



## Implementasi Metode *K-Means* Untuk *Clustering* Data Penduduk Miskin Dengan *Systematic Random Sampling*

Rafli Junaidi Kasim<sup>1</sup>, Samsul Bahri<sup>2</sup>, Syukirman Amir<sup>3</sup>

<sup>1,2,3</sup>Magister Teknik Informatika, Fakultas Teknik, Universitas AMIKOM Yogyakarta  
jkasim840@gmail.com

### Abstract

The poor population grouping aims to differentiate the population with the highest or the most appropriate level of poverty to get assistance specifically for the population with the highest level of poverty. Grouping is done by using the *k-means* method. Grouping with the *k-means* method will increase the level of similarity in groups and reduce the level of similarity between groups. Random grouping on *k-means* will be applied systematic random sampling methods that will influence and narrow down the possibility of many initial centroid values to be generated, while speeding up the computation process for random grouping. Furthermore, the silhouette coefficient is validated to determine the best group in grouping the poor population. The number of groups determined is 2 clusters, 3 clusters, and 4 clusters. The results obtained are the number of groups of 2 clusters is better than 3 clusters and 4 clusters with a value of 2 clusters namely 0.435489, while in 3 clusters 0.434857 and 4 clusters 0.30832.

Keyword: Clustering, *K-means*, Systematic Random Sampling, Silhouette Coefficient, Poor Population

### Abstrak

Pengelompokan penduduk miskin bertujuan untuk membedakan penduduk dengan tingkat kemiskinan paling tinggi atau paling layak untuk mendapatkan bantuan yang dikhususkan kepada penduduk dengan tingkat kemiskinan tertinggi. Pengelompokan dilakukan dengan menggunakan metode *k-means*. Pengelompokan dengan metode *k-means* akan memperbesar tingkat kemiripan dalam kelompok dan memperkecil tingkat kemiripan antar kelompok. Pengelompokan acak pada *k-means* akan diterapkan metode *systematic random sampling* yang akan mempengaruhi dan mempersempit kemungkinan banyak nilai *centroid* awal yang akan dihasilkan, selain itu mempercepat proses komputasi untuk pengelompokan acak. Selanjutnya dilakukan validasi dengan *silhouette coefficient* untuk menentukan kelompok terbaik pada pengelompokan penduduk miskin. Jumlah kelompok yang ditentukan yaitu 2 klaster, 3 klaster, dan 4 klaster. Hasil yang didapatkan adalah jumlah kelompok sebanyak 2 klaster lebih baik dibandingkan 3 klaster dan 4 klaster dengan nilai 2 klaster yaitu 0.435489, sedangkan pada 3 *cluster* 0.434857 dan 4 *cluster* 0.30832.

Kata Kunci: Clustering, *K-means*, Systematic Random Sampling, Silhouette Coefficient, Penduduk miskin

### 1. Pendahuluan

Kemiskinan merupakan sebuah permasalahan yang kompleks, tidak hanya pada negara-negara berkembang seperti Indonesia tetapi juga pada negara maju. Pada bulan maret tahun 2018, Badan Pusat Statistik mencatat jumlah penduduk miskin di Indonesia mencapai 25,95 juta jiwa (9,82 persen).

Berbagai upaya telah dilakukan oleh pemerintah untuk menanggulangi kemiskinan di Indonesia. Salah satu bentuk upaya dari pemerintah adalah dengan menerapkan bantuan program penanggulangan kemiskinan. Akan tetapi masalah yang terjadi adalah masih terdapat masyarakat miskin yang belum menerima bantuan dan bantuan yang tersalurkan belum tepat sasaran atau tidak layak dalam menerima bantuan. Agar masalah tersebut dapat teratasi akan dikelompokkan penduduk miskin sehingga terbentuk sebuah kelompok dengan masyarakat yang paling layak untuk mendapatkan bantuan program penanggulangan kemiskinan.

Untuk melakukan pengelompokan dibutuhkan metode *clustering* pada *data mining*. Metode *clustering* adalah sebuah metode untuk mengelompokkan atau mengklaster setiap data yang ada pada sebuah set data berdasarkan sifat yang dimiliki oleh setiap data. Metode *clustering* yang digunakan dalam penelitian ini adalah metode *k-means* dimana metode ini akan mengklaster berdasarkan jarak setiap data dengan *titik pusat* pada masing-masing klaster sehingga kemiripan yang ada pada satu kelompok akan diminimalkan dan kemiripan antar kelompok akan dimaksimalkan. Kelebihan dari metode *k-means* terletak pada implementasi yang sederhana, mampu mengklaster data dalam jumlah yang cukup besar, dan kemampuan komputasi yang relatif cepat (Alfina *et al*, 2012). Akan tetapi metode ini juga memiliki kekurangan di karenakan penentuan jumlah klaster secara manual sehingga hasil *clustering* akan tidak optimal apabila tidak tepat dalam menentukan jumlah klaster. Untuk menangani hal tersebut akan digunakan metode *silhouette coefficient* untuk melihat kualitas kekuatan

hasil kluster agar nantinya dapat diketahui jumlah kluster mana yang paling optimal. Selain itu kekurangan lainnya pada metode *k-means* adalah pada alokasi data acak pertama dapat menghasilkan kemungkinan nilai *centroid* awal yang lebih banyak sehingga menghasilkan hasil *clustering* yang berbeda-beda apabila proses dilakukan secara berulang, serta alokasi data acak pertama dapat terjadi lebih lama dalam proses komputasi karena iterasi yang terjadi tidak dapat diprediksi. Untuk itu pada tahap alokasi data acak pertama metode *k-means* akan digunakan metode *systematic random sampling* yang berguna untuk meminimalisir nilai *centroid* awal yang dihasilkan, memungkinkan alokasi dapat lebih cepat dilakukan dalam proses komputasi dan implementasi yang lebih sederhana. Untuk itulah penulis mengangkat judul “Implementasi Metode *K-Means* Untuk *Clustering* Data Penduduk Miskin Dengan Pengelompokan Acak Menggunakan *Systematic Random Samplin*.”

## 2. Tinjauan Pustaka

### 2.1. K-Means

*K-means* merupakan salah satu metode yang termasuk kedalam metode non hierarki atau “menggunakan pendekatan *partitional clustering*” (Fajar, 2013). Metode *k-means* sangat umum digunakan untuk melakukan *clustering*. Selain melakukan *clustering*, *k-means* juga dapat diterapkan pada klasifikasi (Suprawoto, 2016). Metode *k-means* adalah metode yang paling mendasar pada analisa kluster dan biasanya digunakan sebagai pemrosesan awal atau dikombinasikan dengan metode lain. Untuk menggunakan metode *k-means* pertama-tama ditentukan terlebih dahulu nilai dari jumlah kluster, kemudian titik pusat dari masing-masing kluster ditentukan secara random lalu setiap data dihitung jarak dengan titik pusat dan pengelompokan dilakukan berdasarkan jarak terdekat pada setiap data dengan titik pusat. *K-means method* juga memiliki kelebihan dan kekurangan seperti yang ditulis dibawah ini.

Kelebihan *K-means method*:

1. Implementasi yang cukup mudah.
2. Mampu mengolah data yang cukup besar
3. Waktu yang terbilang cepat untuk menjalankan pembelajaran.

Kekurangan *K-means method*:

1. Menentukan nilai k secara manual sehingga tidak bisa dipastikan nilai k mana yang paling terbaik.
2. Lemah terhadap jumlah dimensi yang tinggi.

Untuk menghitung jarak, *k-means* menggunakan beberapa rumus yaitu *euclidean* yang umum dipergunakan dan digunakan untuk mencari jarak terdekat, kemudian *manhattan*, *minkowsky*, dan yang terakhir *chebysev*.

Pengelompokan dengan metode *k-means* secara umum dilakukan menggunakan algoritma berikut (Prasetyo,

- 2012):
1. Menentukan jumlah kelompok.
  2. Mengalokasikan data secara acak kedalam kelompok.
  3. Menghitung *centroid* awal atau titik pusat masing-masing kelompok.
  4. Mengalokasikan data ke *centroid* sesuai jarak terdekat setiap data.
  5. Ulangi langkah 3 apabila masih ada data yang berpindah kluster.
- Pada langkah ke tiga, untuk perhitungan *centroid* digunakan persamaan (1) (Prasetyo, 2012).

$$C_i = \frac{1}{M} \sum_{j=1}^M X_j \quad (1)$$

Keterangan :

- C : *Centroid*  
 M : Jumlah data  
 X : Total data pada suatu dimensi

Pada penelitian ini menggunakan rumus *manhattan* yang dapat dilihat pada persamaan 1 (Miyamoto dan Agusta, 1995 *as cited in* Prasetyo 2012):

$$D(X_2, X_1) = \|X_2 - X_1\|_1 = \sum_{j=1}^p |X_{2j} - X_{1j}| \quad (2)$$

Keterangan :

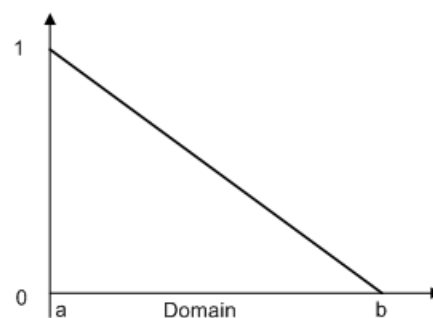
- D : Jarak  
 X2 : Data salah satu dokumen  
 X1 : Data *centroid*  
 / . / : Nilai mutlak

### 2.2. Representasi Linear

Pada representasi linear, pemetaan *Input* ke derajat keanggotaannya digambarkan sebagai suatu garis lurus. Bentuk ini paling sederhana dan menjadi pilihan yang baik untuk mendekati suatu konsep yang kurang jelas [4].

Ada 2 keadaan himpunan *Fuzzy* yang linear:

1. Representasi Linear Turun  
 Garis lurus dimulai dari nilai domain dengan derajat keanggotaan tertinggi pada sisi kiri, kemudian bergerak menurun ke nilai domain yang memiliki derajat keanggotaan yang lebih rendah. Representasi linear turun dapat dilihat pada gambar 1.



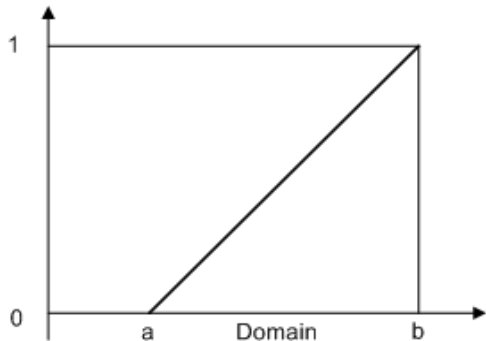
Gambar 1. Representasi linear turun

Rumus fungsi keanggotaan representasi linear turun

$$\left\{ \begin{array}{ll} 1; & x \leq a \\ \frac{b-x}{b-a}; & a \leq x \leq b \\ 0; & x \geq b \end{array} \right\}$$

2. Representasi Linear Naik

Kenaikan Himpunan dimulai pada nilai domain yang memiliki derajat keanggotaan nol (0) bergerak ke kanan menuju ke nilai domain yang memiliki derajat keanggotaan lebih tinggi, Representasi linear naik dapat dilihat pada gambar 2.



Gambar 2. Representasi linear naik

Rumus fungsi keanggotaan representasi linear naik

$$\left\{ \begin{array}{ll} 0; & x \leq a \\ \frac{x-a}{b-a}; & a \leq x \leq b \\ 1; & x \geq b \end{array} \right\}$$

2.3. Naïve Bayes

Algoritma Naïve Bayes merupakan sebuah metode klasifikasi menggunakan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes. Algoritma Naïve Bayes memprediksi peluang di masa depan berdasarkan pengalaman di masa sebelumnya sehingga dikenal sebagai Teorema Bayes [5].

Keuntungan penggunaan adalah bahwa metoda ini hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Karena yang diasumsikan sebagai variabel independent, maka hanya varians dari suatu variabel dalam sebuah kelas yang dibutuhkan untuk menentukan klasifikasi, bukan keseluruhan dari matriks kovarians.

Tahapan Naïve Bayes

1. Menghitung Mean dan Standar Deviasi ( Data Numerik)  
 Mencari Mean dapat dilihat pada persamaan 3.

$$\bar{x} = \frac{1}{n} \sum_{i=1}^n x_i \dots\dots\dots (3)$$

2. Mencari Standar Deviasi dapat dilihat pada persamaan 4.

$$s^2 = \frac{\sum_{i=1}^n (x_i - \bar{x})^2}{n-1} \dots\dots\dots (4)$$

3. Menghitung Probabilitas (Data Diskrit) dapat dilihat pada persamaan 5.

$$P(E) = X/N \dots\dots\dots (5)$$

4. Menghitung *Densitas Gauss* dapat dilihat pada persamaan 6.

$$f(x_i) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}} \dots\dots\dots (6)$$

5. Menghitung *Likelihood* dapat dilihat pada persamaan 7.

$$L(x) = f(x_1) \times f(x_2) \times \dots \times f(x_n) \dots\dots\dots (7)$$

6. Menghitung Probabilitas *Likelihood* dapat dilihat pada persamaan 8.

$$P(L_i) = \frac{L_i}{\sum_{i=1}^n L_i} \dots\dots\dots (8)$$

3. Metode Penelitian

3.1. Objek Penelitian

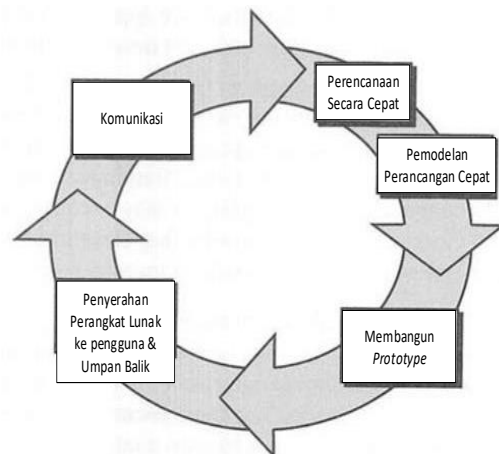
Penulis melakukan penelitian terhadap pembuatan aplikasi status UKT menggunakan *Fuzzy Naïve Bayes* dan perbandingan akurasi *Naïve Bayes* dan *Fuzzy Naïve Bayes*.

3.2. Metode Pengumpulan Data

Pengumpulan data yaitu melalui tahap wawancara dan pengambilan data dokumen tertulis dari tim UKT Universitas Khairun.

3.3. Metode Pengembangan Sistem

Metode pengembangan yang digunakan dalam pembuatan aplikasi penentuan status UKT adalah metode *Prototype*. Metode *prototype* adalah metode dalam pengembangan rekayasa *software* yang bertahap dan berulang, serta mementingkan sisi *user* sistem. Metode Pengembangan Aplikasi yang digunakan untuk membangun aplikasi kriptografi SMS yaitu dengan menggunakan metode *prototype* yang terdiri dari 5 tahapan pengembangan perangkat lunak yaitu Tahap Komunikasi, Tahap Perencanaan secara cepat, Tahap Pemodelan Perancangan secara cepat, Tahap Pembentukan *Prototype*, Tahap Pembuatan Sistem Serta Penyerahan & Umpan Balik. Berikut tahapan-tahapan pada model *prototype* ini dapat dilihat pada gambar 3.

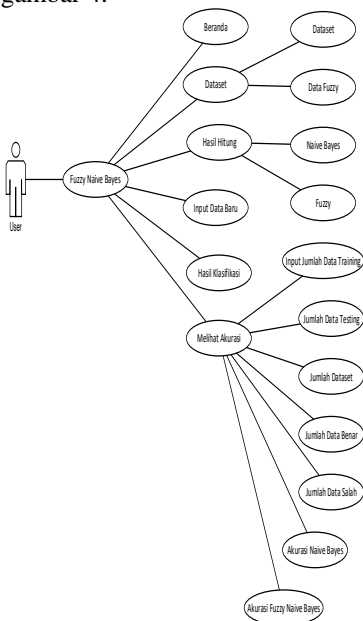


Gambar 3. Metode *Prototype*

1. Tahap Komunikasi: Pada tahap ini, akan dikumpulkan data UKT, kemudian mengidentifikasi secara garis besar kebutuhan apa saja yang akan digunakan dalam pembuatan aplikasi penentuan status UKT menggunakan metode *Fuzzy Naïve Bayes*
2. Tahap Perencanaan Secara Cepat: Pada tahap perencanaan pembuatan *prototyping* dilakukan secara cepat, namun dalam bentuk rencana awal. Hasil dari rencana awal yakni, *Flowchart* metode *Naïve Bayes*, *Flowchart* metode *Fuzzy Naïve Bayes*, *Use Case Diagram* dan perancangan *database*.
3. Tahap Pemodelan Perancangan Secara Cepat: Pada tahap ini akan dianalisa kebutuhan-kebutuhan perangkat lunak yang akan di butuhkan nantinya dan akan dibuatkan desain *interface* aplikasi penentuan status UKT.
4. Tahap Pembentukan *Prototype*: Pada tahap ini akan dilakukan penulisan kode kedalam bahasa pemrograman PHP sesuai dengan tahap perancangan yang telah dibuat dan *tools* yang telah ditentukan sebelumnya. Kemudian akan dilakukan pengujian sistem dengan *White box testing*. Tahap pengujian ini bertujuan untuk menunjukkan apakah algoritma dalam program telah sesuai.
5. Tahap Pembuatan Sistem serta Penyerahan & Umpan Balik: Sistem yang sudah lulus tahapan pengujian dibuat *prototype* untuk kemudian diserahkan ke pengguna atau pengembang sistem.

### 3.4. Perancangan Aplikasi

Pada tahap ini penulis akan menggambarkan sistem yang akan dibangun dengan menggunakan *Use Case Diagram* sebagai media visualisasi sebuah perangkat lunak atau sistem. Adapun *Use Case Diagram*nya dapat dilihat pada gambar 4.



Gambar 4. *Use Case Diagram*

Berdasarkan Gambar 4, *User* dapat mengakses dan melakukan beberapa hal yaitu:

1. *User* dapat mengakses *dataset*, di dalam *dataset* ada 2 menu yaitu menu *dataset* dan menu data *Fuzzy*, menu *dataset* adalah menu untuk melihat data UKT asli, sedangkan menu data *Fuzzy* adalah menu untuk melihat data UKT yang sudah dihitung menggunakan metode *Fuzzy*.
2. *User* dapat melihat hasil hitung berdasarkan metode *Naïve Bayes* dan *Fuzzy*.
3. *User* dapat mengInput data baru yang akan diklasifikasi menggunakan metode *Naïve Bayes* dan metode *Fuzzy Naïve Bayes*.
4. *User* dapat melihat data-data yang telah diklasifikasi menggunakan metode *Naïve Bayes* dan *Fuzzy Naïve Bayes*.
5. *User* dapat menentukan jumlah *data training* untuk menghitung akurasi.

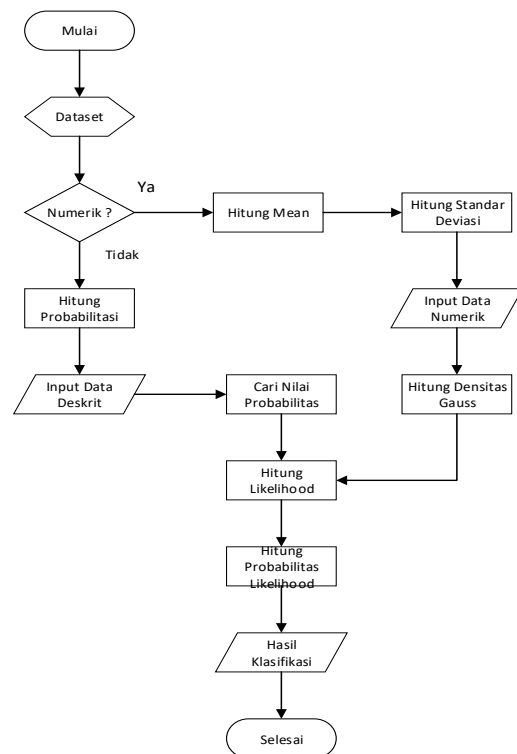
### 4. Hasil Dan Pembahasan

Pada tahap ini dibahas tentang bagaimana mengklasifikasi status UKT mahasiswa dan mencari akurasi metode yang digunakan dengan kriteria-kriteria yang telah ditentukan dengan jumlah data yang digunakan sebanyak 99 data yang diambil dari tim UKT Universitas Khairun.

#### 4.2. Flowchart

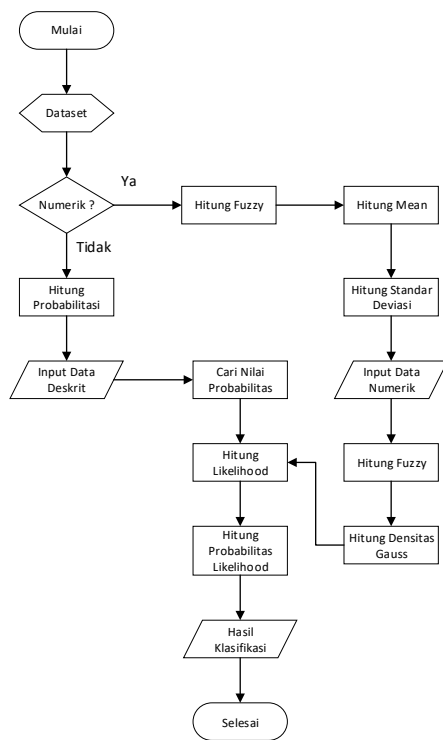
*Flowchart* atau Diagram Alir ini memperlihatkan tahap-tahap proses metode yang digunakan yaitu metode *Naïve Bayes* dan *Fuzzy Naïve Bayes*.

##### 1. Flowchart Naïve Bayes



Gambar 5. *Flowchart Naïve Bayes*

Flowchart Fuzzy Naïve Bayes

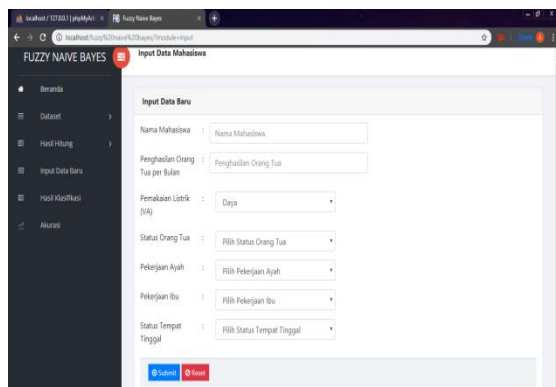


Gambar 6. Flowchart Fuzzy Naïve Bayes

4.3. Implementasi Interfaces

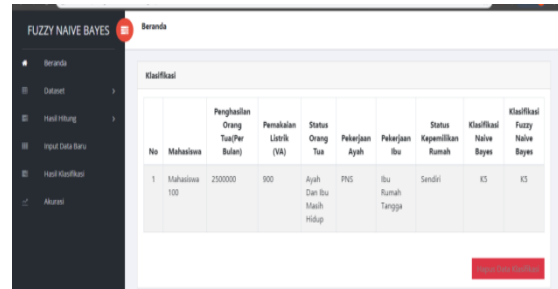
Dalam aplikasi penentuan status UKT menggunakan Fuzzy Naïve Bayes ini mengimplementasikan beberapa interfaces utama. Diantaranya adalah:

1. Tampilan Menu *Input*: Halaman menu *input* menampilkan form *Inputan* untuk klasifikasi data baru. Tampilan menu *Input* dapat dilihat pada gambar 7.



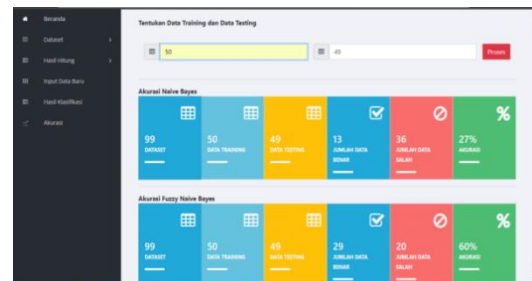
Gambar 7. Tampilan Menu *Input*

2. Tampilan Hasil Klasifikasi: Tampilan halaman hasil klasifikasi menampilkan hasil klasifikasi dari metode *Naïve Bayes* dan *Fuzzy Naïve Bayes* yang telah di *Input* di menu *Input*. Tampilan halaman Hasil Kasifikasi dapat dilihat pada gambar 8.



Gambar 8. Tampilan Menu Hasil Klasifikasi

3. Tampilan Akurasi: Tampilan halaman akurasi menampilkan form *Inputan* jumlah *data testing* dan *data training* yang akan digunakan untuk menghitung akurasi setelah proses maka akan ditampilkan hasil akurasi dari metode *Naïve Bayes* dan *Fuzzy Naïve Bayes*. Tampilan halaman akurasi dapat dilihat pada gambar 9.



Gambar 9. Tampilan Akurasi

4.4. Akurasi

Untuk membandingkan akurasi metode *Naïve Bayes* dan *Fuzzy Naïve Bayes*, dilakukan pengujian metode yaitu:

1. Pengujian *Naïve Bayes*

Pada pengujian ini digunakan 50 *data training* dan 20 *data testing*. Pengujian pertama menggunakan 4 kriteria yaitu penghasilan orang tua, daya pemakaian listrik, status orang tua dan kepemilikan rumah. Setelah dihitung didapatkan 14 data benar dan 6 data salah sehingga untuk mendapatkan akurasi dapat dihitung sebagai berikut:

$$\frac{14}{14+6} \times 100\% = 70\%$$

Jadi, dengan menggunakan 4 kriteria, tingkat akurasi *Naïve Bayes* yang didapat adalah 70%.

Pengujian kedua menggunakan 6 kriteria yaitu penghasilan orang tua, daya pemakaian listrik, status orang tua, kepemilikan rumah, pekerjaan ayah dan pekerjaan ibu. Setelah dihitung didapat 18 data benar dan 2 data salah sehingga untuk mendapatkan akurasi dapat dihitung dengan cara berikut:

$$\frac{18}{18+2} \times 100\% = 90\%$$

Jadi, dengan menggunakan 6 kriteria, tingkat Akurasi *Naïve Bayes* sebesar 90%.

## 2. Pengujian *Fuzzy Naïve Bayes*

Pada pengujian ini digunakan data yang sama yaitu 50 *data training* dan 20 *data testing*. Pengujian pertama menggunakan 4 kriteria yaitu penghasilan orang tua, daya pemakaian listrik, status orang tua dan kepemilikan rumah. Setelah dihitung didapat 11 data benar dan 9 data salah sehingga:

$$\frac{11}{11+9} \times 100\% = 55\%$$

Jadi, dengan menggunakan 4 kriteria, akurasi *Fuzzy Naïve Bayes* yaitu 55%.

Pengujian kedua menggunakan 6 kriteria yaitu penghasilan orang tua, daya pemakaian listrik, status orang tua, kepemilikan rumah, pekerjaan ayah dan pekerjaan ibu. Setelah dihitung didapat 17 data benar dan 3 data salah sehingga akurasi dapat dihitung dengan cara sebagai berikut:

$$\frac{17}{17+3} \times 100\% = 85\%$$

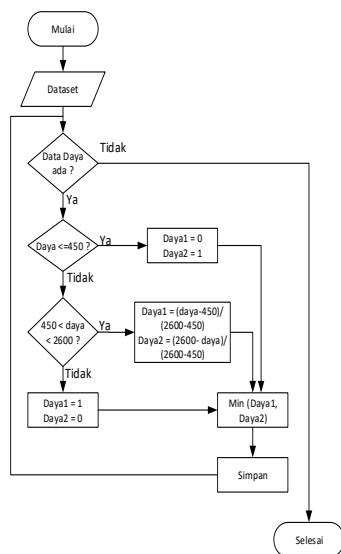
Jadi, dengan menggunakan 6 kriteria, tingkat akurasi *Fuzzy Naïve Bayes* sebesar 85%

## 4.5. Pengujian Sistem

Pengujian merupakan tahap yang utama dalam pembuatan suatu aplikasi perangkat lunak. Hasil pengujian yang didapat, akan dijadikan sebagai tolak ukur dalam proses pengembangan selanjutnya.

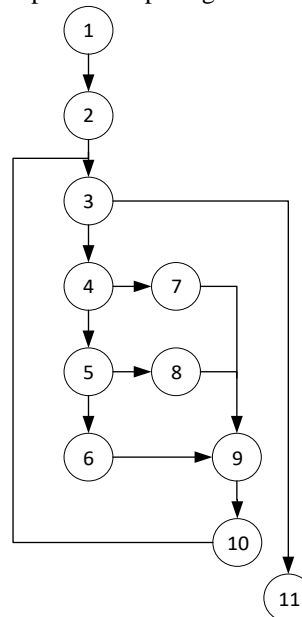
Pada pengujian ini dilakukan dengan menggunakan teknik *White box*, dimana terlebih dahulu memetakan *Flowchart* ke dalam *flowgraph* kemudian menghitung besarnya jumlah *edge* dan *node* dimana jumlah *node* dan *edge* ini akan menentukan besarnya *cyclomatic complexity*. Adapun *Flowchart* dan *flowgraph* sistem yang dibangun. Metode yang digunakan pada pengujian ini yaitu Metode logika *Fuzzy*.

*Flowchart* logika *Fuzzy* dapat dilihat pada gambar 4.10.



Gambar 4.10 Flowchart logika Fuzzy

Pada *Flowgraph* logika *Fuzzy* ini diambil dari *Flowchart* pada logika *Fuzzy*. Adapun *Flowgraph* dari logika *Fuzzy* dapat dilihat pada gambar 4.11.



Gambar 4.11 Flowgraph Naïve Bayes

Keterangan :

Node (N) : 13

Edge : 11

Predikat (P) : 3

*Cyclomatic Complexity*

$$\begin{aligned} V(G) &= (E - N) + 2 \\ &= (11 - 13) + 2 \\ &= -2 + 2 \\ &= 0 \end{aligned}$$

$$\begin{aligned} V(G) &= P + 1 \\ &= 3 + 1 \\ &= 4 \end{aligned}$$

*Independent path* :

Path 1 = 1 - 2 - 3 - 4 - 7 - 9 - 10 - 3 - 11

Path 2 = 1 - 2 - 3 - 4 - 5 - 8 - 9 - 10 - 3 - 11

Path 3 = 1 - 2 - 3 - 4 - 5 - 6 - 9 - 10 - 3 - 11

Path 4 = 1 - 2 - 3 - 11

Berdasarkan pada hasil yang didapatkan *Cyclomatic Complexity*, dan *Independent Path* bernilai 4, maka dapat diambil kesimpulan bahwa pengujian *White box* berhasil.

## 5. Kesimpulan

1. Perbandingan antara *Naïve Bayes* dengan *Fuzzy Naïve Bayes* terhadap penetapan UKT yang telah dilakukan oleh tim UKT Universitas Khairun pada program studi Teknik Informatika dapat dilakukan
2. Membandingkan akurasi dengan cara melakukan beberapa percobaan yaitu mencari akurasi *Naïve Bayes* dan *Fuzzy Naïve Bayes* dengan menggunakan 4 kriteria dan percobaan berikutnya mencari akurasi *Naïve Bayes* dan *Fuzzy Naïve Bayes* dengan menggunakan 6 kriteria, saat

- mencari akurasi, *data training* yang digunakan sebanyak 50 dan *data testing* sebanyak 20. Dengan menggunakan 4 kriteria, akurasi *Naïve Bayes* sebesar 70% dan akurasi *Fuzzy Naïve Bayes* sebesar 55%, sedangkan dengan menggunakan 6 kriteria, akurasi *Naïve Bayes* sebesar 90% dan akurasi *Fuzzy Naïve Bayes* sebesar 85%.
3. Pembuatan aplikasi penentuan status UKT dengan menggunakan metode *Fuzzy* dan *Naïve Bayes* dimulai dengan menganalisis kebutuhan sistem yaitu dilakukan dengan meminta data UKT di tim UKT Unkhair dan studi pustaka, kemudian mendesain sistem yaitu yang dilakukan dengan pemodelan diagram alir (Flowchart). Kedua, metode dan perancangan database dengan menggunakan *entity relationship diagram* (ERD) serta perancangan *interfaces* atau antar muka. Setelah tahapan perancangan dilakukan, dimulai dengan tahapan implementasi sistem yaitu *coding* dan *testing*.
  4. Banyaknya kriteria mempengaruhi tingkat akurasi.

#### Saran

1. Penelitian ini dapat dikembangkan dengan mengkombinasikan metode *Naïve Bayes* dengan metode *Fuzzy* lainnya.
2. Penormalisasian data menggunakan *Fuzzy* menggunakan pendekatan fungsi representasi lainnya.
3. Sebaiknya jumlah data dan jumlah kriteria ditambah, sehingga dapat diperoleh hasil akurasi yang lebih baik lagi.

#### Daftar Pustaka

1. Permenristekdikti, "Biaya Kuliah Tunggal Dan Uang Kuliah Tunggal Pada Perguruan Tinggi Negeri Di Lingkungan Kementerian Riset, Teknologi, Dan Pendidikan Tinggi," 2015. [Online]. Available: <http://kelembagaan.ristekdikti.go.id/wp-content/uploads/2016/08/Permenristekdikti22-2015BKT-UKT-PTN>.
2. A. Saelan, "Logika *Fuzzy*," no. 13508029, pp. 1–5, 2009.
3. L. Zadeh, "*Fuzzy sets*." 1964.
4. S. Kusumadewi, "Klasifikasi Status Gizi Menggunakan *Naïve Bayesian Classification*," *J. Univ. Binus*, vol. 3, no. 1, pp. 6–11, 2009.
5. A. A. Muin, "Metode *Naïve Bayes* Untuk Prediksi Kelulusan ( Studi Kasus : Data Mahasiswa Baru Perguruan Tinggi )," vol. 2, no. 1, 2016.