebijaksanaan-Nya.

# BAB III

# METODOLOGI PENELITIAN

Pada bagian metodologi penelitian dijelaskan tahapan runtut yang dilakukan selama penelitian. Tahapan tersebut terdiri atas pengumpulan data penelitian desain sistem, *data preprocessing*, pelatihan model menggunakan algoritma *Fuzzy c-Means* dan algoritma *k-Nearest Neighbors*, serta skenario uji coba.

## 3.1 Data Penelitian

### 3.1.1 Data Primer

Data primer dalam penelitian ini yaitu data akademik mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang tahun 2014-2018 yang berjumlah 631 *records*. Data didapatkan dari BAK UIN Malang melalui narasumber M.Abror pada tanggal 2 Mei 2023 dengan file format *excel*.

Adapun data yang digunakan dalam penelitian meliputi atribut-atribut berikut.

1. Indeks prestasi semester (IPS 1-5).
2. Jenis kelamin (PEREMPUAN, LAKI-LAKI).
3. Jenis pembiayaan (BIDIKMISI, NON BIDIK MISI).
4. Label (LULUS TEPAT WAKTU, LULUS TIDAK TEPAT WAKTU).

Pemilihan atribut data didasarkan pada beberapa pertimbangan. Menurut pedoman pendidikan Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang, distribusi mata kuliah wajib ada pada semester 1-5,

sedangkan semester 6 sudah ada penyisipan mata kuliah pilihan, PKL, dan Skripsi (Pedoman Pendidikan, 2017). Hal tersebut diasumsikan pada semester 1-5 kumpulan data mahasiswa masih memiliki data IPS dengan jumlah SKS yang relatif sama sehingga data tidak memiliki banyak *outliers* dan tidak banyak yang otomatis disisihkan ketika memasuki proses *preprocessing*.

Pemilihan atribut jenis kelamin didasarkan pada ilmu psikologi tentang pengaruh perbedaan jenis kelamin pada kemampuan intelektual. Menurut (Santrock, 2014) tidak ada perbedaan kemampuan intelektual menyeluruh antara perempuan dan laki-laki, tetapi perbedaan muncul pada beberapa area kognitif. Laki-laki sedikit lebih unggul dalam sains jika dibandingkan perempuan (Suprapto et al., 2018). Hal tersebut mendasari pemilihan atribut jenis kelamin karena memiliki kemungkinan pengaruh dalam ketepatan waktu kelulusan mahasiswa, terutama yang mempelajari bidang sains. Selain itu, pemilihan atribut jenis kelamin juga didasarkan pada penggunaannya di penelitian sebelumnya (Zainuddin, 2019).

Pemilihan atribut jenis pembiayaan didasarkan pada syarat-syarat bidikmisi yang mendukung ketepatan waktu kelulusan mahasiswa. Pada website bidikmisi resmi Kementerian Riset, Teknologi, dan Pendidikan Tinggi (RISTEKDIKTI) dinyatakan, mahasiswa dengan jenis pembiayaan bidikmisi dibebankan beberapa hal. Monitoring indeks prestasi oleh PT Belmawa Kemenristekdikti dengan syarat harus stabil dari semester ke semester, dibina untuk lulus tepat waktu, menjadi mahasiswa aktif, serta selalu memenuhi syarat akademik yang ditetapkan (Ristekdikti, 2019).

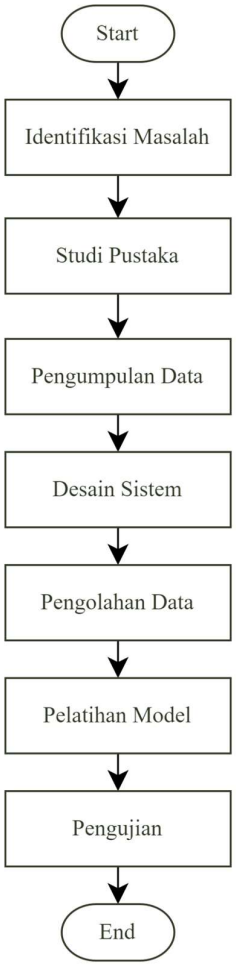
### 3.1.2 Data Sekunder

Data sekunder dalam penelitian ini merupakan data-data pendukung penelitian. Adapun item data dan sumber data sebagai berikut.

1. Data jumlah mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang setiap tahun dalam kurun waktu 2014-2018, didapat dari website Pddikti.kemdikbud.go.id.
2. Data tahun lulus masing-masing mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang setiap tahun dalam kurun waktu 2014-2018, didapat dari website Pddikti.kemdikbud.go.id.
3. Data referensi dan data penelitian-penelitian terdahulu, didapat dari jurnal ilmiah yang disitasi oleh *Google Scholar*, *IEEE*, *Springer*, dan sejenisnya.

## 3.2 Prosedur Penelitian

Prosedur penelitian berisi rangkaian tahap yang akan dilakukan, disusun secara sistematis untuk mencapai tujuan penelitian. Adapun prosedur pada penelitian ini diilustrasikan oleh bagan pada gambar 3.1 berikut.



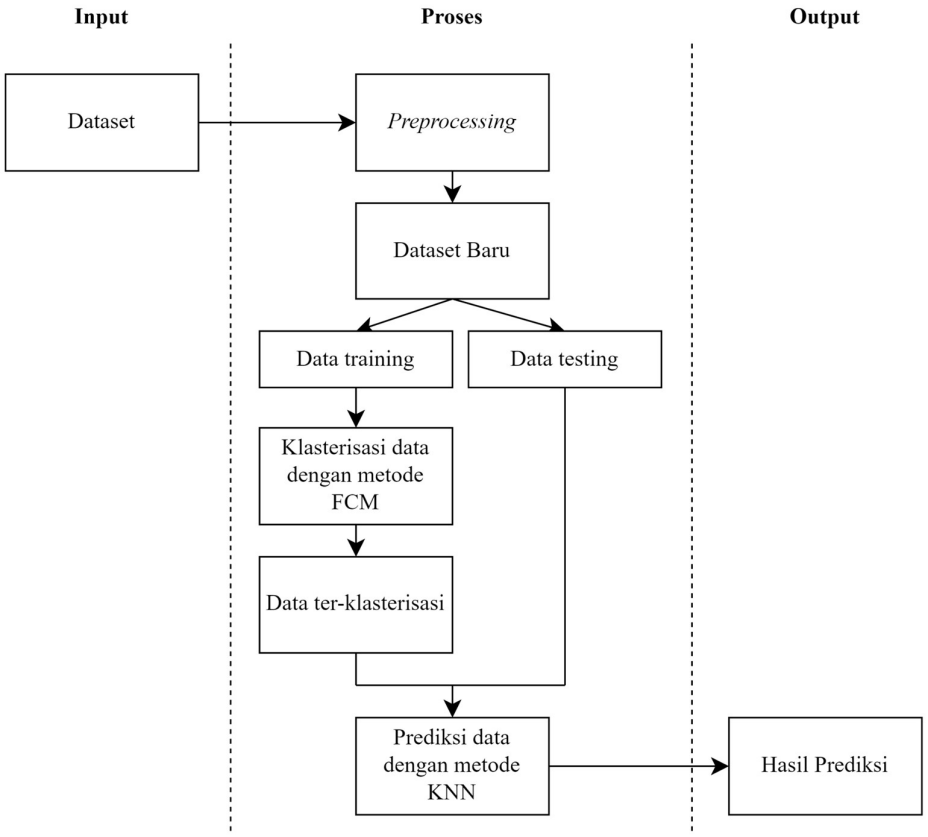

Gambar 3. 1 Bagan Prosedur Penelitian

Berdasarkan gambar di atas, prosedur penelitian dimulai dengan identifikasi masalah. Pada tahap identifikasi masalah dilakukan perumusan latar belakang masalah, masalah yang coba diselesaikan, dan batasan masalah. Tahap studi pustaka, dilakukan proses literasi referensi yang didapat dari jurnal ilmiah, skripsi, tesis, ataupun disertasi yang memiliki kemiripan topik dan tema dengan penelitian yang dilakukan. Tahap pengumpulan data bertujuan untuk mengumpulkan data-data yang dijadikan data primer dan sekunder dalam penelitian

ini. Data primer merupakan data utama yang akan diolah dalam penelitian, sedangkan data sekunder merupakan data pembanding antara penelitian yang dilakukan dengan penelitian-penelitian sebelumnya. Pada tahap perancangan atau desain sistem dilakukan penggambaran skema sistem yang terdiri atas input, proses dan output. Pada tahap pengolahan data dilakukan *data preprocessing* untuk menyeragamkan bentuk dan atribut data sebelum diproses sistem. Pada tahap pelatihan model dilakukan implementasi metode FCM-KNN terhadap dataset. Pada tahap pengujian dilakukan implementasi desain sistem yang telah dibuat. Setelah itu, dilakukan analisis hasil pengujian dan membuat kesimpulan dengan melakukan perhitungan akurasi implementasi metode dalam sistem yang telah dibuat.

## 3.3 Desain Sistem

Desain sistem merupakan gambaran skema alur sistem yang akan dijadikan acuan perancangan sistem oleh peneliti. Adapun desain sistem penelitian ini ditunjukkan oleh diagram blok pada gambar 3.2 berikut.




Gambar 3.2 Bagan Desain Sistem

Pada gambar 3.2 di atas dapat dilihat bahwa sistem dijalankan dengan tiga langkah utama.

1) Input

Tabel 3.1 Desain Input

| Nama Proses | Bentuk | Deskripsi | Data |
|---|---|---|---|
| Input dataset | Data text | Memasukkan data mahasiswa ke dalam tabel. | NIM, IPS 1-5, Jenis Kelamin, Jenis Pembiayaan |

2) Proses

Tabel 3.2 Desain Proses

| Nama Proses | Deskripsi | Output Sementara |
|---|---|---|
| *Data preprocessing* | Menyeragamkan format data dan mengurangi data yang noise atau tidak valid. | Data numerik dengan format seragam. |
| klastering data dengan FCM | Melakukan perhitungan klastering data yang serupa dalam beberapa *cluster* kemudian diberi label. | Dataset baru, Label data, Nilai pusat masing-masing cluster. |
| Proses pemilihan *cluster* | Mengambil dataset dan menghitung jarak data *testing* ke pusat *cluster.* Kemudian memilih *cluster* terdekat. | List anggota *cluster* terdekat. |
| Proses prediksi KNN | Membandingkan jarak *euclidean* antar data anggota *cluster* terpilih. Memilih jarak terdekat sesuai dengan nilai *k*-neighbors. | Prediksi label ketepatan waktu kelulusan. |

3) Output

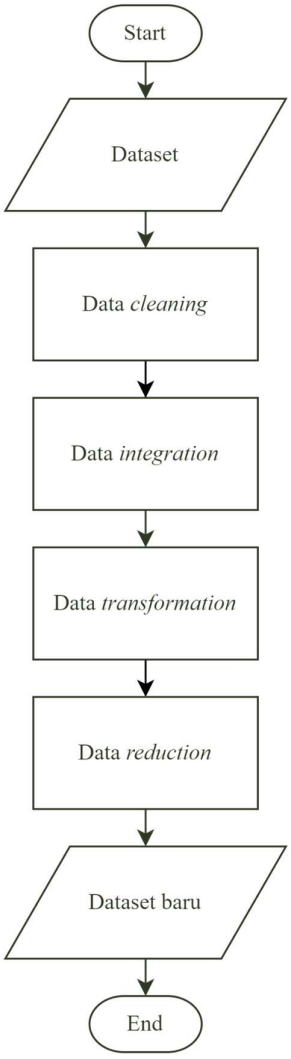
Tabel 3.3 Desain Output

| Jenis Output | Informasi yang Ditampilkan | Bentuk |
|---|---|---|
| Prediksi ketepatan waktu kelulusan | Prediksi mahasiswa lulus tepat waktu dan mahasiswa tidak lulus tepat waktu. | *Report* |
| Performa metode | Tingkat akurasi metode FCM-KNN dalam klastering dan memprediksi data. | Persentase, Diagram |

## 3.4 Data Preprocessing

Sebelum diinputkan ke dalam sistem, data diolah terlebih dahulu menggunakan teknik *preprocessing*. Semua jenis data yang akan digunakan dalam *data mining* perlu diolah terlebih dahulu untuk mengurangi data yang bias dan noise (Soni et al., 2011). *Data preprocessing* mengubah format data dari yang awalnya tidak terstruktur menjadi data yang siap dianalisis (Deviyanto & Wahyudi, 2018). Penelitian ini menggunakan 4 langkah *preprocessing* antara lain *data cleansing*, *data integration*, *data transformation*, dan *data reduction*.

Di bawah ini merupakan *flowchart* proses *text preprocessing* yang berjalan pada sistem.




Gambar 3.3 Bagan *Data Preprocessing*

Dapat dilihat dari gambar 3.3 proses pertama yang dilakukan dalam *preprocessing* yaitu menginputkan dataset. *Preprocessing* manual pada penelitian ini akan menggunakan 12 data *dummy* sebagai sampel. Data sampel dapat dilihat pada tabel 3.4 berikut.

Tabel 3.4 Data Sampel

| No | NIM | IP | | | | | Jenis Kelamin | Jenis Pembiayaan |
|---|---|---|---|---|---|---|---|---|
| | | S.1 | S.2 | S.3 | S.4 | S.5 | | |
| 1 | 14650021 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | L | Bidikmisi |
| 2 | 14650044 | 2.38 | 2.69 | 0 | 0 | 0 | - | Non Bidikmisi |
| 3 | 14650034 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | L | Non Bidikmisi |
| 4 | 15650100 | 3.65 | 3.70 | 3.39 | 3.30 | 3.25 | L | - |
| 5 | 15650099 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | P | Bidikmisi |
| 6 | 16650076 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | L | Bidikmisi |
| 7 | 17650105 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | L | Bidikmisi |
| 8 | 17650032 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | L | Non Bidikmisi |
| 9 | 18650024 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | P | Non Bidikmisi |
| 10 | 18650066 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | L | Bidikmisi |
| 11 | 18650028 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | L | Bidikmisi |
| 12 | 18650001 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | L | Non Bidikmisi |

### 3.4.1 Data Cleaning

*Data cleaning* merupakan proses pembersihan data sehingga menyisakan data-data yang memiliki *value* dengan format seragam dan konsisten (Malley et al., 2019). Data di dunia nyata memiliki kemungkinan untuk bersifat tidak terstruktur, tidak lengkap, tidak valid, tidak konsisten, dan mengandung *error* atau *outliers*.

Contoh tahap *cleaning* terhadap data sampel pada Tabel 3.4 yaitu penghapusan data mahasiswa dengan nim 14650044 dan 15650100 karena terdapat beberapa atribut yang tidak terisi, tidak valid, dan berpotensi memiliki *outlier* (nilainya terlalu berbeda dengan data yang lain).

Berikut output data setelah melalui tahap *data cleaning* ditunjukkan pada tabel 3.5.

Tabel 3.5 Hasil *Data Cleaning*

| No | NIM | IP | | | | | Jenis Kelamin | Jenis Pembiayaan |
|---|---|---|---|---|---|---|---|---|
| | | S.1 | S.2 | S.3 | S.4 | S.5 | | |
| 1 | 14650021 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | L | Bidikmisi |
| 2 | 14650034 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | L | Non Bidikmisi |
| 3 | 15650099 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | P | Bidikmisi |
| 4 | 16650076 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | L | Bidikmisi |
| 5 | 17650105 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | L | Bidikmisi |
| 6 | 17650032 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | L | Non Bidikmisi |
| 7 | 18650024 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | P | Non Bidikmisi |
| 8 | 18650066 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | L | Bidikmisi |
| 9 | 18650028 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | L | Bidikmisi |
| 10 | 18650001 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | L | Non Bidikmisi |

### 3.4.3 Data Integration

*Data integration* merupakan proses menggabungkan data-data yang berasal dari berbagai sumber atau database. Masalah yang biasa muncul dalam proses penggabungan data yaitu kemungkinan perbedaan standar antara dua atau lebih sumber data. Sehingga yang bisa dilakukan untuk mengatasi hal tersebut antara lain menghapus, menambah, atau memodifikasi variabel tertentu yang dapat dirujuk oleh sumber data yang berbeda-beda (Malley et al., 2019).

Contoh tahap *integration* terhadap data sampel pada tabel 3.4 dilakukan dengan menghapuskan atribut yang tidak dibutuhkan dalam proses *mining*, antara lain atribut NIM. Penyesuaian atribut dimaksudkan untuk meminimalisir konflik nilai data dan redundansi ketika masuk dalam perhitungan metode FCM. Perhitungan FCM hanya membutuhkan nilai data dari atribut IPS, Jenis Kelamin, dan Jenis Pembiayaan.

Berikut output data setelah melalui tahap *data integration* ditunjukkan oleh tabel 3.6.

Tabel 3.6 Hasil *Data Integration*

| No | IP | | | | | Jenis Kelamin | Jenis Pembiayaan |
|---|---|---|---|---|---|---|---|
| | S.1 | S.2 | S.3 | S.4 | S.5 | | |
| 1 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | L | Bidikmisi |
| 2 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | L | Non Bidikmisi |
| 3 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | P | Bidikmisi |
| 4 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | L | Bidikmisi |
| 5 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | L | Bidikmisi |
| 6 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | L | Non Bidikmisi |
| 7 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | P | Non Bidikmisi |
| 8 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | L | Bidikmisi |
| 9 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | L | Bidikmisi |
| 10 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | L | Non Bidikmisi |

### 3.4.4 Data Transformation

*Data transformation* bertujuan untuk menyeragamkan format, skala, atau unit data sesuai kebutuhan penelitian (Malley et al., 2019). Proses ini bisa dilakukan dengan beberapa metode tergantung dengan kebutuhan penelitian; normalisasi, agregasi, dan generalisasi.

Contoh tahap *transformation* terhadap data sampel pada tabel 3.4 dilakukan dengan penyeragaman jenis data dan perubahan properti atribut. Data kategorikal pada atribut ‘Jenis Kelamin’ dan ‘Jenis Pembiayaan’ perlu diubah menjadi data numerik. Hal tersebut karena metode FCM-KNN hanya menerima input numerik. Perubahan jenis data kategorikal menjadi bentuk numerik berupa angka desimal disebut dengan teknik *Label Encoding* (Dahlan, 2022). Dalam fase pengkodean, suatu nilai diubah menjadi nilai baru melalui beberapa opsi operasi; *one-hot conversion*, encoding linier, dan operasi kaskade (Jia & Zhang, 2021). Label encoding termasuk dalam encoding linier dimana pengkodean dilakukan

dengan memberikan kode angka desimal pada data kategorikal yaitu dari 0 sampai *n* (Latifah et al., 2019).

Adapun proses transformasi data yang dilakukan yaitu sebagai berikut.

Tabel 3.7 *Label Encoding*

| Atribut | Kode |
|---|---|
| Jenis Kelamin | 1) Laki-laki = 1<br>2) Perempuan = 2 |
| Jenis Pembiayaan | 1) Bidikmisi = 1<br>2) Non Bidikmisi = 2 |

Pengkodean atribut jenis kelamin pada tabel 3.7 di atas didasarkan pada aturan format resmi oleh salah satu badan milik negara, BPJS, yaitu kode 1 untuk laki-laki dan kode 2 untuk perempuan. Selain itu, pengkodean tersebut juga didasarkan pada penelitian sebelumnya (Zainuddin, 2019). Sedangkan untuk pengkodean atribut jenis pembiayaan belum pernah dikodekan sebelumnya, sehingga pada penelitian ini digunakan kode 1 untuk jenis pembiayaan bidikmisi dan kode 2 untuk jenis pembiayaan non bidikmisi.

Tabel 3.8 Transformasi Properti Atribut Data

| Atribut Lama | Atribut Baru |
|---|---|
| IPS 1 | ip_1 |
| IPS 2 | ip_2 |
| IPS 3 | ip_3 |
| IPS 4 | ip_4 |
| IPS 5 | ip_5 |
| Jenis Kelamin | jk |
| Jenis Pembiayaan | jp |

Setelah dilakukan transformasi pada data kategorikal dan nama properti atribut data, berikut merupakan hasil dari tahapan *data transformation*.

Tabel 3.9 Hasil *Data Transformation*

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
|---|---|---|---|---|---|---|---|
| D1 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | 1 | 1 |
| D2 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | 2 | 2 |
| D3 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | 2 | 1 |
| D4 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | 2 | 1 |
| D5 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | 1 | 1 |
| D6 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | 1 | 1 |
| D7 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | 2 | 2 |
| D8 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | 1 | 1 |
| D9 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | 1 | 1 |
| D10 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | 1 | 2 |

### 3.4.2 Data Reduction

*Data reduction* bertujuan untuk menyediakan versi kumpulan data yang lebih efektif dengan ukuran yang mudah dianalisis. Analisis yang kompleks pada kumpulan data yang besar akan memakan waktu lama. *Reduction* dilakukan dengan pengurangan data dengan kriteria tertentu, misalnya data yang mengandung duplikasi (Malley et al., 2019).

Tahapan *reduction* dilakukan dengan menghapus data yang dobel untuk mengurangi ukuran dimensi data. Tetapi, jenis data akademik mahasiswa yang digunakan dalam penelitian ini tidak memiliki duplikasi data karena *primary key* nya adalah atribut NIM. Sehingga tahapan *reduction* dilakukan dengan menghapuskan data-data mahasiswa yang memiliki status studi Non Aktif, Mutasi, dan Pernah Studi. Data-data tersebut diasumsukan tidak termasuk kedalam kategori data mahasiswa yang pernah lulus dari program studi terkait.

Adapun tabel 3.10 di bawah ini merupakan hasil dari *data preprocessing* yang dilakukan.

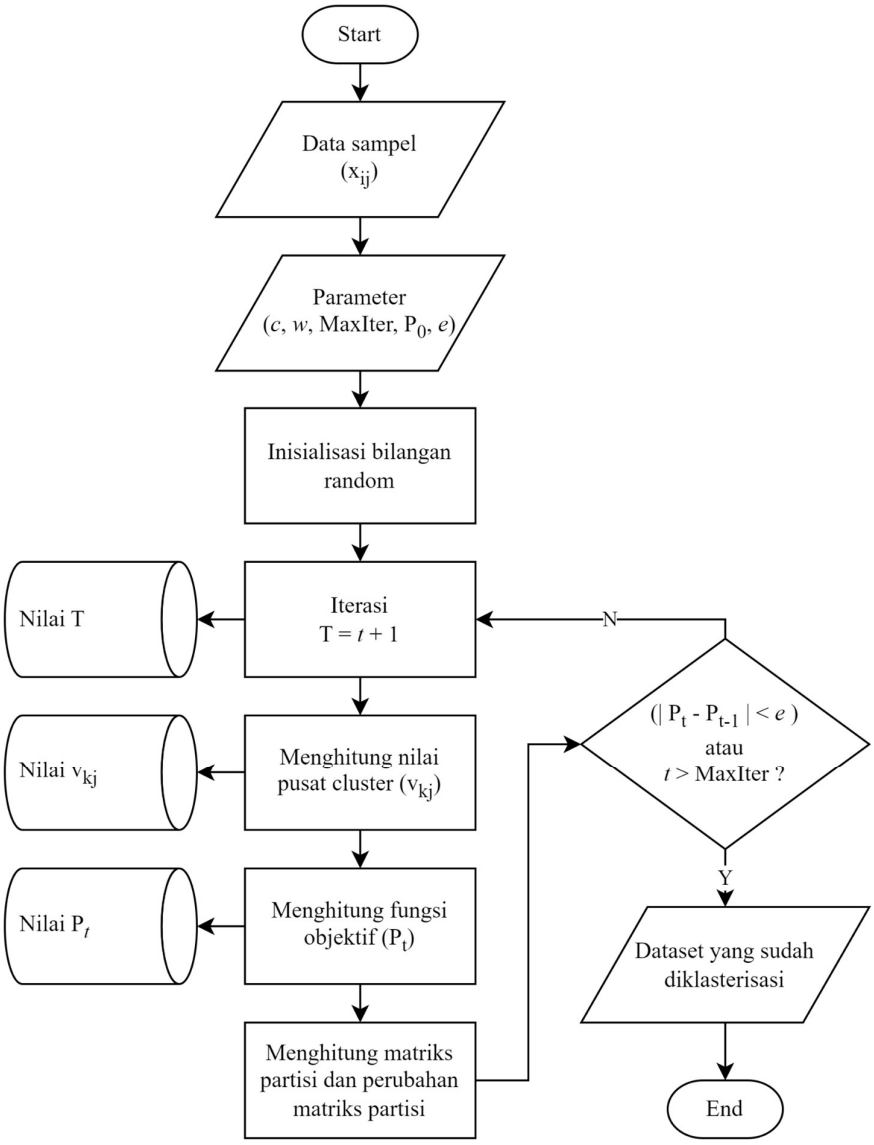
Tabel 3.10 Hasil *Data Preprocessing*

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
|---|---|---|---|---|---|---|---|
| D1 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | 1 | 1 |
| D2 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | 2 | 2 |
| D3 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | 2 | 1 |
| D4 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | 2 | 1 |
| D5 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | 1 | 1 |
| D6 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | 1 | 1 |
| D7 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | 2 | 2 |
| D8 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | 1 | 1 |
| D9 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | 1 | 1 |
| D10 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | 1 | 2 |

## 3.5 Algoritma Fuzzy c-Means

Dalam penelitian ini, algoritma *Fuzzy c-Means* digunakan untuk klastering data sebelum digunakan sebagai dataset untuk metode KNN. Data yang telah melalui proses *preprocessing* akan dikelompokkan menjadi beberapa *cluster* berdasarkan derajat keanggotaan. klastering membantu mengurangi data noise dan data bias sehingga dapat meningkatkan performansi metode selanjutnya. Selain itu, klastering data menggunakan *Fuzzy c-Means* memiliki kelebihan pada tahap penempatan nilai cluster yang tepat dibanding algoritma lain, dimana perbaikan pusat cluster dilakukan berulang hingga menemukan lokasi yang tepat.

Di bawah ini merupakan bagan proses algoritma *Fuzzy c-Means* yang berjalan pada sistem.




Gambar 3.4 Bagan Algoritma *Fuzzy c-Means*

Berikut sistem yang berjalan ketika klastering data menggunakan *Fuzzy c-Means* :

a. Input data.

Data yang diinputkan adalah data yang sudah melalui tahap *preprocessing* yaitu data matriks dalam tabel 3.10. Data berjumlah 10 (D1-D10) dan memiliki 7 atribut.

b. Inisialisasi parameter.

Parameter yang dibutuhkan dalam implementasi algoritma *Fuzzy c-Means* antara lain sebagai berikut.

Tabel 3.11 Inisialisasi Parameter

| *Cluster* *(c)* | Derajat Pembobot *(w)* | Iterasi *(MaxIter)* | Error *(ε)* | Fungsi Objektif *( $P_0$ )* | Iterasi Awal *(t)* |
|---|---|---|---|---|---|
| 3 | 2 | 1000 | $10^{-5}$ | 0 | 1 |

Berdasarkan contoh inisialisasi parameter di atas, data akan dikelompokkan menjadi 3 *cluster*. Iterasi diawali dengan nilai $t$ = 1, sedangkan iterasi maksimum yang akan dilakukan adalah 1000. Error terkecil yang diharapkan dari proses klastering yaitu $10^{-5}$. Hal tersebut berarti iterasi dalam proses klastering akan berhenti jika error sudah lebih kecil dari $10^{-5}$ atau iterasi sudah melebihi 1000.

c. Inisialisasi nilai derajat keanggotaan data pada masing-masing *cluster*.

Derajat keanggotaan diinisialisasi dengan bilangan *random* pada iterasi ke-1, dengan syarat $\mu_{□1} + \mu_{□2} + \mu_{□3} = 1$. Sedangkan untuk iterasi ke-2 dan seterusnya nilai derajat keanggotaan ditentukan dari pembentukan matriks partisi.

Tabel 3.12 berikut menunjukkan derajat keanggotaan data D1-D10 pada ketiga *cluster* di iterasi ke-17.

Tabel 3.12 Derajat Keanggotaan Data pada Iterasi Terakhir

| Data | Derajat Keanggotaan | | | Total |
|---|---|---|---|---|
| | $\mu_{i1}$ | $\mu_{i2}$ | $\mu_{i3}$ | |
| D1 | 0.12 | 0.36 | 0.52 | 1 |
| D2 | 0.03 | 0.08 | 0.90 | 1 |
| D3 | 0.55 | 0.29 | 0.16 | 1 |
| D4 | 0.24 | 0.58 | 0.17 | 1 |
| D5 | 0.09 | 0.83 | 0.07 | 1 |
| D6 | 0.83 | 0.12 | 0.05 | 1 |
| D7 | 0.05 | 0.10 | 0.85 | 1 |
| D8 | 0.88 | 0.09 | 0.03 | 1 |
| D9 | 0.83 | 0.13 | 0.04 | 1 |
| D10 | 0.13 | 0.38 | 0.49 | 1 |

d. Menghitung nilai pusat *cluster*.

Perhitungan pusat *cluster* diawali dengan memangkatkan derajat keanggotaan yang telah ditemukan di tahap sebelumnya dengan nilai *w* (w= 2).

Tabel 3.13 Perhitungan Pangkat Derajat Keanggotaan

| Data | Derajat Keanggotaan | | |
|---|---|---|---|
| | $(\mu_{i1})^2$ | $(\mu_{i2})^2$ | $(\mu_{i3})^2$ |
| D1 | 0.01 | 0.13 | 0.27 |
| D2 | 0.00 | 0.01 | 0.80 |
| D3 | 0.30 | 0.09 | 0.03 |
| D4 | 0.06 | 0.34 | 0.03 |
| D5 | 0.01 | 0.70 | 0.01 |
| D6 | 0.69 | 0.01 | 0.00 |
| D7 | 0.00 | 0.01 | 0.72 |
| D8 | 0.77 | 0.01 | 0.00 |
| D9 | 0.69 | 0.02 | 0.00 |
| D10 | 0.02 | 0.14 | 0.24 |
| **Total** | **2.55** | **1.46** | **2.10** |

Langkah selanjutnya, mengalikan nilai data dengan nilai derajat keanggotaan yang telah dipangkatkan menggunakan persamaan (II.1). Perhitungan

perkalian nilai data dengan derajat keanggotaan pada masing-masing *cluster* dapat dilihat pada tabel berikut.

Tabel 3.14 Perkalian Data dengan Derajat Keanggotaan *Cluster* 1

| Data | $(\mu_{i1})^2 \cdot x_{ij}$ | | | | | | |
|---|---|---|---|---|---|---|---|
| | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
| D1 | 0.03 | 0.04 | 0.04 | 0.03 | 0.04 | 0.01 | 0.01 |
| D2 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| D3 | 1.09 | 0.96 | 1.09 | 1.01 | 1.07 | 0.59 | 0.30 |
| D4 | 0.17 | 0.24 | 0.16 | 0.19 | 0.19 | 0.12 | 0.06 |
| D5 | 0.02 | 0.03 | 0.02 | 0.03 | 0.02 | 0.01 | 0.01 |
| D6 | 2.49 | 2.73 | 2.61 | 2.75 | 2.60 | 0.69 | 0.69 |
| D7 | 0.01 | 0.00 | 0.01 | 0.01 | 0.01 | 0.00 | 0.00 |
| D8 | 3.06 | 2.63 | 2.31 | 2.91 | 2.89 | 0.77 | 0.77 |
| D9 | 2.49 | 2.77 | 1.93 | 2.62 | 2.76 | 0.69 | 0.69 |
| D10 | 0.05 | 0.05 | 0.05 | 0.06 | 0.04 | 0.02 | 0.03 |
| **Total** | **9.41** | **9.45** | **8.21** | **9.61** | **9.63** | **2.91** | **2.57** |

Tabel 3.15 Perkalian Data dengan Derajat Keanggotaan *Cluster* 2

| Data | $(\mu_{i2})^2 \cdot x_{ij}$ | | | | | | |
|---|---|---|---|---|---|---|---|
| | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
| D1 | 0.28 | 0.33 | 0.32 | 0.31 | 0.35 | 0.13 | 0.13 |
| D2 | 0.01 | 0.02 | 0.01 | 0.01 | 0.02 | 0.01 | 0.01 |
| D3 | 0.32 | 0.28 | 0.32 | 0.29 | 0.31 | 0.17 | 0.09 |
| D4 | 0.96 | 1.35 | 0.92 | 1.11 | 1.11 | 0.68 | 0.34 |
| D5 | 1.96 | 2.30 | 1.83 | 2.62 | 1.96 | 0.70 | 0.70 |
| D6 | 0.05 | 0.06 | 0.06 | 0.06 | 0.06 | 0.01 | 0.01 |
| D7 | 0.03 | 0.02 | 0.03 | 0.03 | 0.03 | 0.02 | 0.02 |
| D8 | 0.03 | 0.03 | 0.02 | 0.03 | 0.03 | 0.01 | 0.01 |
| D9 | 0.06 | 0.06 | 0.04 | 0.06 | 0.06 | 0.02 | 0.02 |
| D10 | 0.42 | 0.40 | 0.43 | 0.47 | 0.38 | 0.14 | 0.29 |
| **Total** | **4.12** | **4.85** | **3.98** | **5.00** | **4.31** | **1.90** | **1.62** |

Tabel 3.16 Perkalian Data dengan Derajat Keanggotaan *Cluster* 3

| Data | $(\mu_{i3})^2 \cdot x_{ij}$ | | | | | | |
|---|---|---|---|---|---|---|---|
| | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
| D1 | 0.58 | 0.69 | 0.65 | 0.63 | 0.72 | 0.27 | 0.27 |
| D2 | 1.91 | 2.16 | 2.07 | 2.11 | 2.26 | 1.61 | 1.61 |
| D3 | 0.09 | 0.08 | 0.09 | 0.09 | 0.09 | 0.05 | 0.03 |
| D4 | 0.08 | 0.12 | 0.08 | 0.09 | 0.09 | 0.06 | 0.03 |
| D5 | 0.01 | 0.02 | 0.01 | 0.02 | 0.01 | 0.01 | 0.01 |
| D6 | 0.01 | 0.01 | 0.01 | 0.01 | 0.01 | 0.00 | 0.00 |
| D7 | 2.02 | 1.44 | 1.94 | 2.16 | 2.02 | 1.44 | 1.44 |
| D8 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 | 0.00 |
| D9 | 0.01 | 0.01 | 0.00 | 0.01 | 0.01 | 0.00 | 0.00 |
| D10 | 0.69 | 0.66 | 0.72 | 0.77 | 0.62 | 0.24 | 0.48 |
| **Total** | **5.41** | **5.19** | **5.57** | **5.90** | **5.85** | **3.68** | **3.86** |

Selanjutnya, pusat *cluster* dihitung menggunakan elemen-elemen 'total' yang telah didapat dari langkah-langkah sebelumnya. Adapun perhitungan pusat *cluster* dilakukan dengan persamaan (II.2).

$$v_{kj} = \frac{\sum_{i=1}^{n} \left((\mu_{ik})^w \cdot x_{ij}\right)}{\sum_{i=1}^{n} (\mu_{ik})^w}$$

$v_{11} = \frac{9.41}{2.55} = 3.68$ $v_{12} = \frac{9.45}{2.55} = 3.70$ $v_{13} = \frac{8.21}{2.55} = 3.21$ $v_{14} = \frac{9.61}{2.55} = 3.76$ $v_{15} = \frac{9.63}{2.55} = 3.77$ $v_{16} = \frac{2.91}{2.55} = 1.14$ $v_{17} = \frac{2.57}{2.55} = 1.01$

$v_{21} = \frac{4.12}{1.46} = 2.83$ $v_{22} = \frac{4.85}{1.46} = 3.33$ $v_{23} = \frac{3.98}{1.46} = 2.73$ $v_{24} = \frac{5.00}{1.46} = 3.43$ $v_{25} = \frac{4.31}{1.46} = 2.96$ $v_{26} = \frac{1.90}{1.46} = 1.31$ $v_{27} = \frac{1.62}{1.46} = 1.11$

$v_{31} = \frac{5.41}{2.10} = 2.58$ $v_{32} = \frac{5.19}{2.10} = 2.47$ $v_{33} = \frac{5.57}{2.10} = 2.66$ $v_{34} = \frac{5.90}{2.10} = 2.81$ $v_{35} = \frac{5.85}{2.10} = 2.79$ $v_{36} = \frac{3.68}{2.10} = 1.75$ $v_{37} = \frac{3.86}{2.10} = 1.84$

Pusat *cluster* pada iterasi terakhir berdasarkan hasil perhitungan :

Tabel 3.17 Pusat Cluster

| Pusat *Cluster* ($v_{kj}$) pada Iterasi Ke-17 | | | | | | | |
|---|---|---|---|---|---|---|---|
| Pusat *Cluster* ( $v_{kj}$ ) | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
| Cluster 1 | 3.68 | 3.70 | 3.21 | 3.76 | 3.77 | 1.14 | 1.01 |
| Cluster 2 | 2.83 | 3.33 | 2.73 | 3.43 | 2.96 | 1.31 | 1.11 |
| Cluster 3 | 2.58 | 2.47 | 2.66 | 2.81 | 2.79 | 1.75 | 1.84 |

e. Perhitungan fungsi objektif.

Fungsi objektif digunakan untuk mengetahui apakah error yang dihasilkan lebih kecil dari yang diharapkan. Perhitungan fungsi objektif membutuhkan elemen nilai data, nilai pusat cluster, dan nilai perpangkatan dari derajat keanggotaan yang telah dihitung di tahap sebelumnya. Fungsi objektif dihitung menggunakan persamaan (II.3).

$$P_t = \sum_{i=1}^{n} \sum_{k=1}^{3} \left( \left[ \sum_{j=1}^{m} (x_{ij} - v_{kj})^2 \right] (\mu_{ik})^w \right)$$

Fungsi objektif iterasi ke-17 :

$$P_{17} = ( [(2.15 - 3.68)^2 ] 0.01 ) + ( [(2.56 - 3.70)^2 ] 0.00 ) +$$

$$( [(2.43 - 3.21)^2 ] 0.30 ) + \ldots + ( [(1 - 0.84)^2 ] 0.24 )$$

$$P_{17} = 4.365$$

Tabel 3.18 Perhitungan Fungsi Objektif pada Iterasi ke-17

| Data | $L_{i1}$ | $L_{i2}$ | $L_{i3}$ | Total |
|---|---|---|---|---|
| D1 | 0.11 | 0.33 | 0.47 | 0.90 |
| D2 | 0.01 | 0.01 | 0.18 | 0.20 |
| D3 | 0.39 | 0.21 | 0.12 | 0.72 |
| D4 | 0.14 | 0.34 | 0.10 | 0.58 |
| D5 | 0.02 | 0.17 | 0.01 | 0.21 |
| D6 | 0.32 | 0.05 | 0.02 | 0.38 |
| D7 | 0.02 | 0.04 | 0.29 | 0.34 |
| D8 | 0.18 | 0.02 | 0.01 | 0.20 |
| D9 | 0.24 | 0.04 | 0.01 | 0.29 |
| D10 | 0.07 | 0.21 | 0.27 | 0.54 |
| **Fungsi Objektif ($P_{17}$)** | | | | **4.36** |

f. Cek kondisi pemberhentian iterasi.

Iterasi proses perhitungan dihentikan jika selisih fungsi objektif kurang dari error terkecil ($|P_t - P| < \varepsilon$) dan atau jumlah iterasi lebih dari iterasi maksimum ($t >$ MaxIter). Jika kondisi belum terpenuhi, maka iterasi dilanjutkan.

$$|P_{17} - P_{16}| = 4.364835393 - 4.364842615$$

$$|P_{17} - P_{16}| = 0.000007 = 7^{-6}$$

Error terkecil yang diharapkan yaitu $\varepsilon = 10^{-5}$.

$$|P_{17} - P_{16}| < \varepsilon$$

Sehingga iterasi berhenti pada **iterasi ke-17** karena kondisi terpenuhi.

g. Menghitung matriks partisi.

Matriks partisi dihitung untuk mendapatkan kelompok derajat keanggotaan baru di iterasi berikutnya. Tetapi jika iterasi yang sedang berjalan adalah iterasi terakhir, maka matriks partisi menjadi penentu *cluster* data. Adapun matriks partisi dapat dihitung menggunakan persamaan (II.4).

Matriks partisi data D1 pada iterasi ke-17 :

$$L_{i1} = \left[\sum_{j=1}^{n} \left(x_{ij} - v_{kj}\right)^2\right]^{\frac{-1}{w-1}}$$

$$L_{i1} = [(2.15 - 3.68)^2 + (2.56 - 3.70)^2 + (2.43 - 3.21)^2 + (2.35 - 3.76)^2 + (2.69 - 3.77)^2 + (1 - 1.14)^2 + (1 - 1.01)^2]^{\frac{-1}{2-1}}$$

$$= 0.13$$

$$L_{i2} = [(2.15 - 2.83)^2 + (2.56 - 3.33)^2 + (2.43 - 2.73)^2 + (2.35 - 3.43)^2 + (2.69 - 2.96)^2 + (1 - 1.31)^2 + (1 - 1.11)^2]^{\frac{-1}{2-1}}$$

$$= 0.40$$

$$L_{i3} = [(2.15 - 2.58)^2 + (2.56 - 2.47)^2 + (2.43 - 2.66)^2 + (2.35 - 2.81)^2 + (2.69 - 2.79)^2 + (1 - 1.75)^2 + (1 - 1.84)^2]^{\frac{-1}{2-1}}$$

$$= 0.57$$

Perhitungan dilanjutkan hingga data D10 dengan langkah perhitungan yang sama. Tabel 3.19 merupakan hasil perhitungan matriks partisi pada iterasi ke-17.

Tabel 3.19 Hasil Perhitungan Matriks Partisi pada Iterasi ke-17

| Data | $L_{i1}$ | $L_{i2}$ | $L_{i3}$ |
|---|---|---|---|
| D1 | 0.13 | 0.40 | 0.57 |
| D2 | 0.14 | 0.38 | 4.57 |
| D3 | 0.76 | 0.41 | 0.22 |
| D4 | 0.42 | 1.00 | 0.29 |
| D5 | 0.45 | 4.03 | 0.34 |
| D6 | 2.19 | 0.32 | 0.12 |
| D7 | 0.14 | 0.31 | 2.50 |
| D8 | 4.31 | 0.44 | 0.16 |
| D9 | 2.86 | 0.43 | 0.14 |
| D10 | 0.24 | 0.70 | 0.90 |

h. Pengelompokkan data.

Data masuk ke anggota suatu *cluster* jika ada nilai derajat keanggotaan yang paling besar diantara lainnya (Nabila et al., 2021). Dengan demikian, berikut adalah klastering data berdasarkan hasil perhitungan yang dilakukan.

Tabel 3.20 Proses Klastering Data

| Data | $L_{i1}$ | $L_{i2}$ | $L_{i3}$ | Nilai Maksimum | Cluster |
|---|---|---|---|---|---|
| D1 | 0.13 | 0.40 | 0.57 | 0.57 | 3 |
| D2 | 0.14 | 0.38 | 4.57 | 4.57 | 3 |
| D3 | 0.76 | 0.41 | 0.22 | 0.76 | 1 |
| D4 | 0.42 | 1.00 | 0.29 | 1.00 | 2 |
| D5 | 0.45 | 4.03 | 0.34 | 4.03 | 2 |
| D6 | 2.19 | 0.32 | 0.12 | 2.19 | 1 |
| D7 | 0.14 | 0.31 | 2.50 | 2.50 | 3 |
| D8 | 4.31 | 0.44 | 0.16 | 4.31 | 1 |
| D9 | 2.86 | 0.43 | 0.14 | 2.86 | 1 |
| D10 | 0.24 | 0.70 | 0.90 | 0.90 | 3 |

Tabel 3.21 Hasil Klastering Data

| *Cluster* 1 | *Cluster* 2 | *Cluster* 3 |
|---|---|---|
| D3 | D4 | D1 |
| D6 | D5 | D2 |
| D8 | | D7 |
| D9 | | D10 |

Dari hasil tersebut diberikan contoh *knowledge* untuk masing-masing *cluster*. Klastering berdasarkan *range* indeks prestasi akademik sebagai berikut.

- Cluster 1 : kumpulan data mahasiswa dengan *range* IPS 3.51 - 3.63.
- Cluster 2 : kumpulan data mahasiswa dengan *range* IPS 3.06 - 3.19.
- Cluster 3 : kumpulan data mahasiswa dengan *range* IPS 2.44 - 2.90.

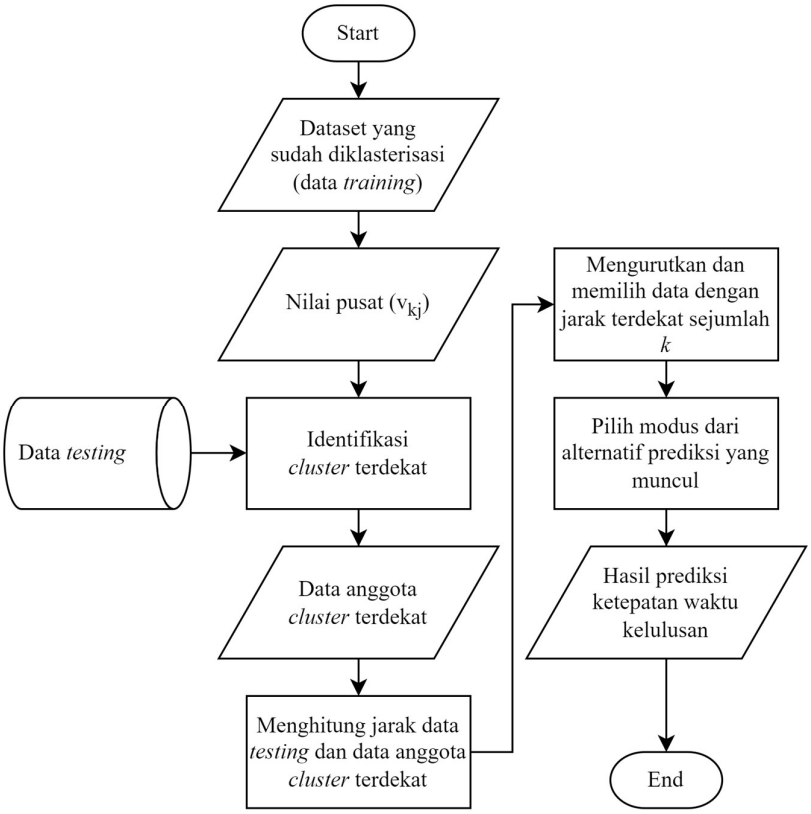
Berdasarkan contoh perhitungan algoritma *Fuzzy c-Means*, Tabel 3.22 berikut merupakan dataset baru yang terbentuk setelah melalui proses klastering.

Tabel 3.22 Dataset Baru

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | Cluster |
|---|---|---|---|---|---|---|---|---|
| D1 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | 1 | 1 | 3 |
| D2 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | 2 | 2 | 3 |
| D3 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | 2 | 1 | 1 |
| D4 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | 2 | 1 | 2 |
| D5 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | 1 | 1 | 2 |
| D6 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | 1 | 1 | 1 |
| D7 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | 2 | 2 | 3 |
| D8 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | 1 | 1 | 1 |
| D9 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | 1 | 1 | 1 |
| D10 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | 1 | 2 | 3 |

## 3.6 Algoritma k-Nearest Neighbors

Di bawah ini merupakan bagan proses algoritma *k-Nearest Neighbors* yang berjalan pada sistem.




Gambar 3.5 Bagan Metode *k-Nearest Neighbors*

Berikut sistem yang berjalan ketika prediksi data menggunakan *k-Nearest Neighbors* :

a. Input dataset dan nilai pusat *cluster*.

Dataset yang diinputkan didapat dari output algoritma *fuzzy c-means* yang telah diberi label. Data tersebut dijadikan data *training* untuk model *k-Nearest Neighbors*. Nilai pusat *cluster* ditunjukkan oleh tabel 3.17. Sedangkan dataset ditunjukkan dalam tabel 3.23 di bawah ini.

Tabel 3.23 Data Training

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | Cluster | Label |
|---|---|---|---|---|---|---|---|---|---|
| D1 | 2.15 | 2.56 | 2.43 | 2.35 | 2.69 | 1 | 1 | 3 | Lulus Tepat Waktu |
| D2 | 2.38 | 2.69 | 2.57 | 2.62 | 2.81 | 2 | 2 | 3 | Lulus Tidak Tepat Waktu |
| D3 | 3.65 | 3.23 | 3.65 | 3.39 | 3.61 | 2 | 1 | 1 | Lulus Tidak Tepat Waktu |
| D4 | 2.81 | 3.95 | 2.69 | 3.25 | 3.25 | 2 | 1 | 2 | Lulus Tidak Tepat Waktu |
| D5 | 2.81 | 3.30 | 2.62 | 3.76 | 2.81 | 1 | 1 | 2 | Lulus Tepat Waktu |
| D6 | 3.60 | 3.95 | 3.78 | 3.98 | 3.76 | 1 | 1 | 1 | Lulus Tepat Waktu |
| D7 | 2.81 | 2.00 | 2.69 | 3.00 | 2.81 | 2 | 2 | 3 | Lulus Tepat Waktu |
| D8 | 3.98 | 3.42 | 3.00 | 3.78 | 3.76 | 1 | 1 | 1 | Lulus Tepat Waktu |
| D9 | 3.60 | 4.00 | 2.78 | 3.78 | 3.98 | 1 | 1 | 1 | Lulus Tepat Waktu |
| D10 | 2.89 | 2.77 | 3.00 | 3.25 | 2.61 | 1 | 2 | 3 | Lulus Tidak Tepat Waktu |

b. Identifikasi *cluster* terdekat

Pencarian *cluster* terdekat dilakukan dengan menghitung jarak euclidean data *testing* dengan masing-masing nilai pusat *cluster*. Jarak euclidean bisa dihitung menggunakan persamaan (II.4).

Tabel 3.24 Data Testing

| Data | NIM | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | Label |
|---|---|---|---|---|---|---|---|---|---|
| D11 | 19650020 | 3.60 | 3.95 | 3.00 | 3.98 | 3.47 | 1 | 1 | ? |
| D12 | 19650031 | 3.86 | 2.20 | 3.89 | 2.81 | 3.00 | 2 | 1 | ? |
| D13 | 19650045 | 2.80 | 2.98 | 3.60 | 2.76 | 3.70 | 1 | 2 | ? |

Perhitungan jarak data D11 dengan pusat *cluster* 1, 2, dan 3 :

$$d_i = \sqrt[d_i]{\sum_{i=1}^{n} (x_i - y_i)^2}$$

$$d_1 = \sqrt{(3.6 - 3.68)^2 + (3.95 - 3.7)^2 + (3 - 3.21)^2 + \dots}$$

$$\sqrt{(3.98 - 3.76)^2 + (3.47 - 3.77)^2 + (1 - 1.14)^2 + (1 - 1.01)^2}$$

$$= 0.52$$

$$d_2 = \sqrt{(3.6 - 2.83)^2 + (3.95 - 3.33)^2 + (3 - 2.73)^2 + \dots}$$

$$\sqrt{(3.98 - 3.43)^2 + (3.47 - 2.96)^2 + (1 - 1.31)^2 + (1 - 1.11)^2}$$

$$= 1.31$$

$d_3 = \sqrt{(3.6 - 2.58)^2 + (3.95 - 2.47)^2 + (3 - 2.66)^2 + \ldots}$

$\sqrt{(3.98 - 2.81)^2 + (3.47 - 2.79)^2 + (1 - 1.75)^2 + (1 - 1.84)^2}$

$= 2.54$

Perhitungan jarak data D12 dengan pusat *cluster* 1, 2, dan 3 :

$d_i = \sqrt[d_i]{\sum_{i=1}^{n} (x_i - y_i)^2}$

$d_1 = \sqrt{(3.86 - 3.68)^2 + (2.2 - 3.7)^2 + (3.89 - 3.21)^2 + \ldots}$

$\sqrt{(2.81 - 3.76)^2 + (3 - 3.77)^2 + (2 - 1.14)^2 + (1 - 1.01)^2}$

$= 2.23$

$d_2 = \sqrt{(3.86 - 2.83)^2 + (2.2 - 3.33)^2 + (3.89 - 2.73)^2 + \ldots}$

$\sqrt{(2.81 - 3.43)^2 + (3 - 2.96)^2 + (2 - 1.31)^2 + (1 - 1.11)^2}$

$= 2.13$

$d_3 = \sqrt{(3.86 - 2.58)^2 + (2.2 - 2.47)^2 + (3.89 - 2.66)^2 + \ldots}$

$\sqrt{(2.81 - 2.81)^2 + (3 - 2.79)^2 + (2 - 1.75)^2 + (1 - 1.84)^2}$

$= 2.01$

Perhitungan jarak data D13 dengan pusat *cluster* 1, 2, dan 3 :

$d_i = \sqrt[d_i]{\sum_{i=1}^{n} (x_i - y_i)^2}$

$d_1 = \sqrt{(2.8 - 3.68)^2 + (2.98 - 3.7)^2 + (3.6 - 3.21)^2 + \ldots}$

$\sqrt{(2.76 - 3.76)^2 + (3.7 - 3.77)^2 + (1 - 1.14)^2 + (2 - 1.01)^2}$

$= 1.86$

$d_2 = \sqrt{(2.8 - 2.83)^2 + (2.98 - 3.33)^2 + (3.6 - 2.73)^2 + \ldots}$

$$\sqrt{(2.76-3.43)^2+(3.7-2.96)^2+(1-1.31)^2+(2-1.11)^2}$$

$$= 1.66$$

$$d_3 = \sqrt{(2.8-2.58)^2+(2.98-2.47)^2+(3.6-2.66)^2+\dots}$$

$$\sqrt{(2.76-2.81)^2+(3.7-2.79)^2+(1-1.75)^2+(2-1.84)^2}$$

$$= 1.62$$

Setelah menghitung jarak data *testing* ke pusat masing-masing *cluster*, maka didapati hasil pada tabel 3.25 berikut.

Tabel 3.25 Pemilihan Cluster Terdekat

| Data Testing | *Cluster* Terdekat |
|---|---|
| D11 | *Cluster* 1 |
| D12 | *Cluster* 3 |
| D13 | *Cluster* 3 |

Dari hasil dapat dilihat bahwa data D11 paling dekat dengan *cluster* 1 sehingga D11 akan diuji dengan anggota data *training* dalam *cluster* 1. Data D12 paling dekat dengan *cluster* 2 sehingga D12 akan diuji dengan anggota data *training* dalam *cluster* 2. Data D13 paling dekat dengan *cluster* 3 sehingga D13 akan diuji dengan anggota data *training* dalam *cluster* 3.

c. Menghitung jarak data *testing* dengan data tetangga dalam *cluster* terpilih.

Perhitungan jarak data D11 dengan data anggota *cluster* 1 :

$$d_{(D11\ ke\ D3)} = \sqrt{(3.6-3.65)^2+(3.95-3.23)^2+(3-3.65)^2+}$$

$$\sqrt{+(3.98-3.39)^2+(3.47-3.61)^2+(1-2)^2+}$$

$$\sqrt{+(1-1)^2}$$

$$= 1.52$$

$d_{(D1\ \ ke\ D6)}$ = $\sqrt{(3.6-3.6)^2 + (3.95-3.95)^2 + (3-3.78)^2 +}$

$\sqrt{+(3.98-3.98)^2 + (3.47-3.76)^2 + (1-1)^2 +}$

$\sqrt{+(1-1)^2}$

= 0.83

$d_{(D11\ ke\ D8)}$ = $\sqrt{(3.6-3.98)^2 + (3.95-3.42)^2 + (3-3)^2 +}$

$\sqrt{+(3.98-3.78)^2 + (3.47-3.76)^2 + (1-1)^2 +}$

$\sqrt{+(1-1)^2}$

= 0.74

$d_{(D1\ \ ke\ D9)}$ = $\sqrt{(3.6-3.6)^2 + (3.95-4)^2 + (3-2.78)^2 +}$

$\sqrt{+(3.98-3.78)^2 + (3.47-3.98)^2 + (1-1)^2 +}$

$\sqrt{+(1-1)^2}$

= 0.59

Perhitungan jarak data D12 dengan data anggota *cluster* 3 :

$d_{(D12\ ke\ D1)}$ = $\sqrt{(3.86-2.15)^2 + (2.2-2.56)^2 + (3.89-2.43)^2 +}$

$\sqrt{+(2.81-2.35)^2 + (3-2.69)^2 + (2-1)^2 +}$

$\sqrt{+(1-1)^2}$

= 2.55

$d_{(D12\ ke\ D2)}$ = $\sqrt{(3.86-2.38)^2 + (2.2-2.69)^2 + (3.89-2.57)^2 +}$

$\sqrt{+(2.81-2.62)^2 + (3-2.81)^2 + (2-2)^2 +}$

$\sqrt{+(1-2)^2}$

$= 2.29$

$d_{(D1\ \ ke\ D7)} = \sqrt{(3.86-2.81)^2 + (2.2-2)^2 + (3.89-2.69)^2 + \sqrt{+(2.81-3)^2 + (3-2.81)^2 + (2-2)^2 + \sqrt{+(1-2)^2}}}$

$= 1.91$

$d_{(D12\ ke\ D1\ )} = \sqrt{(3.86-2.89)^2 + (2.2-2.77)^2 + (3.89-3)^2 + \sqrt{+(2.81-3.25)^2 + (3-2.61)^2 + (2-1)^2 + \sqrt{+(1-2)^2}}}$

$= 2.10$

Perhitungan jarak data D13 dengan data anggota *cluster* 3 :

$d_{(D13\ ke\ D1)} = \sqrt{(2.8-2.15)^2 + (2.98-2.56)^2 + (3.6-2.43)^2 + \sqrt{+(2.76-2.35)^2 + (3.7-2.69)^2 + (1-1)^2 + \sqrt{+(2-1)^2}}}$

$= 2.04$

$d_{(D1\ \ ke\ D2)} = \sqrt{(2.8-2.38)^2 + (2.98-2.69)^2 + (3.6-2.57)^2 + \sqrt{+(2.76-2.62)^2 + (3.7-2.81)^2 + (1-2)^2 + \sqrt{+(2-2)^2}}}$

$= 1.77$

$d_{(D1\ \ ke\ D7)} = \sqrt{(2.8-2.81)^2 + (2.98-2)^2 + (3.6-2.69)^2 + \sqrt{+(2.76-3)^2 + (3.7-2.81)^2 + (1-2)^2 +}}$

$$\sqrt{+(2-2)^2}$$

$$= 1.91$$

$$d_{(D1 \ \ ke\ D10)} = \sqrt{(2.8-2.89)^2 + (2.98-2.77)^2 + (3.6-3)^2 +}$$

$$\sqrt{+(2.76-3.25)^2 + (3.7-2.61)^2 + (1-1)^2 +}$$

$$\sqrt{+(2-2)^2}$$

$$= 1.36$$

d. Menetapkan nilai *k* dan prediksi data *testing*.

Perhitungan nilai k-tetangga terdekat hanya dilakukan terhadap anggota cluster terpilih sebab telah dilakukan proses klastering sebelumnya. Hal ini akan mempercepat proses komputasi karena nilai k tidak perlu dibandingkan dengan seluruh data dalam dataset (Nabila et al., 2021). Pemilihan kelas data menggunakan konsep modus pada sejumlah *k* data.

Data *testing* diuji dengan data anggota *cluster* terpilih (lihat Tabel 3.25). Adapun pengujian data *testing* ditunjukkan oleh Tabel 3.26 berikut.

Tabel 3.26 Pengujian Data Testing

| Data Testing | *Cluster* Terdekat | Data Tetangga | Label | Jarak | $k$ =1 | $k$ = 3 |
|---|---|---|---|---|---|---|
| D11 | *Cluster* 1 | D3 | Lulus Tidak Tepat Waktu | 1.52 | | |
| | | D6 | Lulus Tepat Waktu | 0.83 | | ✓ |
| | | D8 | Lulus Tepat Waktu | 0.74 | | ✓ |
| | | D9 | Lulus Tepat Waktu | 0.59 | ✓ | ✓ |
| D12 | *Cluster* 3 | D1 | Lulus Tepat Waktu | 2.55 | | |
| | | D2 | Lulus Tidak Tepat Waktu | 2.29 | | ✓ |
| | | D7 | Lulus Tepat Waktu | 1.91 | ✓ | ✓ |
| | | D10 | Lulus Tidak Tepat Waktu | 2.10 | | ✓ |
| D13 | *Cluster* 3 | D1 | Lulus Tepat Waktu | 2.04 | | |
| | | D2 | Lulus Tidak Tepat Waktu | 1.77 | | ✓ |
| | | D7 | Lulus Tepat Waktu | 1.91 | | ✓ |
| | | D10 | Lulus Tidak Tepat Waktu | 1.36 | ✓ | ✓ |

Berdasarkan contoh perhitungan, hasil prediksi ketepatan waktu mahasiswa yang diwakili oleh data D11, D12, dan D13 adalah sebagai berikut.

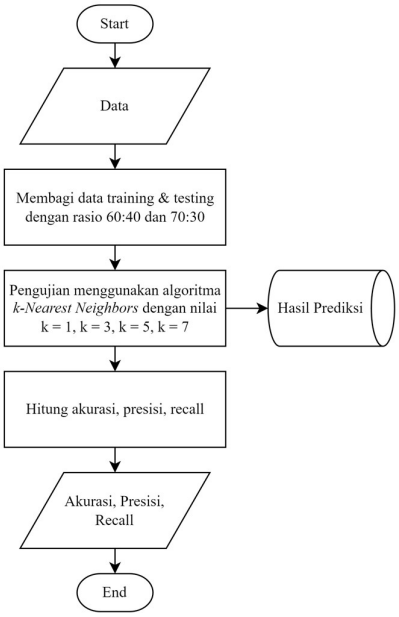
Tabel 3.27 Contoh Hasil Prediksi

| Data | Prediksi ($k$ =1) | Prediksi ($k$ =3) |
|---|---|---|
| D11 | Lulus Tepat Waktu | Lulus Tepat Waktu |
| D12 | Lulus Tepat Waktu | Lulus Tidak Tepat Waktu |
| D13 | Lulus Tidak Tepat Waktu | Lulus Tidak Tepat Waktu |

## 3.7 Skenario Uji Coba

Pengujian dilakukan dengan beberapa perlakuan, bertujuan untuk mencari skenario dan model prediksi yang paling optimal (Han et al., 2022). Skenario uji coba dilakukan dalam dua bagian utama yaitu pengujian dengan algoritma *k-Nearest Neighbors* tanpa penambahan algoritma *fuzzy c-means*, dan pengujian dengan penambahan algoritma *fuzzy c-means*.

Bagan skenario uji coba tanpa penambahan algoritma *fuzzy c-means* ditunjukkan oleh gambar 3.6 di bawah ini.

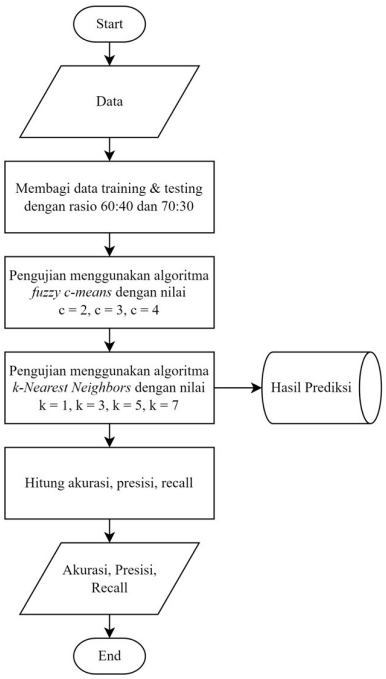



Gambar 3.6 Bagan Skenario Uji Coba Tanpa Optimasi menggunakan *Fuzzy c-Means*

Skenario uji coba diawali dengan membagi dataset menjadi data *training* dan data *testing* dengan rasio 60:40 dan 70:30. Pada uji coba pertama, dibuat model prediksi dengan mengimplementasikan metode *k-Nearest Neighbors* tanpa penambahan algoritma *Fuzzy c-Means*. Masing-masing model prediksi yang dibuat, dihitung nilai accuracy, precision, dan recall-nya guna menemukan model prediksi yang terbaik. Skenario kustomisasi nilai parameter nilai *k* pada implementasi algoritma *k-Nearest Neighbors* ditunjukkan oleh Tabel 3.28 di bawah ini.

Tabel 3.28 Skenario Uji Coba Tanpa Optimasi menggunakan Algoritma *Fuzzy c-Means*

| Model Prediksi | Skenario Nilai *k* (*k-Nearest Neighbors*) |
|---|---|
| Model 1 | $k = 1$ |
| Model 2 | $k = 3$ |
| Model 3 | $k = 5$ |
| Model 4 | $k = 7$ |




Gambar 3.7 Bagan Skenario Uji Coba dengan Optimasi menggunakan *Fuzzy c-Means*

Seperti yang ditunjukkan Gambar 3.7 pada uji coba kedua, dibuat model prediksi dengan mengimplementasikan metode *k-Nearest Neighbors* dengan penambahan proses klastering data menggunakan algoritma *Fuzzy c-Means*. Masing-masing model prediksi yang dibuat, dihitung nilai *accuracy*, *precision*, dan *recall*-nya guna menemukan model prediksi yang terbaik.

Berikut skenario inisialisasi parameter pada tahap pengimplementasian algoritma *Fuzzy c-Means*.

Tabel 3.29 Parameter untuk Algoritma Fuzzy c-Means

| *Cluster* *(c)* | Derajat Pembobot *(w)* | Iterasi *(MaxIter)* | Error *(ε)* | Fungsi Objektif *( $P_0$ )* | Iterasi Awal *(t)* |
|---|---|---|---|---|---|
| *C = 2*<br>*C = 3*<br>*C = 4* | 2 | 1000 | $10^{-5}$ | 0 | 1 |

Skenario kustomisasi nilai parameter jumlah *cluster* (*c*) dan nilai parameter nilai *k* pada implementasi algoritma *Fuzzy c-Means* dan *k-Nearest Neighbors* ditunjukkan oleh Tabel 3.30 di bawah ini.

Tabel 3.30 Skenario Uji Coba dengan Optimasi menggunakan Algoritma *Fuzzy c-Means*

| Model Prediksi | Skenario Jumlah *Cluster* (*Fuzzy c-Means*) | Skenario Nilai *k* (*k-Nearest Neighbors*) |
|---|---|---|
| Model 5 | *Cluster* = 2 | *k* = 1 |
| Model 6 | | *k* = 3 |
| Model 7 | | *k* = 5 |
| Model 8 | | *k* = 7 |
| Model 9 | Cluster = 3 | *k* = 1 |
| Model 10 | | *k* = 3 |
| Model 11 | | *k* = 5 |
| Model 12 | | *k* = 7 |
| Model 13 | Cluster = 4 | *k* = 1 |
| Model 14 | | *k* = 3 |
| Model 15 | | *k* = 5 |
| Model 16 | | *k* = 7 |

Setiap model prediksi akan diuji ketepatan performa prediksinya menggunakan teknik *confusion matrix*. Adapun konsep *confusion matrix* adalah sebagai berikut.

Tabel 3.31 Konsep Confusion Matrix

| Aktual | Hasil Prediksi | |
|---|---|---|
| | *Positive* | *Negative* |
| *Positive* | *True Positive* (TP) | *False Positive* (FP) |
| *Negative* | *False Negative* (FN) | *True Negative* (TN) |

Berdasarkan Tabel 3.31 perhitungan kinerja algoritma dapat dilakukan dengan mengacu pada persamaan (III, 1) - (III, 3) berikut ini.

$$\text{Akurasi} = \frac{TP + TN}{TP + FP + TN + FN} \times 100\% \quad \text{(III, 1)}$$

$$\text{Presisi} = \frac{TP}{TP + FP} \times 100\% \quad \text{(III, 2)}$$

$$\textit{Recall} = \frac{TP}{TP + FN} \times 100\% \quad \text{(III, 3)}$$

Dimana,

TP : Jumlah data yang memiliki nilai positif (tepat waktu) dan diprediksi sebagai data positif (tepat waktu)

FP : Jumlah data yang memiliki nilai negatif (telat) tetapi diprediksi sebagai data positif (tepat waktu)

FN : Jumlah data yang memiliki nilai positif (tepat waktu) tetapi diprediksi sebagai data negatif (telat)

TN : Jumlah data yang memiliki nilai negatif (telat) dan diprediksi sebagai data negatif (telat).

# BAB IV

# UJI COBA DAN PEMBAHASAN

Pada bagian uji coba dan pembahasan akan dijelaskan mengenai implementasi dan pengujian sistem yang telah dibangun. Bab ini terdiri atas beberapa sub bab yang menjelaskan tentang hasil implementasi sistem, hasil uji coba, serta pembahasan.

## 4.1 Hasil Data Preprocessing

*Data preprocessing* pada penelitian ini dilakukan dalam 4 tahap yaitu *data cleaning, data integration, data transformation,* dan *data reduction*. Proses tersebut diimplementasikan menggunakan bahasa pemrograman *python* dengan bantuan *Google Colaboratory*. Data yang masuk kedalam proses *preprocessing* berjumlah 631 record dan berasal dari dua file berekstensi .csv.

*a. Data Integration*

Tahap pertama adalah *data integration*, yaitu mengintegrasikan data yang berasal dari beberapa sumber berbeda menjadi satu. Data yang diintegrasikan yaitu data akademik mahasiswa yang didapat dari BAK UIN Malang dengan atribut No, Angkatan, IPS1-IP5, Jenis Kelamin, Jenis Pembiayaan. Serta data tahun lulus mahasiswa yang didapat dari Pddikti.go.id dengan *primary key* yang sama yaitu ‘no’ yang mewakili ‘nim’.

Adapun hasil dari tahapan *data integration* ditunjukkan oleh Gambar 4.1 berikut ini.

| | No | Angkatan | IPS1 | IPS2 | IPS3 | IPS4 | IPS5 | Jenis Kelamin | Jenis Pembiayaan | Tahun Lulus |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2014 | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | P | Nonbidikmisi | 2018 |
| 1 | 2 | 2014 | 3.71 | 3.43 | 3.73 | 3.60 | 3.65 | L | Nonbidikmisi | 2019 |
| 2 | 3 | 2014 | 3.48 | 3.38 | 3.08 | - | - | P | Nonbidikmisi | - |
| 3 | 4 | 2014 | 3.10 | 3.07 | 2.83 | 2.95 | 2.75 | L | Nonbidikmisi | - |
| 4 | 5 | 2014 | 3.71 | 3.60 | 3.43 | 3.55 | 3.10 | P | Nonbidikmisi | 2019 |

Gambar 4.1 Hasil *Data Integration*

*b. Data Cleaning*

Tahap kedua adalah *data cleaning*, yaitu pembersihan record data yang bersifat tidak terstruktur, tidak lengkap, tidak valid, tidak konsisten, dan mengandung error atau outliers. *Data cleaning* dilakukan dengan menghapus record data yang bernilai null dan record data yang tidak memiliki nilai tahun lulus (mahasiswa yang dinyatakan tidak atau belum lulus). Contohnya yaitu menghapus record data dengan No 3 dan 4.

Adapun hasil dari tahapan *data cleaning* ditunjukkan oleh Gambar 4.2 di bawah ini.

| | No | Angkatan | IPS1 | IPS2 | IPS3 | IPS4 | IPS5 | Jenis Kelamin | Jenis Pembiayaan | Tahun Lulus |
|---|---|---|---|---|---|---|---|---|---|---|
| 0 | 1 | 2014 | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | P | Nonbidikmisi | 2018 |
| 1 | 2 | 2014 | 3.71 | 3.43 | 3.73 | 3.60 | 3.65 | L | Nonbidikmisi | 2019 |
| 4 | 5 | 2014 | 3.71 | 3.60 | 3.43 | 3.55 | 3.10 | P | Nonbidikmisi | 2019 |
| 5 | 6 | 2014 | 3.33 | 3.42 | 3.00 | 3.27 | 3.08 | L | Nonbidikmisi | 2020 |
| 6 | 7 | 2014 | 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | P | Nonbidikmisi | 2019 |

Gambar 4.2 Hasil *Data Cleaning*

*c. Data Transformation*

Tahap ketiga adalah *data transformation*, bertujuan untuk menyeragamkan format, skala, atau unit data sesuai kebutuhan penelitian. Tahap ini dilakukan

dengan melakukan transformasi data kategorikal menjadi data numerik dengan pengkodean seperti pada Tabel 4.1 berikut.

Tabel 4.1 Label Encoding

| Atribut | Kode |
|---|---|
| Jenis Kelamin | 1) Laki-laki = 1<br>2) Perempuan = 2 |
| Jenis Pembiayaan | 1) Bidikmisi = 1<br>2) Non Bidikmisi = 2 |

Selain itu juga dilakukan transformasi properti atribut data menjadi format yang lebih komputasional, ditunjukkan oleh Tabel 4.2.

Tabel 4.2 Transformasi Properti Atribut Data

| Atribut Lama | Atribut Baru |
|---|---|
| IPS 1 | ip_1 |
| IPS 2 | ip_2 |
| IPS 3 | ip_3 |
| IPS 4 | ip_4 |
| IPS 5 | ip_5 |
| Jenis Kelamin | jk |
| Jenis Pembiayaan | jp |

Pada tahap ini juga dilakukan penambahan kolom ‘label’ untuk menyimpan status ketepatan waktu kelulusan mahasiswa. Label berisi properti “Lulus Tepat Waktu” dan “Lulus Tidak Tepat Waktu”. Pengisian label didasarkan pada operasi pengurangan variabel **tahun_lulus** yang mewakili tahun lulus mahasiswa dan variabel **angkatan** yang mewakili tahun masuk mahasiswa. Jika hasil operasi kurang dari atau sama dengan 4 tahun ($\leq 4\ tahun$), maka dilabeli “Lulus Tepat Waktu”. Jika hasil operasi lebih dari 4 tahun ($> 4\ tahun$), dilabeli “Lulus Tidak Tepat Waktu”.

Adapun hasil dari tahap *data transformation* ditunjukkan oleh Gambar 4.3 di bawah ini.

| | no | angkatan | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | tahun_lulus | label |
|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 1 | 2014 | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | 2 | 2 | 2018 | Lulus Tepat Waktu |
| **1** | 2 | 2014 | 3.71 | 3.43 | 3.73 | 3.60 | 3.65 | 1 | 2 | 2019 | Lulus Tidak Tepat Waktu |
| **4** | 5 | 2014 | 3.71 | 3.60 | 3.43 | 3.55 | 3.10 | 2 | 2 | 2019 | Lulus Tidak Tepat Waktu |
| **5** | 6 | 2014 | 3.33 | 3.42 | 3.00 | 3.27 | 3.08 | 1 | 2 | 2020 | Lulus Tidak Tepat Waktu |
| **6** | 7 | 2014 | 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | 2 | 2 | 2019 | Lulus Tidak Tepat Waktu |

Gambar 4.3 Hasil *Data Transformation*

*d. Data Reduction*

Tahap terakhir adalah *data reduction*, yaitu reduksi atau pengurangan record data dengan kriteria tertentu, misalnya data yang mengandung duplikasi. Pada penelitian ini, *data reduction* dilakukan dengan menghapuskan data yang mengandung duplikasi dan menghapuskan kolom yang tidak diperlukan dalam perhitungan algoritma selanjutnya, yaitu kolom **tahun_lulus**, **angkatan**, dan **no**.

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 395 entries, 0 to 630
Data columns (total 8 columns):
 #   Column  Non-Null Count  Dtype
---  ------  --------------  -----
 0   ip_1    395 non-null    float64
 1   ip_2    395 non-null    float64
 2   ip_3    395 non-null    float64
 3   ip_4    395 non-null    float64
 4   ip_5    395 non-null    float64
 5   jk      395 non-null    int64
 6   jp      395 non-null    int64
 7   label   395 non-null    object
dtypes: float64(5), int64(2), object(1)
memory usage: 27.8+ KB
```

Gambar 4.4 Hasil *Data Reduction*

Seperti yang ditunjukkan oleh Gambar 4.4 di atas bahwa setelah melalui tahap *data reduction*, menyisakan data dengan format yang seragam, bebas dari bias

dan outliers sejumlah 395 record dengan 8 atribut. Tampilan output dataset setelah melalui tahapan *preprocessing*, ditunjukkan oleh Gambar 4.5 di bawah.

| | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | label |
|---|---|---|---|---|---|---|---|---|
| **0** | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | 2 | 2 | Lulus Tepat Waktu |
| **1** | 3.71 | 3.43 | 3.73 | 3.60 | 3.65 | 1 | 2 | Lulus Tidak Tepat Waktu |
| **4** | 3.71 | 3.60 | 3.43 | 3.55 | 3.10 | 2 | 2 | Lulus Tidak Tepat Waktu |
| **5** | 3.33 | 3.42 | 3.00 | 3.27 | 3.08 | 1 | 2 | Lulus Tidak Tepat Waktu |
| **6** | 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | 2 | 2 | Lulus Tidak Tepat Waktu |

Gambar 4.5 Hasil Akhir *Data Preprocessing*

## 4.2 Penerapan Algoritma *Fuzzy C-Means*

Setelah melalui tahap *preprocessing*, data kemudian dikombinasikan menjadi data *training* dan data *testing* dengan rasio 60:40. Data yang akan menjadi data *training*, diinputkan ke dalam algoritma *Fuzzy c-Means* untuk dilakukan klastering. Klastering berfungsi untuk mengelompokkan data berdasarkan karakteristik sejenis. Pengelompokkan data dimaksudkan untuk memperpendek proses pada algoritma selanjutnya serta meningkatkan akurasi.

Berikut adalah skenario inisialisasi parameter yang dibutuhkan pada implementasi algoritma *Fuzzy c-Means*.

Tabel 4.3 Skenario Parameter Algoritma *Fuzzy c-Means*

| *Cluster* *(c)* | Derajat Pembobot *(w)* | Iterasi *(MaxIter)* | Error *(ε)* | Fungsi Objektif *( $P_0$ )* | Iterasi Awal *(t)* |
|---|---|---|---|---|---|
| 3 | 2 | 1000 | $10^{-5}$ | 0 | 1 |

Proses klastering data diimplementasikan menggunakan bahasa pemrograman *python* dengan tools *Google Collaboratory*. Berikut merupakan

implementasi program dari proses klastering data menggunakan algoritma *Fuzzy c-Means* yang digambarkan dengan *pseudocode*.

```
   Program
      klasterisasi_data_dengan_fcm
   Deskripsi
      Library : numpy, pandas, fcmeans
      X_train,y_train,X_test, y_test :
array[]
   Implementasi
      #Input data dan Parameter
       Def FCM(X_train):
       fcm ← jml_cluster
       fcm.fit(X_train)
       centers ← fcm.centers
       labels ← fcm.u.argmax(axis=1)
       Return [centers, labels]
      #Mencari centroid dan label
cluster
       centers, labels = get
FCM(X_train)
      #Identifikasi cluster terdekat
       for j in range(len(X_test)):
           jarak_min ← 100000
           indeks_min ← -1
           for in range(jml_cluster):
                jarak ←
distance(X_test[j], centers[i])
                If jarak < jarak_min:
                     jarak_min ← jarak
                     Indeks_min ← i
      #Menyimpan cluster terdekat

ClusterTerdekat.append(indeks_min)
```

Gambar 4.6 Pseudocode Algoritma Fuzzy c-Means

Dapat dilihat dari Gambar 4.6 di atas bahwa proses klastering data dilakukan dengan urutan tahap yaitu input data dan parameter, mencari centroid, mencari label *cluster*, serta identifikasi *cluster* yang terdekat dengan data *testing*.

Tabel 4.4 Label Cluster

| *Cluster* ke- | Label *Cluster* |
|---|---|
| 1 | 0 |
| 2 | 1 |
| 3 | 2 |

Tabel 4.5 Pusat Cluster

| Pusat Cluster ( $v_{kj}$ ) | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp |
|---|---|---|---|---|---|---|---|
| Cluster 0 | 3.52 | 3.40 | 3.55 | 3.59 | 3.50 | 1.05 | 1.98 |
| Cluster 1 | 3.45 | 3.34 | 3.48 | 3.51 | 3.37 | 1.94 | 1.98 |
| Cluster 2 | 3.36 | 3.06 | 3.09 | 3.15 | 2.90 | 1.16 | 1.97 |

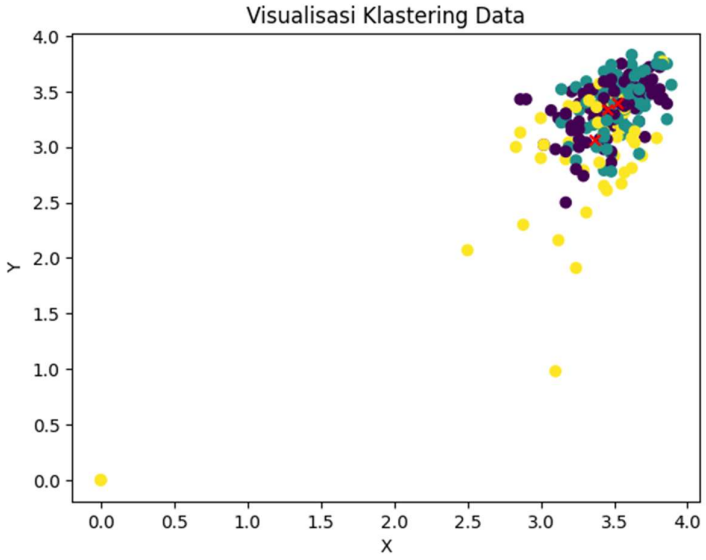
Seperti yang ditunjukkan oleh Tabel 4.4, maka setiap record data dalam dataset yang diinputkan ke dalam algoritma *Fuzzy c-Means* akan memiliki properti baru, yaitu label *cluster.* Output dataset yang dihasilkan algoritma *Fuzzy c-Means* dapat dilihat pada Gambar 4.7 di bawah ini.

| ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | label | Cluster |
|---|---|---|---|---|---|---|---|---|
| 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | 2 | 2 | Lulus Tepat Waktu | 2 |
| 3.71 | 3.43 | 3.73 | 3.60 | 3.65 | 1 | 2 | Lulus Tidak Tepat Waktu | 0 |
| 3.71 | 3.60 | 3.43 | 3.55 | 3.10 | 2 | 2 | Lulus Tidak Tepat Waktu | 2 |
| 3.33 | 3.42 | 3.00 | 3.27 | 3.08 | 1 | 2 | Lulus Tidak Tepat Waktu | 1 |
| 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | 2 | 2 | Lulus Tidak Tepat Waktu | 2 |

Gambar 4.7 Dataset Baru

Ditunjukkan bahwa data dengan indeks 1 masuk ke dalam kelompok *cluster* 0, data dengan indeks 3 masuk ke dalam *cluster* 1, data dengan indeks 0, 2, 4 masuk ke dalam *cluster* 2. Begitu juga seluruh data yang ada di dalam dataset.

Di bawah ini merupakan visualisasi hasil klastering data menggunakan algoritma *Fuzzy c-Means*.




Gambar 4.8 Visualisasi Hasil Klastering Data

Pada Gambar 4.8, warna ungu menunjukkan data point anggota *cluster* 0, warna hijau menunjukkan data point anggota *cluster* 1, dan warna kuning untuk *cluster* 2. Simbol 'x' menunjukkan letak pusat masing-masing *cluster*. Dapat dilihat dari gambar tersebut bahwa data berhasil dikelompokkan menjadi 3 *cluster* berdasarkan jarak terdekatnya dengan pusat *cluster* berdasarkan perhitungan algoritma *fuzzy c-means* di dalam sistem.

Dataset yang sudah dikelompokkan ini akan menjadi data *training* untuk algoritma *k-Nearest Neighbors*.

## 4.3 Penerapan Algoritma *k-Nearest Neighbors*

Metode *k-Nearest Neighbors* membutuhkan dua jenis data input yaitu data *training* dan data *testing*. Data *training* didapatkan dari dataset yang sudah melalui tahapan klastering menggunakan algoritma *Fuzzy c-Means,* yaitu berjumlah 60%

dari seluruh total input. Sedangkan data *testing* berjumlah 40% dari seluruh total input.

Adapun Gambar 4.11 berikut adalah proses implementasi *k-Nearest Neighbors* untuk prediksi dengan percobaan nilai *k=3* yang digambarkan menggunakan *pseudocode*.

```
   Program
     prediksi_data_dengan_knn
   Deskripsi
     X_train,y_train,X_test, y_test :
array[]
   Implementasi
     #Mengambil anggota data cluster
terdekat
      for k in
range(len(ClusterTerdekat)):
          anggota_cluster ←
np.where(labels === ClusterTerdekat[k])
          list_anggota_cluster ←
anggota_cluster[0].toList()
          IndeksClusterTerdekat.append(list_in
dex_xclust)

     #Mencari data tetangga terdekat dari
data testing
      For h in
range(len(IndeksCLusterTerdekat)):
          a ←
X_train[IndeksClusterTerdekat[h]]
          x_train_new.append(a)
          b ←
y_train[IndeksClusterTerdekat[h]]
          y_train_new.append(b)

     #Prediksi
      for z in range(len(x_train_new)):
          model ← prediksi(k=3)
          model.datatraining(x_train_new[z
],y_train_new[z])
          hasil ←
model.prediction([X_test[z]])
```

Gambar 4.9 *Pseudocode* Algoritma *k-Nearest Neighbors*

Gambar 4.12 di atas menunjukkan data input dipecah lagi menjadi 4 yaitu data *training*, label data *training*, data *testing*, dan label data *testing*. Kemudian dari data *centroid* terdekat yang di dapat dari proses *Fuzzy c-Means*, dicari anggota cluster di dalamnya untuk dilakukan *testing*. Berikut merupakan contoh beberapa data testing yang diprediksi.

Tabel 4.6 Data *Testing*

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | Label |
|---|---|---|---|---|---|---|---|---|
| D1 | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | 2 | 2 | ? |
| D2 | 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | 2 | 2 | ? |
| D3 | 3.19 | 3.07 | 3.31 | 3.52 | 3.62 | 1 | 2 | ? |

Selanjutnya, dicari *cluster* terdekat dengan data *testing* menggunakan perhitungan jarak dengan rumus *euclidean* di bawah ini..

$$d_i = \sqrt[d_i]{\sum_{i=1}^{n} \quad (x_i - y_i)^2}$$

Tabel 4.7 Perhitungan Cluster Terdekat

| Data | Jarak ke- | | | *Cluster* Terdekat |
|---|---|---|---|---|
| | Pusat *Cluster* 0 | Pusat *Cluster* 1 | Pusat *Cluster* 2 | |
| D1 | 0.97 | 0.32 | 1.33 | 1 |
| D2 | 1.17 | 0.52 | 0.94 | 1 |
| D3 | 0.55 | 1.06 | 0.87 | 0 |

Dari hasil perhitungan dapat dilihat bahwa data D1 paling dekat dengan *cluster* 1 sehingga D1 akan diuji dengan anggota data *training* dalam *cluster* 1. Data D2 paling dekat dengan *cluster* 1 sehingga D2 akan diuji dengan anggota data *training* dalam *cluster* 1. Data D13 paling dekat dengan *cluster* 0 sehingga D13 akan diuji dengan anggota data *training* dalam *cluster* 0.

Menggunakan skenario nilai *k = 3*, berikut adalah hasil prediksi dari 3 record data *testing*.

Tabel 4.8 Hasil Prediksi

| Data | ip_1 | ip_2 | ip_3 | ip_4 | ip_5 | jk | jp | Label |
|---|---|---|---|---|---|---|---|---|
| D11 | 3.57 | 3.46 | 3.50 | 3.75 | 3.48 | 2 | 2 | Lulus Tepat Waktu |
| D12 | 3.38 | 3.00 | 3.29 | 3.50 | 3.04 | 2 | 2 | Lulus Tidak Tepat Waktu |
| D13 | 3.19 | 3.07 | 3.31 | 3.52 | 3.62 | 1 | 2 | Lulus Tidak Tepat Waktu |

## 4.4 Hasil Uji Coba

Uji coba dilakukan dengan membagi dataset menjadi data *training* dan data *testing*. Rasio pembagian data yaitu 60:40 dan 70:30. Data *training* digunakan untuk melatih model sedangkan data *testing* digunakan untuk mengevaluasi hasil dari model prediksi.

Pada bagian pertama, dibuat model prediksi dengan mengimplementasikan metode *k-Nearest Neighbors* tanpa penambahan algoritma *Fuzzy c-Means*. Kustomisasi nilai parameter *k* ditunjukkan oleh Tabel 4.9 berikut.

Tabel 4.9 Skenario Uji Coba Tanpa Optimasi menggunakan *Fuzzy c-Means*

| Model Prediksi | Skenario Nilai *k* (*k-Nearest Neighbors*) |
|---|---|
| Model 1 | $k = 1$ |
| Model 2 | $k = 3$ |
| Model 3 | $k = 5$ |
| Model 4 | $k = 7$ |

Pada uji coba kedua, dibuat model prediksi dengan mengimplementasikan metode *k-Nearest Neighbors* dengan penambahan proses klastering data menggunakan algoritma *Fuzzy c-Means*. Kustomisasi nilai parameter jumlah *cluster* (*c*) dan nilai *k* ditunjukkan oleh Tabel 4.10 berikut.

Tabel 4.10 Skenario Uji Coba dengan Optimasi menggunakan *Fuzzy c-Means*

| Model Prediksi | Skenario Jumlah *Cluster* (*Fuzzy c-Means*) | Skenario Nilai *k* (*k-Nearest Neighbors*) |
|---|---|---|
| Model 5 | *Cluster* = 2 | *k* = 1 |
| Model 6 | | *k* = 3 |
| Model 7 | | *k* = 5 |
| Model 8 | | *k* = 7 |
| Model 9 | Cluster = 3 | *k* = 1 |
| Model 10 | | *k* = 3 |
| Model 11 | | *k* = 5 |
| Model 12 | | *k* = 7 |
| Model 13 | Cluster = 4 | *k* = 1 |
| Model 14 | | *k* = 3 |
| Model 15 | | *k* = 5 |
| Model 16 | | *k* = 7 |

Uji coba menghasilkan 16 model prediksi dengan perhitungan performanya sebagai berikut.

**Model 1**

Pengujian model prediksi 1 menggunakan parameter nilai *k=1* dengan dua kali percobaan. Percobaan pertama, kombinasi rasio data *training* dan data *testing* sebesar 60:40. Dalam percobaan tersebut, menghasilkan data yang diprediksi lulus tepat waktu sejumlah 48 (True Positive), data lulus tidak tepat waktu yang diprediksi lulus tepat waktu sejumlah 26 (False Negative), data lulus tepat waktu yang diprediksi lulus tidak tepat waktu sejumlah 20 (False Positive), dan data lulus tidak tepat waktu sejumlah 64 (True Negative).

Sehingga, kinerja algoritma pada percobaan pertama (kombinasi data 60:40) dapat dihitung sebagai berikut.

Presisi = $\frac{TP}{TP + FP} \times 100\%$

$= \frac{48}{48+20} \times 100\%$

$= 70.59\ \%$

Recall $= \frac{TP}{TP + FN} \times 100\%$

$= \frac{48}{48 + 26} \times 100\%$

$= 64.86\ \%$

Akurasi$= \frac{TP + TN}{TP + FP + TN + FN} \times 100\%$

$= \frac{64 + 48}{64 + 20 + 48 + 26} \times 100\%$

$= 70.89\ \%$

Percobaan kedua, kombinasi rasio data *training* dan data *testing* sebesar 70:30. Dalam percobaan tersebut, menghasilkan data yang diprediksi lulus tepat waktu sejumlah 31 (True Positive), data lulus tidak tepat waktu yang diprediksi lulus tepat waktu sejumlah 18 (False Negative), data lulus tepat waktu yang diprediksi lulus tidak tepat waktu sejumlah 23 (False Positive), dan data lulus tidak tepat waktu sejumlah 47 (True Negative).

Sehingga, kinerja algoritma pada percobaan kedua (kombinasi data 70:30) dapat dihitung sebagai berikut.

Presisi $= \frac{TP}{TP + FP} \times 100\%$

$= \frac{31}{31 \quad 23} \times 100\%$

$= 57.41\ \%$

Recall $= \frac{TP}{TP + FN} \times 100\%$

$= \frac{31}{31 + 18} \times 100\%$

$= 63.27\ \%$

Akurasi $= \frac{TP + TN}{TP + FP + TN + FN} \times 100\%$

$= \frac{31 + 47}{31 + 23 + 47 + 18} \times 100\%$

$= 65.55\ \%$

**Model 2**

Pengujian model prediksi 2 menggunakan parameter nilai *k=3* dengan dua kali percobaan. Percobaan pertama, kombinasi rasio data *training* dan data *testing* sebesar 60:40. Dalam percobaan tersebut, menghasilkan data yang diprediksi lulus tepat waktu sejumlah 51 (True Positive), data lulus tidak tepat waktu yang diprediksi lulus tepat waktu sejumlah 24 (False Negative), data lulus tepat waktu yang diprediksi lulus tidak tepat waktu sejumlah 17 (False Positive), dan data lulus tidak tepat waktu sejumlah 66 (True Negative).

Sehingga, kinerja algoritma pada percobaan pertama (kombinasi data 60:40) dapat dihitung sebagai berikut.

Presisi $= \frac{TP}{TP + FP} \times 100\%$

$= \frac{51}{51 + 17} \times 100\%$

$= 75.00\ \%$

Recall $= \frac{TP}{TP + FN} \times 100\%$

$= \frac{51}{51 + 24} \times 100\%$

$= 68.00\ \%$

Akurasi$= \frac{TP + TN}{TP + FP + TN + FN} \times 100\%$

$= \frac{51 + 66}{51 + 17 + 66 + 24} \times 100\%$

$= 74.05\ \%$

Percobaan kedua, kombinasi rasio data *training* dan data *testing* sebesar 70:30. Dalam percobaan tersebut, menghasilkan data yang diprediksi lulus tepat waktu sejumlah 34 (True Positive), data lulus tidak tepat waktu yang diprediksi lulus tepat waktu sejumlah 14 (False Negative), data lulus tepat waktu yang diprediksi lulus tidak tepat waktu sejumlah 20 (False Positive), dan data lulus tidak tepat waktu sejumlah 51 (True Negative).

Sehingga, kinerja algoritma pada percobaan kedua (kombinasi data 70:30) dapat dihitung sebagai berikut.

Presisi $= \frac{TP}{TP + FP} \times 100\%$

$= \frac{34}{34 + 20} \times 100\%$

$= 62.96\ \%$

Recall $= \frac{TP}{TP + FN} \times 100\%$

$= \frac{34}{34 + 14} \times 100\%$

$= 70.83\ \%$

Akurasi$= \frac{TP + TN}{TP + FP + TN + FN} \times 100\%$

$= \frac{34 + 51}{34 + 20 + 51 + 14} \times 100\%$

$= 71.43\ \%$

Dilakukan perhitungan serupa untuk model prediksi 3 sampai 16 sesuai dengan skenario kustomisasi parameter nilai *c* dan nilai *k* pada Tabel 4.9 dan Tabel 4.10.

Adapun hasil perhitungan performa 16 model tersebut dilihat dari nilai akurasi, presisi, dan recall-nya sebagai berikut.

Tabel 4. 11 Hasil Uji Coba Tanpa Optimasi menggunakan Algoritma *Fuzzy c-Means*

| Model Prediksi | Skenario Nilai *k* (*KNN*) | Rasio Data *Training* & *Testing* | Presisi (%) | Recall (%) | Akurasi (%) |
|---|---|---|---|---|---|
| Model 1 | *k* = 1 | 60 : 40 | 70.59 | 64.86 | 70.89 |
| | | 70 : 30 | 57.41 | 63.27 | 65.55 |
| Model 2 | *k* = 3 | 60 : 40 | 75.00 | 68.00 | 74.05 |
| | | 70 : 30 | 62.96 | 70.83 | 71.43 |
| Model 3 | *k* = 5 | 60 : 40 | 72.06 | 70.00 | 73.65 |
| | | 70 : 30 | 62.96 | 70.83 | 71.43 |
| Model 4 | *k* = 7 | 60 : 40 | 66.18 | 69.23 | 72.78 |
| | | 70 : 30 | 66.67 | 67.92 | 70.59 |

Tabel 4.12 Hasil Uji Coba dengan Optimasi menggunakan Algoritma *Fuzzy c-Means*

| Model Prediksi | Skenario Jumlah *Cluster* (*FCM*) | Skenario Nilai *k* (*KNN*) | Rasio Data *Training* & *Testing* | Presisi (%) | Recall (%) | Akurasi (%) |
|---|---|---|---|---|---|---|
| Model 5 | *c = 2* | *k* = 1 | 60 : 40 | 70.59 | 64.86 | 70.89 |
| | | | 70 : 30 | 57.41 | 63.27 | 65.55 |
| Model 6 | *c = 2* | *k* = 3 | 60 : 40 | 75.00 | 68.00 | 74.05 |
| | | | 70 : 30 | 62.96 | 70.83 | 71.43 |
| Model 7 | *c = 2* | *k* = 5 | 60 : 40 | 72.06 | 70.00 | 74.68 |
| | | | 70 : 30 | 62.96 | 70.83 | 71.43 |
| Model 8 | *c = 2* | *k* = 7 | 60 : 40 | 66.18 | 69.23 | 72.78 |
| | | | 70 : 30 | 66.67 | 67.92 | 70.59 |
| Model 9 | *c = 3* | *k* = 1 | 60 : 40 | 70.59 | 66.67 | 72.15 |
| | | | 70 : 30 | 57.41 | 63.27 | 65.55 |
| Model 10 | *c = 3* | *k* = 3 | 60 : 40 | 73.53 | 66.67 | 72.78 |
| | | | 70 : 30 | 62.96 | 69.39 | 70.59 |
| Model 11 | *c = 3* | *k* = 5 | 60 : 40 | 70.59 | 70.59 | 74.68 |
| | | | 70 : 30 | 62.96 | 69.39 | 70.59 |
| Model 12 | *c = 3* | *k* = 7 | 60 : 40 | 66.18 | 68.18 | 72.15 |
| | | | 70 : 30 | 66.67 | 69.23 | 71.43 |
| Model 13 | *c = 4* | *k* = 1 | 60 : 40 | 72.06 | 62.03 | 68.99 |
| | | | 70 : 30 | 53.70 | 61.70 | 63.87 |

| Model Prediksi | Skenario Jumlah *Cluster* (*FCM*) | Skenario Nilai *k* (*KNN*) | Rasio Data *Training* & *Testing* | Presisi (%) | Recall (%) | Akurasi (%) |
|---|---|---|---|---|---|---|
| Model 14 | *c = 4* | *k* = 3 | 60 : 40 | 72.06 | 66.22 | 72.15 |
| | | | 70 : 30 | 61.11 | 68.75 | 69.75 |
| Model 15 | *c = 4* | *k* = 5 | 60 : 40 | 75.00 | 68.00 | 74.05 |
| | | | 70 : 30 | 75.00 | 68.00 | 74.05 |
| Model 16 | *c = 4* | *k* = 7 | 60 : 40 | 73.53 | 73.42 | 73.42 |
| | | | 70 : 30 | 38.89 | 65.62 | 63.03 |

## 4.5 Pembahasan

Berdasarkan hasil uji coba, langkah awal yaitu melakukan pengolahan data penelitian menggunakan teknik *preprocessing*. Sebanyak 631 *record* data mahasiswa dan 631 *record* data tahun lulus dari dua file yang berbeda diinputkan ke dalam algoritma *preprocessing*. Adapun tahapan *preprocessing* yang dilakukan terdiri atas 4 tahap; *data integration*, *data cleaning*, *data transformation*, dan *data reduction*. Secara garis besar, tahapan tersebut bertujuan untuk menggabungkan dua data dari dua sumber yang berbeda, menghapuskan data yang memiliki nilai rumpang atau tidak valid, mentransformasikan nilai kategorikal menjadi numerik, serta mengurangi volume matriks data dengan mengurangi kolom yang tidak dibutuhkan atau yang mengandung duplikasi. Data *preprocessing* menghasilkan 390 *record* dataset baru yang bebas dari nilai rumpang, bias, dan *outliers*.

Dataset sebanyak 390 *record* yang sudah melalui proses *preprocessing* dikombinasikan menjadi data *training* dan data *testing* dengan rasio 60 : 40 dan 70 : 30. Labelling data didasarkan pada operasi pengurangan tahun lulus dan tahun masuk mahasiswa, jika hasil kurang dari atau sama dengan 4 tahun maka diberi label ‘Lulus Tepat Waktu’, jika lebih dari 4 tahun diberi label ‘Lulus Tidak Tepat

Waktu'. Selanjutnya dilakukan implementasi *Fuzzy c-Means* untuk klastering data dan *k-Nearest Neighbors* untuk prediksi data.

Uji coba menghasilkan 16 model prediksi dengan nilai performa yang berbeda-beda, tergantung pada kustomisasi parameter *c* dan *k*. Model 1 sampai 4 diuji menggunakan algoritma *k-Nearest Neighbors* tanpa melalui proses optimasi menggunakan *Fuzzy c-Means*. Sedangkan model 5 sampai 16 diuji menggunakan algoritma *k-Nearest Neighbors* dengan penambahan proses optimasi menggunakan *Fuzzy c-Means*.

Berdasarkan hasil pada Tabel 4.11, dari keempat model prediksi *k-Nearest Neighbors* yang diuji tanpa melalui proses optimasi menggunakan *Fuzzy c-Means*, didapati Model 3 memiliki performa paling baik dengan akurasi 73.65%. Sedangkan jika dilihat hasil pada Tabel 4.12, dari 12 model prediksi *k-Nearest Neighbors* yang diuji dengan melalui proses optimasi menggunakan *Fuzzy c-Means*, didapati Model 7 dan Model 11 memiliki performa paling baik dengan akurasi 74.68%. Meskipun demikian, Model 7 dan 11 memiliki nilai *recall* dan *presisi* yang berbeda. Nilai *recall* menunjukkan jumlah mahasiswa lulus tepat waktu yang diprediksi lulus tidak tepat waktu. Sedangkan nilai presisi menunjukkan jumlah mahasiswa lulus tidak tepat waktu yang diprediksi tepat waktu. Jika dikaitkan dengan kasus ketepatan waktu kelulusan mahasiswa dan tujuan evaluasi akademik, lebih baik suatu model prediksi memiliki nilai *recall* yang tinggi dibandingkan nilai presisi yang tinggi. Oleh karena itu, meski memiliki nilai akurasi yang sama, Model 11 lebih unggul dengan nilai *recall* lebih besar daripada Model

7. Tabel 4.13 menunjukkan perbandingan pengaruh optimasi *k-Nearest Neighbors* menggunakan *Fuzzy c-Means* terhadap akurasi model prediksi yang dihasilkan.

Tabel 4.13 Perbandingan Performa Model Prediksi berdasarkan Optimasi *k-Nearest Neighbors* menggunakan *Fuzzy c-Means*

| Perlakuan | Model Terbaik | Akurasi |
|---|---|---|
| Tanpa optimasi | Model 3 | 73.65% |
| Dengan optimasi | Model 11 | 74.68% |

Dari perbandingan tersebut, dapat dilihat bahwa pengimplementasian optimasi menggunakan *Fuzzy c-Means* menghasilkan model prediksi dengan akurasi yang paling baik dari seluruh hasil uji coba, yakni 74.68%.

Kombinasi rasio data *training* dan data *testing* pada uji coba juga berpengaruh pada performa model prediksi. Tabel 4.14 menunjukkan perbandingan rerata performa model prediksi dengan rasio data *training* dan data *testing* yang digunakan.

Tabel 4.14 Perbandingan Performa Model Prediksi berdasarkan Rasio Data Input

| Rasio Data<br>*Training* & *Testing* | Rerata Akurasi |
|---|---|
| 60 : 40 | 73% |
| 70 : 30 | 69% |

Model prediksi yang dihasilkan dari sistem dengan rasio input data 60 : 40 memiliki rerata akurasi yang lebih tinggi jika dibandingkan rasio 70 : 30. Jadi, pembagian kombinasi data *training* dan data *testing* dalam membangun model prediksi yang paling baik yaitu dengan rasio 60 : 40.

Kustomisasi nilai *c* atau jumlah *cluster* pada implementasi algoritma *Fuzzy c-Means* juga berpengaruh pada performa model prediksi. Tabel 4.15 menunjukkan perbandingan rerata performa model prediksi dengan skenario jumlah *cluster* yang berbeda.

Tabel 4.15 Perbandingan Performa Model Prediksi berdasarkan Jumlah *Cluster*

| Skenario Jumlah *Cluster* *(Fuzzy c-Means)* | Rerata Akurasi |
|---|---|
| 2 | 73% |
| 3 | 73% |
| 4 | 74% |

Model prediksi dengan jumlah *cluster*=2 memiliki rerata akurasi sebesar 73%, model prediksi dengan jumlah *cluster=3* memiliki rerata akurasi sebesar 73%, dan model prediksi dengan jumlah *cluster=4* memiliki rerata akurasi sebesar 74%. Sehingga, Skenario jumlah *cluster* yang menghasilkan kumpulan model prediksi dengan rerata performa paling baik adalah skenario jumlah *cluster=4*.

Kustomisasi nilai *k* atau jumlah data tetangga uji pada implementasi metode *k-Nearest Neighbors* juga memiliki pengaruh terhadap performa model prediksi yang dihasilkan. Tabel 4.16 menunjukkan perbandingan rerata performa model prediksi dengan kustomisasi nilai *k* yang berbeda-beda.

Tabel 4.16 Perbandingan Performa Model Prediksi berdasarkan Nilai *k*

| Skenario Nilai *k* (*k-Nearest Neighbors*) | Rerata Akurasi |
|---|---|
| 1 | 71% |
| 3 | 73% |
| 5 | 75% |
| 7 | 73% |

Model prediksi yang diberi skenario nilai *k=1* memiliki rerata akurasi sebesar 71%. Sedangkan model prediksi dengan skenario nilai *k=3* dan *k=7* memiliki rerata akurasi sebesar 73%. Skenario nilai *k* yang menghasilkan model prediksi dengan performa paling baik yaitu nilai *k=5*, dengan rerata akurasi 75%.

Sehingga dapat diidentifikasi, dari seluruh skenario uji coba, model prediksi yang memiliki performa terbaik yaitu Model 11 dengan akurasi 74.68% dan spesifikasi sebagai berikut:

Tabel 4.17 Spesifikasi Model Terbaik

| | Spesifikasi Jumlah *Cluster* (*Fuzzy c-Means*) | Spesifikasi Nilai *k* (*k-Nearest Neighbors*) | Rasio Data *Training* & *Testing* |
|---|---|---|---|
| Model 11 | *c=3* | *k=5* | *60 : 40* |

Dari Tabel 4.17 dapat dilihat bahwa model prediksi terbaik dihasilkan dari implementasi algoritma *k-Nearest Neighbors* dengan nilai k=5, yang dioptimasi menggunakan algoritma *Fuzzy c-Means* dengan jumlah *cluster* paling optimal c=3. Hasil tersebut juga menunjukkan bahwa tahapan *data preprocessing* berpengaruh pada akselerasi performa model prediksi. Dilihat dari perbandingan akurasi model prediksi penelitian ini dengan penelitian sebelumnya (Nabila et al., 2021), dimana dalam penelitian tersebut hanya dilakukan 2 tahap *preprocessing* dan dihasilkan akurasi model sebesar 71%.

Berdasarkan temuan tersebut, berikut merupakan *confusion matrix* yang berisi hasil prediksi ketepatan waktu kelulusan mahasiswa dengan algoritma *k-Nearest Neighbors* yang dioptimasi menggunakan *Fuzzy c-Means*.

Tabel 4.18 Confusion Matrix Model Prediksi 11

| *Actual* | Prediksi | |
|---|---|---|
| | Tepat Waktu | Tidak Tepat Waktu |
| Tepat Waktu | 48 | 20 |
| Tidak Tepat Waktu | 20 | 70 |

Didapatkan hasil prediksi ketepatan waktu kelulusan mahasiswa Program Studi Teknik Informatika angkatan 2014-2018 menggunakan metode *fuzzy c-means*

dan *k-nearest neighbors* adalah sebanyak 48 mahasiswa yang lulus tepat waktu terprediksi benar lulus tepat waktu, sebanyak 20 mahasiswa yang lulus tepat waktu terprediksi lulus tidak tepat waktu, sebanyak 20 mahasiswa lulus tidak tepat waktu terprediksi lulus tepat waktu, dan 70 mahasiswa lulus tidak tepat waktu terprediksi benar lulus tidak tepat waktu. Sehingga dari hasil tersebut dapat diketahui prediksi tingkat kelulusan tepat waktu mahasiswa angkatan 2014-2018 yaitu 43 %. Output persentase tersebut dapat dibandingkan dengan standar yang disyaratkan oleh BAN-PT, maka hal ini bisa menjadi acuan bagi program studi untuk lebih dini menyusun atau memperbarui kebijakan yang berhubungan dengan evaluasi akademik guna memperbaiki kualitas mahasiswa dan program studi.

# BAB V

# PENUTUP

## 5.1 Kesimpulan

Berdasarkan hasil uji coba dan pembahasan, pemilihan parameter jumlah *cluster* (c), nilai *k*, serta rasio data input sangat penting karena mempengaruhi performa model prediksi yang dibangun. Parameter *c=4* menghasilkan rerata performa model prediksi terbaik dibandingkan dengan skenario jumlah *cluster* lain, yaitu 74%. Parameter nilai *k=5* menghasilkan rerata performa model prediksi terbaik jika dibandingkan dengan skenario nilai *k* lain, yaitu 75%. Rasio data *training* dan data *testing* 60:40 menghasilkan model prediksi dengan rerata akurasi terbaik dibandingkan rasio lain, yaitu 73%.

Dalam penelitian ini ditemukan model prediksi terbaik tanpa optimasi memiliki akurasi 73.65%, sedangkan model prediksi terbaik dengan penambahan optimasi menggunakan *Fuzzy c-Means*, menghasilkan akurasi 74.8%. Adapun model prediksi tersebut memiliki spesifikasi parameter jumlah *cluster* paling optimal *c=3*, parameter nilai *k=5*, rasio data *training* dan *testing* 60:40. Sehingga dapat disimpulkan bahwa optimasi algoritma *k-Nearest Neighbors* menggunakan *Fuzzy c-Means* berdampak pada peningkatan performa model prediksi ketepatan waktu kelulusan mahasiswa. Selain itu, temuan angka akurasi tersebut lebih tinggi dari penelitian sebelumnya (Nabila et al, 2021), membuktikan bahwa pengimplementasian 4 tahap *preprocessing* juga berdampak pada peningkatan kinerja algoritma.

Optimasi algoritma *k-Nearest Neighbors* menggunakan *Fuzzy c-Means* juga berhasil menghasilkan model yang dapat memprediksi ketepatan waktu kelulusan mahasiswa Program Studi Teknik Informatika Fakultas Sains dan Teknologi UIN Malang angkatan 2014-2018, yakni prediksi mahasiswa lulus tepat waktu sejumlah 68, mahasiswa lulus tidak tepat waktu sejumlah 90, serta tingkat kelulusan tepat waktu yaitu 43%. Hasil tersebut dapat dibandingkan dengan standar yang disyaratkan oleh BAN-PT, sehingga bisa dijadikan acuan bagi program studi untuk lebih dini menyusun atau memperbarui kebijakan yang berhubungan dengan evaluasi akademik guna memperbaiki kualitas mahasiswa dan program studi.

## 5.2 Saran

Berikut merupakan saran yang dapat digunakan untuk pengembangan penelitian selanjutnya dengan topik dan permasalahan serupa:

1. Penambahan atribut lain dalam perhitungan, yang memiliki pengaruh pada ketepatan waktu kelulusan mahasiswa. Terutama atribut abstrak seperti status pekerjaan selama studi, status pondok pesantren, dan lain-lain.
2. Pemilihan atribut ‘jenis kelamin’ dan ‘jenis pembiayaan’ pada penelitian ini masih mengacu pada penggunaannya di penelitian sebelumnya. Penelitian selanjutnya dapat membahas lebih dalam tentang pengaruh keduanya terhadap ketepatan waktu kelulusan mahasiswa.
3. Perlu dilakukan penambahan ragam skenario uji coba (komposisi data *training* & data *testing*, parameter nilai *c*, parameter nilai *k*) untuk mendapatkan model prediksi dengan akurasi yang lebih baik lagi.

# DAFTAR PUSTAKA


Adithiyaa, T., Chandramohan, D., & Sathish, T. (2020). Optimal prediction of process parameters by GWO-KNN in stirring-squeeze casting of AA2219 reinforced metal matrix composites. Materials Today: Proceedings, 21, 1000-1007.

Amalia, Y. R. (2018). Penerapan Data Mining Untuk Prediksi Penjualan Produk Elektronik Terlaris Menggunakan Metode K-Nearest Neighbor (Studi Kasus: PT. Bintang Multi Sarana Palembang) (Doctoral dissertation, UIN RADEN FATAH PALEMBANG).

BAN-PT. (2022). Naskah Instrumen PEPA-PT Akademik dan Vokasi. Jakarta: Badan Akreditasi Nasional.

Banjarsari, M. A., Budiman, I., & Farmadi, A. (2016). Penerapan K-Optimal Pada Algoritma Knn Untuk Prediksi Kelulusan Tepat Waktu Mahasiswa Program Studi Ilmu Komputer Fmipa Unlam Berdasarkan Ip Sampai Dengan Semester 4. Klik-Kumpulan Jurnal Ilmu Komputer, 2(2), 159-173.

Bezdek, J. C. (2013). Pattern recognition with fuzzy objective function algorithms. Springer Science & Business Media.

Butarbutar, N., Windarto, A. P., Hartama, D., & Solikhun, S. (2017). Komparasi kinerja algoritma fuzzy c-means dan k-means dalam pengelompokan data siswa berdasarkan prestasi nilai akademik siswa. Jurasik (Jurnal Riset Sistem Informasi dan Teknik Informatika), 1(1), 46-55.

Dahlan, I. A. (2022). Klasifikasi Cuaca Provinsi Dki Jakarta Menggunakan Algoritma Random Forest Dengan Teknik Oversampling. Jurnal Teknoinfo, 16(1), 87-92.

Danny, H. (2014). Aplikasi Data Mining Menggunakan Algoritma ID3 Untuk Mengklasifikasi Kelulusan Mahasiswa Pada Universitas Dian Nuswantoro Semarang. Skripsi, Fakultas Ilmu Komputer.

Deviyanto, A., & Wahyudi, M. D. R. (2018). Penerapan analisis sentimen pada pengguna twitter menggunakan metode K-Nearest Neighbor. JISKA (Jurnal Informatika Sunan Kalijaga), 3(1), 1-13.

Fauziah, I. N., Suratno, L. N. N., & Rakhmadi, F. A. (2020). Menelaah Konsep Fluida dalam QS Al-Baqarah Ayat 164 Menggunakan Pendekatan Filsafat Ilmu. Prosiding Konferensi Integrasi Interkoneksi Islam dan Sains, 2, 195-198.

Han, J., Pei, J., & Tong, H. (2022). Data mining: concepts and techniques. Morgan kaufmann. (Dapat diakses secara online di https://books.google.co.id/books?id=NR1oEAAAQBAJ&lpg=PP1&ots=_M8LQMygo-&dq=Data%20mining%3A%20concepts%20and%20techniques&lr&pg=PR7#v=onepage&q&f=false)

Huang, J., Keung, J. W., Sarro, F., Li, Y. F., Yu, Y. T., Chan, W. K., & Sun, H. (2017). Cross-validation based K nearest neighbor imputation for software quality datasets: an empirical study. Journal of Systems and Software, 132, 226-252.

Jamhur, H. (2020). Pemodelan Prediksi Predikat Kelulusan Mahasiswa Menggunakan Fuzzy C-Means Berbasis Particle Swarm Optimization. Teknois: Jurnal Ilmiah Teknologi Informasi dan Sains, 10(1), 13-24.

Jia, B. B., & Zhang, M. L. (2021, July). Multi-dimensional classification via sparse label encoding. In International Conference on Machine Learning (pp. 4917-4926). PMLR.

Kamil, M., & Cholil, W. (2020). Analisis Perbandingan Algoritma C4. 5 dan Naive Bayes pada Lulusan Tepat Waktu Mahasiswa di Universitas Islam Negeri Raden Fatah Palembang. Jurnal Informatika, 7(2), 97-106.

Kusiak, A., Wei, X., Verma, A. P., & Roz, E. (2012). Modeling and prediction of rainfall using radar reflectivity data: A data-mining approach. IEEE Transactions on Geoscience and Remote Sensing, 51(4), 2337-2342.

Larose, D. T., & Larose, C. D. (2014). Discovering knowledge in data: an introduction to data mining (Vol. 4). John Wiley & Sons.

Latifah, R., Wulandari, E. S., & Kreshna, P. E. (2019). Model Decision Tree Untuk Prediksi Jadwal Kerja Menggunakan Scikit-Learn. Prosiding Semnastek.

Li, D., Gu, H., & Zhang, L. (2013). A hybrid genetic algorithm–fuzzy c-means approach for incomplete data clustering based on nearest-neighbor intervals. Soft Computing, 17, 1787-1796.

Maghari, A. (2018). Prediction of student's performance using modified KNN classifiers. In Alfere, SS, & Maghari, AY (2018). Prediction of Student's Performance Using Modified KNN Classifiers. In The First International Conference on Engineering and Future Technology (ICEFT 2018) (pp. 143-150).

Malley, B., Ramazzotti, D., & Wu, J. T. Y. (2019). Data pre-processing. In: Secondary Analysis of Electronic Health Records. Springer, Cham. https://doi.org/10.1007/978-3-319-43742-2_13

Nabila, S. P., Ulinnuha, N., & Yusuf, A. (2021). Model Prediksi Kelulusan Tepat Waktu Dengan Metode Fuzzy C-Means Dan K-Nearest Neighbors Menggunakan Data Registrasi Mahasiswa. Network Engineering Research Operation, 6(1), 38-46.

Pal, K., & Patel, B. V. (2020, March). Data classification with k-fold cross validation and holdout accuracy estimation methods with 5 different machine learning techniques. In 2020 fourth international conference on computing methodologies and communication (ICCMC) (pp. 83-87). IEEE.

Pddikti.kemdikbud.go.id. (2021). Data Prodi Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang. Diakses pada 22 Januari 2022, dari https://pddikti.kemdikbud.go.id/data_prodi/QTNFRDUzMEMtM0MyRS00RjJELUIxNzEtNDA4NjQ4REM0RTc2/20201

Priandini, N., Zaman, B., & Purwanti, E. (2017, August). Categorizing document by fuzzy C-Means and K-nearest neighbors approach. In AIP Conference Proceedings (Vol. 1867, No. 1, p. 020012). AIP Publishing LLC.

Ristekdikti.go.id. (2019). Pertanyaan Umum Seputar Bidikmisi. Diakses pada 14 Maret 2023, dari https://bidikmisi.belmawa.ristekdikti.go.id/petunjuk/20

Rohmawan, E. P. (2018). Prediksi Kelulusan Mahasiswa Tepat Waktu Menggunakan Metode Decision Tree Dan Artificial Neural Network. Jurnal Ilmiah Matrik, 20(1), 21-30.

Rusyda, A. A. S. A., Isnandar, A., Amin, A. B., Anwar, R., & Muhyi, A. A. (2023, June). Demokrasi dalam Islam Perspektif Al-Qur'an. In Gunung Djati Conference Series (Vol. 25, pp. 169-185).

Santrock, J. W. (2014). Psikologi Pendidikan. Edisi 5 Jilid 1. (Harya Bhimasena Translator). Jakarta: Salemba Humanika.

Suprapto, S., Zubaidah, S., & Corebima, A. D. (2018). Pengaruh gender terhadap keterampilan berpikir kreatif siswa pada pembelajaran biologi. Jurnal Pendidikan: Teori, Penelitian, dan Pengembangan, 3(3), 325-329.

Sun, Y., Xu, Y., Ma, L., & Deng, Z. (2009, December). KNN-FCM hybrid algorithm for indoor location in WLAN. In 2009 2nd International Conference on Power Electronics and Intelligent Transportation System (PEITS) (Vol. 2, pp. 251-254). IEEE.

Son, N. H. (2006). Data mining course—data cleaning and data preprocessing. Warsaw University. Available at URL https://www.mimuw.edu.pl/~son/datamining/DM/4-preprocess.pdf

Soni, J., Ansari, U., Sharma, D., & Soni, S. (2011). Predictive data mining for medical diagnosis: An overview of heart disease prediction. International Journal of Computer Applications, 17(8), 43-48.

Shehab, N., Badawy, M., & Arafat, H. (2021). Big data analytics and preprocessing. Machine learning and big data analytics paradigms: analysis, applications and challenges, 25-43.

Tempola, F., Muhammad, M., & Khairan, A. (2018). Perbandingan Klasifikasi Antara KNN dan Naive Bayes pada Penentuan Status Gunung Berapi dengan K-Fold Cross Validation. Jurnal Teknologi Informasi dan Ilmu Komputer, 5(5), 577-584.

Vijiyarani, S., & Sudha, S. (2013). Disease prediction in data mining technique–a survey. International Journal of Computer Applications & Information Technology, 2(1), 17-21.

Wahid, A. Y. I. A., & Girsang, A. S. (2020). Graduate Prediction in University using Fuzzy Logic. International Journal, 9(2).

Wahidah, F. (2019). Konsep Tarbiyah Dalam Perspektif Surat Az-Zumar Pendekatan Tafsir Ijmali. Qolamuna: Jurnal Studi Islam, 5(1), 97-110.

Yi, H. (2010). Highlighting Subjectivity of Students:An Inevitable Choice of Student Assessment Reform in University. Theory and practice of education.

Zainuddin, M. (2018). Perbandingan 4 Algoritma Berbasis Particle Swarm Optimization (pso) Untuk Prediksi Kelulusan Tepat Waktu Mahasiswa. Jurnal Ilmiah Teknologi Informasi Asia, 13(1), 1-12.