

IMPLEMENTASI METODE FUZZY K-NEAREST NEIGHBOR UNTUK KLASIFIKASI JENIS KANKER BERDASARKAN SUSUNAN PROTEIN

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun oleh:
Tahtri Nadia Utami
NIM: 145150207111079



**PROGRAM STUDI TEKNIK INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS BRAWIJAYA
MALANG
2018**

PENGESAHAN

Implementasi Metode *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

SKRIPSI

Untuk memenuhi sebagian persyaratan
memperoleh gelar Sarjana Komputer

Disusun Oleh :
Tahtri Nadia Utami
NIM: 145150207111079

Skripsi ini telah diuji dan dinyatakan lulus pada:
2 Agustus 2018

Telah diperiksa dan disetujui oleh:

Dosen Pembimbing I

Drs. Marji, M.T

NIP: 19670801 199203 1 001

Dosen Pembimbing II

Laili Muflikhah, S.Kom, M.Sc

NIP: 19741113 200501 2 001

Mengetahui

Ketua Jurusan Teknik Informatika



Tri Astoto Kurniawan, S.T, M.T, Ph.D

NIP: 19710518 200312 1 001

PERNYATAAN ORISINALITAS

Saya menyatakan dengan sebenar-benarnya bahwa sepanjang pengetahuan saya, di dalam naskah skripsi ini tidak terdapat karya ilmiah yang pernah diajukan oleh orang lain untuk memperoleh gelar akademik di suatu perguruan tinggi, dan tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis disitasi dalam naskah ini dan disebutkan dalam daftar pustaka.

Apabila ternyata didalam naskah skripsi ini dapat dibuktikan terdapat unsur-unsur plagiasi, saya bersedia skripsi ini digugurkan dan gelar akademik yang telah saya peroleh (sarjana) dibatalkan, serta diproses sesuai dengan peraturan perundang-undangan yang berlaku (UU No. 20 Tahun 2003, Pasal 25 ayat 2 dan Pasal 70).

Malang, 19 Juli 2018



Tahtri Nadia Utami

NIM: 145150207111079

KATA PENGANTAR

Puji syukur Penulis panjatkan kepada Allah SWT atas limpahan rahmat, hidayah, inayah dan kesempatan yang telah diberikan kepada Penulis, sehingga Penulis dapat menyelesaikan penelitian dengan judul “Implementasi Metode Fuzzy *k-Nearest Neighbor* (Fk-NN) untuk klasifikasi jenis kanker berdasarkan susunan protein”. Skripsi ini diajukan untuk memperoleh gelar Sarjana Komputer di Fakultas Ilmu Komputer (FILKOM) Universitas Brawijaya.

Dalam melakukan penelitian dan penyusunan dokumen skripsi ini, Penulis banyak sekali mendapatkan panduan, dukungan, dan arahan dari berbagai pihak. Dengan tulisan ini Penulis ingin menyampaikan terimakasih yang sebesar-besarnya kepada :

1. Bapak Drs. Marji, M.T selaku dosen pembimbing satu yang telah meluangkan waktu serta memberikan arahan dan bimbingan dalam pengerjaan skripsi ini.
2. Ibu Lailil Muflikhah, S.Kom, M.Sc selaku dosen pembimbing dua yang telah meluangkan waktu serta memberikan arahan dan bimbingan dalam pengerjaan skripsi ini.
3. Bapak Wayan Firdaus Mahmudy, S.Si, M.T, Ph.D selaku Dekan Fakultas Ilmu Komputer Universitas Brawijaya.
4. Bapak Tri Astoto Kurniawan, S.T, M.T, Ph.D selaku Ketua Jurusan Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
5. Bapak Agus Wahyu Widodo, S.T, M.Cs selaku Ketua Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Brawijaya.
6. Seluruh Dosen Fakultas Ilmu Komputer yang telah bersedia membagi ilmunya dan mendidik Penulis.
7. Bapak dan Mama, serta seluruh keluarga yang selalu mendukung dan mendoakan Penulis demi kemudahan dan kelancaraan pengerjaan skripsi.
8. Teman-teman penghuni Kos Bidadari 14B yang telah mendukung dan mendoakan penulis demi kelancaran pengerjaan skripsi.
9. Fakhruddin, Ilham, Hamim, Aghni, Mauldy, Aldy, Pratama serta teman-teman Kopma Squad yang telah berjuang bersama, saling mendukung dan mendoakan demi kelancaran pengerjaan skripsi.
10. Untuk seluruh teman-teman yang tidak dapat penulis sebutkan satu-persatu, yang telah membantu dan mendukung serta mendoakan penulis dalam pengerjaan skripsi.

Penulis sadar bahwasanya skripsi ini memiliki banyak kekurangan dan jauh dari kata sempurna. Penulis mengharapkan dan menerima kritik maupun saran agar skripsi ini menjadi lebih baik. Akhir kata, Penulis berharap agar skripsi ini dapat bermanfaat untuk seluruh pembacanya dan untuk penelitian-penelitian selanjutnya.

Malang, 19 Juli 2018

Penulis

tahtrinadiautami@gmail.com



ABSTRAK

Tahtri Nadia Utami, 2018. Implementasi Metode *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk klasifikasi jenis kanker berdasarkan susunan protein

Fakultas Ilmu Komputer Universitas Brawijaya. Pembimbing : Drs, Marji, M.T dan Lailil Muflikhah, S.Kom, M.Sc

Penyakit kanker merupakan penyakit paling mematikan setelah penyakit jantung. Penyebab umum terjadinya kanker adalah mutasi gen yang terjadi pada protein 53 yang berfungsi untuk mengendalikan replikasi DNA sebagai pengatur fungsi sel sehingga menyebabkan kesalahan susunan protein. Susunan protein dijadikan dasar dalam mengklasifikasi jenis kanker yang kemudian dapat membantu menentukan penanganan atau terapi yang tepat. Klasifikasi jenis kanker dilakukan dengan menggunakan metode *Fuzzy k-Nearest Neighbor* (Fk-NN). Data yang digunakan merupakan 752 data sekuens protein dengan panjang sekuens yaitu 393. Kelas klasifikasi meliputi kelas bukan kanker, kelas kanker payudara, kelas kanker usus dan kelas kanker paru-paru. Metode Fk-NN menghitung derajat keanggotaan masing-masing kelas pada k jarak terkecil dari hasil perhitungan jarak metode *k-Nearest Neighbor* (k-NN). Rata-rata tingkat akurasi tertinggi yaitu 52.56% dari hasil pengujian menggunakan *k-fold-validation*. Nilai k optimal metode Fk-NN adalah $k = 5$ dengan rata-rata tingkat akurasi sebesar 54.99%. Variasi jumlah data latih yang besar yaitu 90% dari jumlah dataset menghasilkan tingkat akurasi tertinggi sebesar 55.33%.

Kata kunci: Klasifikasi, Kanker, Susunan Protein, *Fuzzy k-Nearest Neighbor*.

ABSTRACT

Tahtri Nadia Utami, 2018. Implementation of Fuzzy k-Nearest Neighbor (Fk-NN) for cancer types classification based on protein sequence

Faculty of Computer Science, Brawijaya University. Advisor : Drs. Marji, M.T dan Lailil Muflikhah, S.Kom, M.Sc

Cancer is the most deadly disease besides heart disease. A common cause of cancer is gene mutation in protein 53 that serves to control the replication of DNA as a regulator of the cell function resulting in the wrong protein sequence. The protein sequences is used as a basis to classifying the types of cancer and then it can ease in determining the right handling or therapeutics method. The classification of cancer using the Fuzzy k-Nearest Neighbor (Fk-NN) method. The data used are 752 protein sequences with 393 sequence length on every sequence. The classification class includes non-cancer, breast cancer, colorectal cancer and lung cancer. The Fk-NN method calculates the degree of membership of each class at the k smallest distances generated from k-Nearest Neighbor method. The highest average accuracy rate is 52.56% of the test results using k-fold-validation. The optimal k value of the Fk-NN method is k = 5 with the average accuracy rate of 54.99%. The large variation in the amount of training data that is 90% of the dataset results in the highest accuracy rate of 55.33%.

Keywords: *classification, cancer, protein sequence, Fuzzy k-Nearest Neighbor.*

DAFTAR ISI

PENGESAHAN	ii
PERNYATAAN ORISINALITAS	iii
KATA PENGANTAR.....	iv
ABSTRAK.....	vi
ABSTRACT	vii
DAFTAR ISI	viii
DAFTAR TABEL.....	xi
DAFTAR GAMBAR.....	xii
DAFTAR KODE PROGRAM	xiii
BAB 1 PENDAHULUAN.....	1
1.1 Latar belakang.....	1
1.2 Rumusan masalah.....	3
1.3 Tujuan	3
1.4 Manfaat.....	4
1.5 Batasan masalah	4
1.6 Sistematika pembahasan.....	4
BAB 2 LANDASAN KEPUSTAKAAN	6
2.1 Kajian Pustaka	6
2.2 Kanker	7
2.2.1 Protein 53 (p-53)	8
2.2.2 Jenis Kanker.....	8
2.3 Protein.....	9
2.3.1 Sintesis Protein.....	10
2.3.2 Kode Genetik.....	10
2.3.3 <i>Point Accepted Mutation</i> (PAM)	11
2.4 <i>Data Mining</i>	12
2.4.1 Klasifikasi.....	13
2.4.2 <i>K-Nearest Neighbor</i> (k-NN)	13
2.5 <i>Fuzzy k-Nearest Neighbor</i> (Fk-NN).....	14

2.6 Akurasi	15
BAB 3 METODOLOGI	16
3.1 Studi Literatur	16
3.2 Tipe Penelitian	16
3.3 Strategi Penelitian.....	16
3.4 Pengumpulan Data	17
3.5 Perancangan	17
3.6 Implementasi	18
3.7 Pengujian	18
3.8 Penarikan Kesimpulan dan Saran	18
BAB 4 perancangan	19
4.1 Perancangan Algoritma	19
4.1.1 Diagram Alir Fk-NN.....	19
4.1.2 Perhitungan Manual.....	27
4.2 Perancangan Antarmuka	32
4.3 Perancangan Pengujian	33
BAB 5 IMPLEMENTASI	35
5.1 Implementasi Kode Program	35
5.1.1 Kode program <i>preprocessing</i>	35
5.1.2 Kode program <i>k-Nearest Neighbor</i> (k-NN)	39
5.1.3 Kode program Fuzzy k-Nearest Neighbor (Fk-NN).....	41
5.1.4 Implementasi perhitungan tingkat akurasi.....	42
5.2 Implementasi antarmuka.....	43
5.2.1 Memasukkan nilai <i>k</i> dan <i>fold</i>	44
5.2.2 Menampilkan data-data.....	44
5.2.3 Menampilkan hasil perhitungan	48
5.2.4 Menampilkan tingkat akurasi.....	52
BAB 6 PEMBAHASAN DAN ANALISIS	53
6.1 Pengujian nilai <i>k</i> pada <i>8-fold-validation</i>	53
6.2 Pengujian pengaruh variasi jumlah data latih	54
6.3 Analisis pengaruh nilai <i>k</i> pada <i>8-fold-validation</i>	55
6.4 Analisis pengaruh variasi jumlah data latih	56

BAB 7 Penutup	58
7.1 Kesimpulan.....	58
7.2 Saran	58
DAFTAR PUSTAKA.....	59



DAFTAR TABEL

Tabel 2.1 Presentase Jenis Kanker pada Pria dan Wanita	8
Tabel 2.2 Kode Genetik	10
Tabel 2.3 Kode Genetik (Lanjutan).....	11
Tabel 2.4 Matriks PAM250	12
Tabel 3.1 Contoh Dataset.....	17
Tabel 4.1 Data Wild	27
Tabel 4.2 Data latih	27
Tabel 4.3 Data uji	28
Tabel 4.4 Hasil konversi data latih	28
Tabel 4.5 Hasil konversi data uji	29
Tabel 4.6 Hasil perhitungan jarak	30
Tabel 4.7 Nilai jarak terdekat	30
Tabel 4.8 Derajat keanggotaan kelas NC	31
Tabel 4.9 Derajat keanggotaan kelas BC.....	31
Tabel 4.10 Derajat keanggotaan kelas CC.....	31
Tabel 4.11 Derajat keanggotaan kelas LC	32
Tabel 4.12 Perancangan pengujian pengaruh nilai k	33
Tabel 4.13 Perancangan pengujian pengaruh variasi data	34
Tabel 6.1 Tingkat akurasi pengaruh nilai k	53
Tabel 6.2 Tingkat akurasi pengaruh variasi jumlah data latih dan data uji	54

DAFTAR GAMBAR

Gambar 2.1 20 Asam Amino Pembentuk Protein	9
Gambar 2.2 Tahapan Sintesis Protein	10
Gambar 4.1 Diagram Alir Metode Fk-NN	20
Gambar 4.2 Diagram alir proses <i>preprocessing</i>	21
Gambar 4.3 Diagram alir proses menghitung jarak	22
Gambar 4.4 Diagram alir proses mengurutkan jarak terkecil.....	23
Gambar 4.5 Diagram alir proses mengambil k jarak terkecil.....	24
Gambar 4.6 Diagram alir proses Fk-NN	25
Gambar 4.7 Diagram alir proses menentukan kelas hasil klasifikasi	26
Gambar 4.8 Perancangan antarmuka	33
Gambar 5.1 Antarmuka <i>input</i> nilai k dan <i>fold</i>	44
Gambar 5.2 Tab data <i>wild</i>	45
Gambar 5.3 Tab data latih.....	46
Gambar 5.4 Tab Data Uji	46
Gambar 5.5 Tab kelas data latih	47
Gambar 5.6 Tab kelas data uji.....	47
Gambar 5.7 Tab normalisasi data latih	48
Gambar 5.8 Tab normalisasi data uji	49
Gambar 5.9 Tab hasil perhitungan jarak.....	49
Gambar 5.10 Tab urutan jarak	50
Gambar 5.11 Tab k jarak terdekat	50
Gambar 5.12 Tab derajat keanggotaan	51
Gambar 5.13 Tab hasil klasifikasi	51
Gambar 5.14 <i>TextField</i> tingkat akurasi	52
Gambar 6.1 Grafik tingkat akurasi nilai k tiap <i>fold</i>	55
Gambar 6.2 Grafik rata-rata tingkat akurasi nilai k	55
Gambar 6.3 Grafik rata-rata tingkat akurasi k - <i>fold</i>	56
Gambar 6.4 Grafik tingkat akurasi pengaruh variasi jumlah data latih	57

DAFTAR KODE PROGRAM

Kode Program 5.1 Proses <i>preprocessing</i>	35
Kode Program 5.2 Proses menghitung jarak data uji terhadap data latih	39
Kode Program 5.3 Proses mengurutkan jarak terkecil	40
Kode Program 5.4 Proses mengambil k jarak terkecil	40
Kode Program 5.5 Proses menghitung derajat keanggotaan	41
Kode Program 5.6 Proses perhitungan tingkat akurasi	42



BAB 1 PENDAHULUAN

1.1 Latar belakang

Selama ini, kanker menjadi penyakit yang paling ditakuti. Selain penyakit jantung, penyakit kanker juga merupakan penyakit penyebab kematian tertinggi. Berdasarkan data yang didapatkan Yayasan Kanker Indonesia dari Kementerian Kesehatan, kasus penyakit kanker di Indonesia mengalami peningkatan hingga 100.000 kasus pada tahun 2016. Pada tahun 2017 lalu, Kementerian Kesehatan memprediksi adanya peningkatan kematian manusia di seluruh dunia akibat kanker hingga 13 juta jiwa per tahun 2030 mendatang. Berdasarkan data dari Riset Kesehatan Dasar (Riskesdas) 2013 seperti yang dilansir pada situs <http://www.depkes.go.id>, jumlah populasi manusia yang mengidap kanker di Indonesia adalah sekitar 347.000 orang.

Kanker merupakan penyakit dengan ciri yaitu adanya sel yang sifat dan pertumbuhannya tidak wajar. Sel kanker dapat tumbuh dimanapun pada bagian tubuh manusia dan tidak memandang usia, walaupun beberapa fakta menyatakan bahwa kebanyakan penderita kanker berusia 40 tahun. Sampai saat ini, penanganan terhadap kasus kanker lebih dikembangkan untuk mendeteksi lebih awal penyakit kanker karena melihat begitu bahayanya penyakit kanker.

Mutasi gen merupakan penyebab umum terjadinya kanker. Mutasi gen disebabkan oleh adanya perubahan genetik dalam sel tubuh. Mutasi gen yang menyebabkan kanker salah satunya adalah yang terjadi pada protein 53 dimana protein tersebut memiliki fungsi sebagai pengendali replikasi DNA (Osterath, 2014). DNA merupakan pengatur fungsi sel di dalam tubuh manusia. Mutasi tersebut menyebabkan terjadinya kesalahan pada replikasi DNA sehingga terdapat kesalahan pada siklus pertumbuhan sel dan menyebabkan terjadinya kesalahan dalam susunan protein (Nugraha, 2009). Kesalahan pada siklus pertumbuhan sel tersebut menyebabkan terjadinya pertumbuhan sel yang tidak terkendali. Sel dapat tumbuh menjadi kanker apabila sel tersebut pecah dan menyebar ke sel-sel yang berada di dekatnya sehingga merusak fungsi organ tubuh lainnya. Biasanya secara fisik yang terlihat pada saat dilakukannya pengecekan misal dengan CT Scan, kanker akan terlihat membentuk benjolan pada organ tertentu yang terserang kanker.

Jenis kanker yang paling banyak diderita penderita kanker saat ini yaitu Kanker Paru-Paru (*Lung Cancer*), Kanker Payudara (*Breast Cancer*) dan Kanker Kolon (*Colorectal Cancer*). Hal tersebut berdasarkan data yang dilansir pada situs <http://p53.free.fr>. Mutasi gen p-53 dapat terjadi di bagian tubuh mana saja, dengan susunan protein yang diekspresikan berbeda satu dengan lainnya sehingga menghasilkan jenis kanker yang juga berbeda (Kurniati, et al., 2013). Berdasarkan hal tersebut, dapat diketahui bahwa susunan protein yang diekspresikan dari hasil siklus pembelahan sel dapat dipelajari sebagai dasar klasifikasi jenis kanker. Berdasarkan susunan protein, hasil klasifikasi yang

didapatkan akan membantu dalam menentukan metode terapi atau pengobatan yang tepat pada masing-masing jenis kanker tersebut.

Dalam penelitian ini, digunakan salah satu metode *Data Mining*. *Data Mining* merupakan ilmu yang digunakan untuk melakukan analisis dan ekstraksi data. Proses pada *Data Mining* yaitu untuk memperoleh dan mempelajari pola tertentu untuk didapatkan suatu informasi dari sekumpulan data (Pramadhani & Setiadi, 2014). Salah satu fungsi *Data Mining* adalah klasifikasi atau menentukan suatu data masuk ke dalam suatu kelas berdasarkan pola yang dipelajari dari data tertentu. Dalam bidang biologi maupun kedokteran fungsi klasifikasi sering digunakan untuk mendeteksi penyakit tertentu. Metode klasifikasi yang sering digunakan salah satunya yaitu *k-Nearest Neighbor* (k-NN).

Proses klasifikasi yang dilakukan dengan metode k-NN yaitu dengan cara menghitung jarak suatu data yang belum memiliki label kelas atau kategori terhadap sekumpulan data lainnya yang telah memiliki kelas atau kategori. Setelah itu akan diperoleh jarak terdekat data terhadap k data lainnya. Metode k-NN dipilih pada beberapa kasus klasifikasi dikarenakan metode ini tergolong mudah digunakan. Penelitian tentang klasifikasi jenis kanker berdasarkan struktur protein pernah dilakukan oleh Arintha Retwintasari pada tahun 2016 dengan menggunakan variasi dari metode k-NN yaitu *Modified k-Nearest Neighbor* (MKNN). Penelitian tersebut menghasilkan nilai akurasi rata-rata maksimum sebesar 43.53% pada dataset sejumlah 150 data dan nilai k sebesar 1. Penelitian terbaru dengan permasalahan yang sama juga dilakukan oleh Tawang Wulandari pada tahun 2018 dengan mengimplementasi metode *Naïve Bayes*. Nilai akurasi rata-rata maksimum yang dihasilkan adalah 67.17% pada dataset sejumlah 848 data.

Untuk menghasilkan nilai akurasi yang lebih baik dari penelitian sebelumnya, dapat dilakukan penggabungan teknik *Fuzzy* dengan metode klasifikasi k-NN yang biasa dikenal dengan metode *Fuzzy k-Nearest Neighbor* (Fk-NN). Menggabungkan teknik *Fuzzy* ke dalam metode k-NN dapat mempermudah proses klasifikasi karena penentuan kelas data akan lebih tegas dan jelas atau tidak ambigu. Dengan menggunakan teknik *Fuzzy*, sebuah sampel tidak dimasukkan ke dalam kelas tertentu melainkan diberi nilai derajat keanggotaan kelasnya sehingga hasil penentuan kelas lebih terjamin ketepatannya (Nugraha, et al., 2017). Konsep metode Fk-NN yaitu kelas terklasifikasi dimasukkan ke dalam pola tertentu bukan memasukkan pola ke dalam kelas tertentu (Keller, et al., 1985). Tingkat akurasi yang dihasilkan metode Fk-NN dinyatakan lebih besar daripada metode k-NN serta *Support Vector Machine* (SVM) pada penelitian yang pernah dilakukan sebelumnya. Nilai akurasi yang dihasilkan metode k-NN dan SVM tersebut masing-masing sebesar 58.23% dan 58.45% sedangkan nilai akurasi yang dihasilkan metode Fk-NN yaitu sebesar 64.12% (Zhang, et al., 2009).

Penelitian sebelumnya yang telah mengimplementasikan metode Fk-NN yaitu penelitian yang dilakukan oleh Dyah Puspitasari pada tahun 2015 untuk pemodelan sistem pakar diagnosa penyakit tanaman cabai merah. Penelitian tersebut menghasilkan tingkat akurasi rata-rata sebesar 96.12% yang mana lebih

besar dibandingkan dengan tingkat akurasi yang dihasilkan pada penelitian sebelumnya menggunakan metode *Modified k-Nearest Neighbor* (Mk-NN) yang menghasilkan tingkat akurasi sebesar 94.29%. Penelitian lain yang juga menggunakan metode Fk-NN dilakukan oleh Rofika pada tahun 2015 untuk sistem pakar diagnosa penyakit kulit pada anak. Dari hasil penelitian tersebut, diketahui bahwa tingkat akurasi yang dihasilkan rata-rata sebesar 93.33%. Pada penelitian tersebut dilakukan pengujian dengan membandingkan metode Fk-NN dengan metode k-NN biasa. Tingkat akurasi yang dihasilkan metode k-NN lebih kecil yaitu 83.33% sehingga disimpulkan bahwa metode Fk-NN lebih akurat dibandingkan metode k-NN pada permasalahan tersebut. Tingkat akurasi Fk-NN juga lebih besar dibandingkan tingkat akurasi yang dihasilkan pada penelitian sebelumnya dengan objek permasalahan yang sama menggunakan metode *Naive Bayes* (85.4%) dan *Dempster-Shafer* (86.67%).

Terdapat salah satu penelitian implementasi metode Fk-NN sebelumnya yang juga menghasilkan tingkat akurasi yang lebih besar dibandingkan dengan tingkat akurasi yang dihasilkan metode *Naive Bayes*. Penelitian tersebut dilakukan oleh Dwi Citra Annisa dengan permasalahan diagnosa awal penyakit DBD, Malaria dan Tifoid pada tahun 2016 dan menghasilkan tingkat akurasi sebesar 87%. Sedangkan tingkat akurasi yang dihasilkan metode *Naive Bayes* hanya sebesar 43%.

Berdasarkan uraian di atas, maka judul untuk penelitian ini yaitu ***"IMPLEMENTASI FUZZY K-NEAREST NEIGHBOR (FK-NN) UNTUK KLASIFIKASI JENIS KANKER BERDASARKAN SUSUNAN PROTEIN"***. Penelitian ini diharapkan mampu dalam penentuan jenis kanker berdasarkan susunan protein secara lebih mudah dan lebih baik yang nantinya dapat membantu menentukan metode terapi atau pengobatan yang tepat dalam penanganan penyakit kanker tersebut.

1.2 Rumusan masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka dapat dirumuskan beberapa rumusan masalah sebagai berikut :

1. Bagaimana mengimplementasikan metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein?
2. Berapa nilai k terbaik untuk metode Fk-NN dalam klasifikasi jenis kanker berdasarkan susunan protein?
3. Bagaimana pengaruh variasi data latih terhadap tingkat akurasi?

1.3 Tujuan

Tujuan dari penelitian ini adalah sebagai berikut :

1. Menerapkan metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein
2. Mengetahui nilai k terbaik untuk metode Fk-NN dalam klasifikasi jenis kanker berdasarkan susunan protein

3. Mengetahui pengaruh variasi data latih terhadap tingkat akurasi

1.4 Manfaat

Manfaat dilaksanakannya penelitian ini yaitu dapat diketahui keakuratan metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein. Hasil penelitian ini dapat digunakan untuk membantu menentukan metode terapi atau pengobatan yang tepat.

1.5 Batasan masalah

Batasan masalah yang digunakan untuk implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein adalah sebagai berikut :

1. Objek yang digunakan sebagai dasar dalam klasifikasi penyakit kanker yaitu susunan protein dalam tubuh manusia.
2. Klasifikasi jenis kanker hanya dilakukan pada 3 jenis kanker yaitu kanker paru-paru, kanker kolon dan kanker payudara.
3. Metode yang akan diimplementasikan adalah metode klasifikasi *Fuzzy k-Nearest Neighbor*.
4. Data yang digunakan dalam melakukan pengujian hasil implementasi metode pada penelitian ini merupakan data yang didapatkan dari <http://www.uniprot.org> sebanyak 752 baris susunan protein dengan panjang sekuens masing-masing data adalah 393.

1.6 Sistematika pembahasan

Penyusunan laporan penelitian pada implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein ini disesuaikan dengan sistematika pembahasan berikut :

BAB 1 PENDAHULUAN

Bab ini berisi latar belakang adanya penelitian tentang implementasi metode Fk-NN untuk mengklasifikasi jenis kanker berdasarkan protein, rumusan masalah yang digunakan, tujuan, manfaat yang diperoleh dari penelitian ini, batasan masalah serta sistematika pembahasan isi laporan penelitian.

BAB 2 LANDASAN KEPUSTAKAAN

Bab ini berisi penjelasan dari kajian pustaka dan dasar teori yang berkaitan dengan penyakit kanker, mutasi gen, fungsi klasifikasi *Data Mining* dan metode Fk-NN.

BAB 3 METODOLOGI

Bab ini berisi penjelasan langkah-langkah metode yang digunakan dalam melakukan implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein.

BAB 4 PERANCANGAN

Bab ini berisi uraian perancangan metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan, perancangan antarmuka dan perancangan pengujian.

BAB 5 IMPLEMENTASI

Bab ini berisi uraian implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein meliputi implementasi kode program dan implementasi antarmuka berdasarkan perancangan yang dilakukan.

BAB 6 PEMBAHASAN DAN ANALISIS

Bab ini berisi pembahasan berupa tingkat akurasi hasil pengujian yang dilakukan terhadap hasil implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein. Bab ini juga berisi analisis terhadap hasil pengujian yang telah dilakukan.

BAB 7 PENUTUP

Bab ini berisi kesimpulan dari hasil implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein serta saran yang dapat digunakan sebagai acuan pada penelitian selanjutnya.

BAB 2 LANDASAN KEPUSTAKAAN

2.1 Kajian Pustaka

Penelitian klasifikasi jenis kanker berdasarkan susunan protein dilakukan oleh Arintha Retwitasari pada tahun 2016 dengan judul “Penentuan Jenis Kanker Berdasarkan Struktur Protein Menggunakan Algoritma *Modified k-Nearest Neighbor* (MKNN)”. Dengan mengimplementasikan metode *Modified k-Nearest Neighbor* tersebut, didapatkan nilai rata-rata akurasi pada pengujian 150 dataset yaitu sebesar 43.53%. Dari hasil penelitian, disimpulkan bahwa jumlah dataset berpengaruh terhadap tingkat akurasi. Tingkat akurasi akan semakin kecil jika nilai k semakin besar. Ketidakseimbangan kelas pada dataset mempengaruhi hasil pengujian nilai k tersebut (Retwitasari, et al., 2016).

Penelitian lainnya berjudul “Klasifikasi Jenis Kanker Berdasarkan Struktur Protein Menggunakan Algoritma *Naive Bayes*” yang dilakukan tahun 2018. Dengan mengimplementasikan metode *Naive Bayes* tersebut, didapatkan nilai rata-rata akurasi pada pengujian 848 dataset sebesar 67.16%. Tingkat akurasi yang dihasilkan tidak terlalu dipengaruhi oleh jumlah dataset, melainkan dipengaruhi oleh sebaran data set atau variasi jumlah data latih dan data uji (Wulandari, 2018).

Tingkat akurasi yang dihasilkan metode Fk-NN dinyatakan lebih besar daripada metode k-NN serta *Support Vector Machine* (SVM) pada penelitian yang pernah dilakukan sebelumnya. Nilai akurasi yang dihasilkan metode k-NN dan SVM tersebut masing-masing sebesar 58.23% dan 58.45% sedangkan nilai akurasi yang dihasilkan metode Fk-NN yaitu sebesar 64.12% (Zhang, et al., 2009).

Penelitian sebelumnya yang mengimplementasikan metode Fk-NN yaitu penelitian yang berjudul “Pemodelan Sistem Pakar Diagnosa Penyakit Tanaman Cabai Merah dengan Metode *Fuzzy k-Nearest Neighbor* (Fk-NN)” yang menghasilkan tingkat akurasi sebesar 96.12% (Puspitasari, et al., 2015). Tingkat akurasi tersebut lebih besar dibandingkan tingkat akurasi yang dihasilkan penelitian dengan objek permasalahan yang sama berjudul “Diagnosa Penyakit Tanaman Cabai Menggunakan Metode *Modified k-Nearest Neighbor* (Mk-NN)” yaitu sebesar 94.29% (Wiraga, et al., 2015).

Penelitian lainnya yang mengimpementasikan metode Fk-NN berjudul “Sistem Pakar Diagnosa Penyakit Kulit pada Anak Menggunakan Metode *Fuzzy k-Nearest Neighbor*”. Tingkat akurasi yang dihasilkan yaitu sebesar 93.33%. Pada penelitian tersebut dilakukan pengujian dengan membandingkan metode Fk-NN dengan metode k-NN biasa. Tingkat akurasi yang dihasilkan metode k-NN lebih kecil yaitu 83.33% sehingga disimpulkan bahwa metode Fk-NN lebih akurat dibandingkan metode k-NN pada permasalahan tersebut. Penelitian sebelumnya dengan permasalahan yang sama juga dijadikan pembanding metode Fk-NN berdasarkan tingkat akurasinya (Soebroto, et al., 2015). Salah satunya yaitu penelitian yang berjudul “Sistem Pakar Diagnosa Penyakit Kulit pada Anak Menggunakan Metode *Naive Bayes*” menunjukkan hasil tingkat akurasi lebih

kecil dibandingkan dengan menggunakan metode Fk-NN yaitu sebesar 85.4% (Srimuddawamah, et al., 2015).

Penelitian lainnya berjudul “Sistem Pakar Diagnosa Penyakit Kulit pada Anak Menggunakan Metode *Dempster-Shafer*” yang juga menghasilkan tingkat akurasi lebih kecil dibandingkan dengan menggunakan metode Fk-NN yaitu sebesar 86.67% (Candrarini, et al., 2015).

Penelitian lain berjudul “Pendeteksi Jenis *Attention Deficit Hyperactivity Disorder* (ADHD) pada Anak Usia Dini Menggunakan Metode *Fuzzy k-Nearest Neighbor*” juga menghasilkan tingkat akurasi yang lebih besar dibandingkan dengan penelitian menggunakan metode *Naive Bayes* yang berjudul “Rancang Bangun Aplikasi Sistem Pakar Diagnosis ADHD (*Attention Deficit Hyperactivity Disorder*) Menggunakan Metode *Certainy Vector*”. Tingkat akurasi yang dihasilkan dengan menggunakan metode Fk-NN yaitu sebesar 90%. Sedangkan tingkat akurasi yang dihasilkan dengan menggunakan metode *Naive Bayes* hanya 84.62% (Laxsmana, et al., 2015).

2.2 Kanker

Kanker adalah penyakit yang dapat menyerang setiap organ manusia dengan ciri utamanya yaitu adanya pembelahan sel yang terus menerus. Gaya hidup manusia yang tidak benar menjadi faktor yang paling berpengaruh terhadap tumbuhnya kanker. Kanker bukanlah suatu penyakit, melainkan kumpulan penyakit. Karena, kanker membelah diri menjadi sel jahat yang dapat terus tumbuh (King, 2000). Ukuran sel abnormal yang disebut kanker merupakan ukuran tingkat keganasan kanker. Karena kanker sangat berpengaruh pada terganggunya sistem pertahanan tubuh, maka kanker juga dapat menimbulkan beberapa gejala. Namun, kebanyakan orang tidak langsung sadar bahwa itu merupakan gejala penyakit kanker (Kelvin & Tyson, 2011). Sel kanker yang tumbuh dapat mempengaruhi jaringan tubuh yang ada di sekitarnya dan sifatnya merusak. Kanker juga dapat menyebar dan tumbuh di bagian tubuh lainnya bahkan yang jauh dari tempat awalnya (Setiati, 2009).

Istilah kanker disebutkan jika suatu sel tumbuh menjadi sel yang tidak normal. Sel menjadi tidak normal dikarenakan adanya kesalahan di dalam siklus pembelahan sel. Pada siklus pembelahan sel tersebut, terdapat mutasi gen pada DNA sel yang menghasilkan gen abnormal di dalam sel padahal gen tersebut merupakan gen pengendali. Salah satu sifat alami sel yaitu sel akan menyusun protein yang tidak dikenali sistem imun sebagai protein yang cacat sehingga sel tersebut dapat dibunuh oleh sistem imun. Namun, beberapa sel yang tidak normal memiliki sifat jahat sehingga sel tersebut mengelabui sistem imun dengan tidak menunjukkan adanya susunan protein asing sebagai tanda bahwa sel merupakan sel yang tidak normal. Sel yang jahat itu melakukan pembelahan yang tidak terkendali agar kebutuhan hidupnya terus terpenuhi. Pembelahan yang terjadi terus menerus menyebabkan ukuran sel terus bertambah (Sudiana, 2008).

2.2.1 Protein 53 (p-53)

Protein 53 salah satu protein dari kelompok *Tumor Suppressor Protein* yang merupakan hasil pengkodean genetic gen p-53 (P53). Gen p-53 bekerja dalam sel tubuh untuk mengatur ekspresi protein yang dihasilkan dari siklus pembelahan sel. Dalam siklus pembelahan sel bisa saja terjadi kesalahan dan menumbuhkan sel yang tidak normal. Protein 53 dan protein lainnya yang diekspresikan gen p-53 dibutuhkan untuk mengatur pertumbuhan dan pembelahan sel tersebut. Namun, jika mutasi terjadi pada gen p-53, maka protein yang diekspresikan mempengaruhi faktor penghambat pertumbuhan atau pembelahan sel sehingga sel yang tidak normal tersebut akan terus tumbuh secara tidak terkendali dan dapat diistilahkan sebagai sel kanker (Sudiana, 2008).

2.2.2 Jenis Kanker

Kanker memiliki jenis yang beragam. Biasanya jenis kanker dibedakan berdasarkan tempat tumbuhnya kanker. Tiga jenis kanker yang memiliki presentase jumlah pengidap tertinggi yaitu kanker payudara, kanker paru-paru dan kanker kolon (King, 2000). Kanker payudara hanya akan dialami oleh wanita sedangkan kanker paru-paru dan kanker kolon bisa dialami oleh siapapun, baik laki-laki maupun perempuan. Tumbuhnya kanker payudara sangat dipengaruhi oleh Hormon Estrogen. Gejala fisik ditunjukkan dengan beberapa tanda seperti adanya benjolan keras dan kerutan di sekitar payudara (Setiati, 2009). Kanker usus terjadi di sekitar usus besar (King, 2000) dan kanker paru-paru tumbuh di sekitar saluran pernapasan sekitar paru-paru (Klamerus, et al., 2012).

Tabel 2.1 Presentase Jenis Kanker pada Pria dan Wanita

Jenis	Pria		Wanita	
	Teridentifikasi	Kematian	Teridentifikasi	Kematian
Payudara	<1	<1	32	18
Paru-Paru	14	33	13	24
Kolon	10	9	12	11
Mulut	3	2	2	1
Kulit	3	1	3	1
Pankreas	2	7	2	5
Leukimia	7	8	6	8
Prostat	36	14	-	-
Ovarium	-	-	5	6
Serviks	-	-	9	4

Sumber : (King, 2000)

2.3 Protein

Salah satu yang merupakan bagian penting dalam tubuh makhluk hidup adalah protein. Protein adalah bahan yang digunakan untuk membantu proses metabolisme serta dalam melindungi tubuh dari faktor penyerang. Protein disusun oleh asam amino yang tergabung dari hasil pengkodean genetik (Sudiana, 2008). Suatu susunan protein dapat terbentuk dari banyak asam amino bahkan mencapai 20 jenis asam amino (Retwitasari, et al., 2016).

Name	Symbol	3-Letter	R group	Hydrophobicity
Aspartic	Asp	D	$\text{O}=\text{C}-\text{CH}_2-\text{COO}^-$	-3.5
Glutamate	Glu	E	$\text{O}=\text{C}-\text{CH}_2-\text{CH}_2-\text{COO}^-$	-3.5
Lysine	Lys	K	$\text{H}_2\text{N}^+-\text{CH}_2-\text{CH}_2-\text{CH}_2-\text{CH}_2-\text{COO}^-$	-3.9
Arginine	Arg	R	$\text{H}_2\text{N}-\text{C}(\text{NH}_2)=\text{NH}-\text{CH}_2-\text{CH}_2-\text{COO}^-$	-4.5
Histidine	His	H	$\text{HC}=\text{C}(\text{NH}_2)-\text{CH}_2-\text{COO}^-$	-3.2
Tyrosine	Tyr	Y	$\text{HO}-\text{C}_6\text{H}_4-\text{CH}_2-\text{COO}^-$	-1.3
Tryptophan	Trp	W	$\text{C}_6\text{H}_5-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	-0.9
Phenylalanine	Phe	F	$\text{C}_6\text{H}_5-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	2.8
Cysteine	Cys	C	$\text{HS}-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	2.3
Methionine	Met	M	$\text{CH}_3-\text{S}-\text{CH}_2-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	1.9
Serine	Ser	S	$\text{HO}-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	-0.8
Threonine	Thr	T	$\text{CH}_3-\text{CH}(\text{OH})-\text{CH}(\text{NH}_2)-\text{COO}^-$	-0.7
Asparagine	Asn	N	$\text{NH}_2-\text{C}(\text{O})-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	-3.5
Glutamine	Gln	Q	$\text{NH}_2-\text{C}(\text{O})-\text{CH}_2-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	-3.5
Glycine	Gly	G	$\text{H}-\text{CH}(\text{NH}_2)-\text{COO}^-$	-0.4
Alanine	Ala	A	$\text{CH}_3-\text{CH}(\text{NH}_2)-\text{COO}^-$	1.8
Valine	Val	V	$\text{CH}_3-\text{CH}(\text{CH}_3)-\text{CH}(\text{NH}_2)-\text{COO}^-$	4.2
Leucine	Leu	L	$\text{CH}_3-\text{CH}_2-\text{CH}(\text{CH}_3)-\text{CH}(\text{NH}_2)-\text{COO}^-$	3.8
Isoleucine	Ile	I	$\text{CH}_3-\text{CH}_2-\text{CH}(\text{CH}_3)-\text{CH}(\text{NH}_2)-\text{COO}^-$	4.5
Proline	Pro	P	$\text{CH}_2-\text{CH}_2-\text{CH}_2-\text{CH}(\text{NH}_2)-\text{COO}^-$	-1.6

Gambar 2.1 20 Asam Amino Pembentuk Protein

Sumber : (Sari, 2007)

2.3.1 Sintesis Protein

Terdapat dua tahap dalam proses sintesis protein. Tahap pertama yaitu transkripsi yang merupakan proses membentuk DNA menjadi *messenger Ribonucleic Acid* (mRNA) dalam inti sel dimana mRNA membawa pesan dari DNA. Kemudian, mRNA menuju ke ribosom untuk dilakukan sintesis asam amino. Setelah proses tersebut, terjadi tahapan yang kedua yaitu tahapan translasi dimana mRNA menjadi protein. Terbentuknya sistem *transfer Ribonucleic Acid* (tRNA) menghasilkan susunan kode genetik dari nukleotida. Asam amino kemudian didapatkan dari hasil menerjemahkan kode genetik tersebut. Terbentuklah protein dari hasil tahapan sistesis tersebut (Sudiana, 2008).



Gambar 2.2 Tahapan Sintesis Protein

Sumber : (Sudiana, 2008)

2.3.2 Kode Genetik

Kode genetik merupakan hasil pengelompokan susunan nukleotida dalam alur sintesis protein. Dibutuhkan 3 huruf susunan kode genetik yang disebut kodon dalam mengkode asam amino yang berasal dari huruf-huruf seperti C, G, T dan A. Salah satu kegunaan kode genetik di dalam protein yaitu mengendalikan alur sel (Retwitasari, et al., 2016). Kesamaan nama asam amino akan terjadi pada beberapa kode genetik, hanya saja dibedakan oleh susunan nukleotida, seperti misalnya (Sudiana, 2008) :

1. Kode genetik UAU dan UAC dimiliki oleh Tyronine (Tyr);
2. Kode genetik CCU, CCC, CCA, CCG dimiliki oleh Proline (Pro);
3. Kode genetik UCU, UCC, UCA, UCG, UGU, UGC dimiliki oleh Serine (Ser).

Tabel 2.2 Kode Genetik

Posisi	Posisi Kedua	Posisi Ketiga
--------	--------------	---------------

Pertama	U	C	A	G	
U	Phe	Ser	Tyr	Cys	U
	Phe	Ser	Tyr	Cys	C
	Leu	Ser	Stop	Stop	A
	Leu	Ser	Stop	Trp	G

Sumber : (Sudiana, 2008)

Tabel 2.3 Kode Genetik (Lanjutan)

Posisi Pertama	Posisi Kedua				Posisi Ketiga
	U	C	A	G	
C	Leu	Pro	His	Arg	U
	Leu	Pro	His	Arg	C
	Leu	Pro	Gln	Arg	A
	Leu	Pro	Gln	Arg	G
A	Ile	Thr	Asn	Ser	U
	Ile	Thr	Asn	Ser	C
	Ile	Thr	Lys	Arg	A
	Ile	Thr	Lys	Arg	G
G	Val	Ala	Asp	Gly	U
	Val	Ala	Asp	Gly	C
	Val	Ala	Glu	Gly	A
	Val	Ala	Asp	Gly	G

Sumber : (Sudiana, 2008)

2.3.3 Point Accepted Mutation (PAM)

Dalam pengkodean asam amino, terdapat penggantian asam amino tunggal dengan asam amino tunggal lainnya. Penggantian yang diterima oleh proses seleksi alam dapat disebut sebagai PAM. Terdapat matriks PAM yang merepresentasikan kemungkinan adanya penggantian asam amino tunggal.

Tabel 2.4 Matriks PAM250

	A	R	N	D	C	Q	E	G	H	I	L	K	M	F	P	S	T	W	Y	V
A	2	-2	0	0	-2	0	0	1	-1	0	-2	-1	-1	-3	1	1	1	-6	-4	0
R	-2	6	0	-1	-4	1	-1	-3	2	-2	-3	3	0	-4	0	0	-1	2	-5	-2
N	0	0	2	2	-3	1	2	1	2	-2	-3	1	-2	-3	0	1	1	-4	-2	-2
D	0	-1	2	4	-5	2	3	1	1	-2	-4	0	-3	-5	-1	0	0	-7	-4	-2
C	-2	-3	-4	-5	12	-5	-5	-4	-3	-3	-6	-5	-5	-4	-2	0	-2	-8	0	-2
Q	0	1	1	2	-5	4	2	-1	3	-2	-2	1	-1	-4	0	-1	-1	-5	-4	-2
E	0	-1	1	3	-5	2	4	0	1	-2	-3	0	-2	-5	0	0	0	-7	-4	-2
G	1	-3	0	1	-3	-1	0	5	-2	-2	-4	-2	-3	-5	0	1	0	-7	-5	-1
H	-1	1	1	1	-3	3	0	-3	6	-3	-3	0	-3	-2	0	-1	-1	-3	0	-3
I	-1	-2	-2	-2	-2	-2	-2	-3	-3	4	2	-2	2	1	-2	-1	0	-5	-1	4
L	-2	-3	-3	-4	-6	-2	-3	-4	-2	2	6	-2	4	2	-2	-3	-2	-2	-1	2
K	-1	3	1	0	-5	1	0	-2	0	-2	-3	5	0	-5	-1	0	0	-4	-5	-2
M	-1	-1	-2	-3	-5	-1	-2	-3	-2	2	4	1	6	0	-2	-2	0	-4	-3	2
F	-3	-4	-3	-5	-4	-4	-5	-4	-2	1	2	-5	0	9	-5	-3	-3	0	7	-1
P	1	0	0	-1	-3	0	0	0	0	-2	-2	-1	-2	-4	6	1	0	-6	-5	-1
S	2	1	2	1	1	0	1	2	0	-1	-2	1	-1	-2	2	2	2	-2	-2	0
T	0	-2	0	-1	-3	-2	-1	-1	-2	-1	-2	-1	-1	-4	0	1	2	-6	-4	0
W	-6	2	-5	-7	-7	-6	-7	-7	-5	-6	-7	-4	-6	1	-6	-2	-5	17	1	-8
Y	-3	-5	-2	-4	1	-4	-4	-5	0	-1	-1	-5	-2	7	-5	-3	-3	0	10	-2
V	0	-2	-2	-2	-2	-2	-2	-2	-2	4	2	-2	-2	-1	-1	-1	0	-6	-3	4

Sumber : (Rizby, 2018)

Kolom dan baris pada matriks tersebut mewakili 20 asam amino. Setiap sel matriks menggambarkan nilai kemungkinan asam amino tunggal pada suatu baris tertentu berubah menjadi asam amino tunggal pada suatu kolom tertentu. Matriks PAM meliputi matriks PAM1 – PAM250 yang berbeda dipengaruhi oleh adanya evolusi sekuens protein (Rizby, et al., 2018). Berikut merupakan tabel Matriks PAM250.

2.4 Data Mining

Data Mining merupakan proses penggalian data yang kemudian mempelajarinya menggunakan *machine learning* berbasis komputer untuk memperoleh informasi (Hermawati, 2013). *Data Mining* melakukan ekstraksi informasi dari sejumlah data yang besar. Salah satu tujuan *Data Mining* yang juga dikenal sebagai *Knowledge Discovery* dalam melakukan ekstraksi informasi yaitu untuk mendukung pengambilan keputusan. *Data Mining* membuat suatu model dari data yang dianalisis untuk melakukan beberapa teknik prediksi dan deskripsi. Teknik deskripsi dilakukan dengan mengenali pola data satu sama lain yang kemudian data dapat dikategorikan menjadi beberapa kelompok. Sedangkan pada teknik prediksi, pola data dipelajari agar dapat mengidentifikasi data baru (Prasetyo, 2012). Beberapa teknik *Data Mining* yang bersifat prediksi dan deskripsi antara lain (Hermawati, 2013) :

- Prediksi (*Predictive*)
 1. *Classification* (Klasifikasi)
 2. *Regression* (Regresi)
 3. *Deviation Detection* (Deteksi deviasi)
- Deskripsi (*Descriptive*)

1. *Clustering* (Pengelompokan)
2. *Association Rule Discovery* (Penentuan Kaidah Asosiasi)
3. *Sequential Pattern Discovery* (Pencarian Pola Sekuensial)

2.4.1 Klasifikasi

Klasifikasi merupakan teknik mengidentifikasi data yang baru masuk sebagai anggota suatu kelas tertentu. Dalam pengerjaan klasifikasi data, beberapa kelas sudah terbentuk dan dapat dipelajari pola datanya atau disebut juga *supervised learning* (Hermawati, 2013). Suatu fungsi menjadi dasar dalam memasukan beberapa atribut data ke suatu kelas tertentu. Klasifikasi memiliki kerangka kerja yang memiliki komponen utama yaitu pembangunan model. Data latih dimasukan ke dalam model kemudian dilakukan pembelajaran menggunakan algoritma pembelajaran. Setelah membangun model, model tersebut dapat diterapkan dalam melakukan proses penentuan label kelas terhadap data baru. Data tersebut dikenal sebagai data uji. Penerapan model tersebut menghasilkan keluaran dari hasil pengujian. Proses pembangunan model dari masuknya data latih serta algoritma pembelajaran menjadi suatu model merupakan proses induksi, sedangkan proses menggunakan data uji untuk menerapkan model disebut deduksi. Induksi dan deduksi merupakan 2 langkah umum dalam pembangunan model teknik klasifikasi. Keluaran (*output*) yang dihasilkan dari penerapan model klasifikasi adalah setiap vektor *input* dipetakan ke dalam label kelas yang tepat (Prasetyo, 2012).

Terdapat beberapa algoritma klasifikasi yang dibagi berdasarkan model pembelajarannya. Algoritma yang selalu melakukan pembelajaran terhadap data latih sehingga pada saat proses pemetaan vektor *input* ke dalam label kelas yang tepat tidak lagi membutuhkan data latih tapi hanya membutuhkan bobot-bobot hasil pembelajaran data latih disebut dengan *eager learner*. Algoritma *eager learner* dapat melakukan proses pemetaan lebih cepat, namun membutuhkan waktu yang lebih lama untuk melakukan pembelajaran tersebut. Sedangkan algoritma yang tidak atau lebih sedikit melakukan pembelajaran disebut juga *lazy learner*. Akibatnya, waktu yang digunakan untuk melakukan pembelajarn lebih cepat walaupun proses pemetaan vektor *input* ke dalam kelas yang tepat menjadi lebih lama dikarenakan sangat bergantung pada data latih. Setiap melakukan proses pemetaan, algoritma *lazy learner* selalu membutuhkan data latih dan membacanya kembali. Algoritma *Decision Tree*, *Support Vector Machine*, *Bayesian* serta *Artificial Neural Network* (ANN) merupakan beberapa contoh algoritma *eager learner*. Sedangkan algoritma Regresi linear dan *k-Nearest Neighbor* (k-NN) merupakan beberapa contoh algoritma *lazy learner* (Prasetyo, 2012).

2.4.2 K-Nearest Neighbor (k-NN)

Algoritma klasifikasi *k-Nearest Neighbor* (k-NN) adalah algoritma yang bergantung pada perhitungan suatu data dengan data yang memiliki label kelas tertentu dalam melakukan klasifikasi. Konsep dasar klasifikasi menggunakan k-

NN adalah seperti misalnya memungkinkan suatu makhluk hidup dinamakan ayam dikarenakan bentuk tubuh dan suaranya sama seperti ayam. Hasil perhitungan jarak antar data berupa nilai yang dijadikan dasar dalam menentukan suatu data tersebut lebih mirip atau lebih dekat dengan data yang memiliki label kelas apa sehingga suatu data tersebut dapat dipetakan ke dalam kelas yang sama. Variabel k pada algoritma ini adalah untuk menentukan berapa banyak data uji dengan nilai jarak terkecil yang dapat diambil dan ditentukan sebagai jarak terdekat (Prasetyo, 2012).

Algoritma k-NN dikenal sebagai algoritma yang sederhana. Banyaknya data latih mempengaruhi keefektifan algoritma k-NN. Semakin banyak data latih yang digunakan maka algoritma k-NN akan semakin efektif. Algoritma k-NN juga dinilai memiliki keakuratan yang baik dalam melakukan klasifikasi. Namun, algoritma k-NN sangat bergantung pada penentuan nilai dari variabel k , sehingga hasil klasifikasi dapat berbeda-beda dipengaruhi oleh nilai k tersebut (Gorunescu, 2011).

Ditentukan nilai k sebagai nilai yang menentukan berapa banyak data yang diambil untuk dilakukan pemetaan ke suatu kelas tertentu. Kemudian dilakukan perhitungan jarak data satu dengan lainnya. Persamaan yang biasa digunakan untuk menghitung jarak yaitu *Euclidean Distance*. Persamaan tersebut dapat dilihat pada Persamaan 2.1.

$$d(y, x) = \sqrt{\sum_{i=1}^n (y_i - x_i)^2} \quad (2.1)$$

Dengan :

d = jarak

n = jumlah atribut data

y_i = nilai data uji pada atribut ke- i

x_i = nilai data latih pada atribut ke- i

Berdasarkan nilai k yang telah ditentukan sebelumnya, hasil k jarak terdekat diambil kemudian ditentukan kelas data yang paling banyak dengan melakukan voting mayoritas (Nugraha, et al., 2017).

2.5 Fuzzy k-Nearest Neighbor (Fk-NN)

Dengan adanya keterbatasan yang dimiliki algoritma *k-Nearest Neighbor* (k-NN), maka algoritma tersebut telah dikembangkan. Pengembangan algoritma k-NN dilakukan dengan menggabungkan logika fuzzy ke dalam proses klasifikasinya. Di dalam logika fuzzy, dinyatakan bahwa setiap data sebenarnya dapat masuk ke semua kelas dengan nilai derajat keanggotaannya yang berbeda-beda (Prasetyo, 2012). Algoritma Fk-NN tidak hanya mempertimbangkan jarak terdekat data yang dievaluasi dengan data di kelas tertentu, tetapi juga mempertimbangkan nilai derajat keanggotaan data di setiap kelas. Fk-NN menghindari keambiguan terhadap hasil klasifikasi sehingga hasil klasifikasi dapat lebih tepat (Nugraha, et al., 2017). Seperti algoritma lainnya yang menggunakan

teknik fuzzy, Fk-NN dapat diterapkan pada objek masalah klasifikasi yang berkaitan dengan biologi atau bidang kedokteran (Syahiwa, et al., 2017). Tahapan yang digunakan untuk mengimplementasikan Fk-NN sama dengan k-NN, hanya saja setelah dilakukannya perhitungan jarak, kemudian digunakan suatu rumus seperti yang ditunjukkan pada Persamaan 2.

$$u(x, c_i) = \frac{\sum_{k=1}^K u(x_k, c_i) \times d(x, x_k)^{\frac{-2}{(m-1)}}}{\sum_{k=1}^K d(x, x_k)^{\frac{-2}{(m-1)}}} \quad (2.2)$$

Dengan :

$u(x, c_i)$ = nilai derajat keanggotaan data x ke kelas c_i

K = jumlah tetangga terdekat

$u(x_k, c_i)$ = nilai derajat keanggotaan data x ke- k pada kelas c_i (0/1)

$d(x, x_k)$ = jarak data x dengan data x_k

m = bobot pangkat (>1)

Berdasarkan hasil perhitungan nilai derajat keanggotaan setiap data uji ke setiap kelas, data uji tersebut dapat ditentukan langsung label kelasnya. Namun, ada pertimbangan lain yang ditemukan yaitu nilai derajat keanggotaan di kelas yang satu dengan kelas lainnya tidak terlalu jauh atau perbandingannya hanya sedikit. Hal itu menyebabkan label kelas yang diberikan terasa kurang meyakinkan. Berdasarkan hal tersebut penentuan label kelas dapat dilakukan dengan cara lain yaitu menentukan nilai *Threshold* yang menjadi batas kelulusan suatu data dapat memasuki suatu kelas tertentu. Jika data uji yang dievaluasi tidak melebihi nilai *Threshold* tersebut, maka data uji yang dievaluasi akan masuk ke kelas baru yang diinisialisasi sebagai c' (Prasetyo, 2012).

2.6 Akurasi

Akurasi merupakan proses evaluasi yang dilakukan agar dapat diketahui tingkat kebenaran suatu metode / algoritma yang digunakan dalam memecahkan suatu permasalahan (Syahiwa, et al., 2017). Perhitungan akurasi digunakan pada evaluasi metode klasifikasi karena metode klasifikasi belum tentu tepat digunakan untuk memecahkan suatu objek permasalahan sehingga diperlukan pengukuran terhadap kinerja metode klasifikasi tersebut (Prasetyo, 2012). Rumus menghitung tingkat akurasi dapat dilihat pada Persamaan 2.3.

$$\text{Akurasi (\%)} = \frac{\sum \text{data uji yang diklasifikasi dengan benar}}{\sum \text{data uji yang digunakan}} \times 100\% \quad (2.3)$$

BAB 3 METODOLOGI

Bab Metodologi menjelaskan langkah-langkah yang diambil pada saat melakukan penelitian. Pada Bab ini akan dijelaskan mengenai tahapan metode yang dilakukan dalam implementasi metode *Fuzzy k-Nearest Neighbor* (Fk-NN) pada permasalahan klasifikasi jenis kanker berdasarkan susunan protein. Dalam melakukan penelitian, tahapan awal yang dilakukan adalah mengidentifikasi sebuah masalah dan latar belakang masalah tersebut. Setelah teridentifikasinya sebuah masalah, ditentukan metode apa yang akan diimplementasikan dalam memecahkan masalah tersebut. Dalam menentukan metode, dilakukan studi literatur. Kemudian ditentukan tipe penelitian serta strategi penelitian.

3.1 Studi Literatur

Literatur yang digunakan, dipelajari dan dijadikan dasar teori dalam penelitian ini yaitu literatur berupa buku, jurnal, artikel web dan laporan-laporan penelitian sebelumnya yang berhubungan dengan objek masalah kanker, protein serta yang berhubungan dengan penerapan Bioinformatika, Klasifikasi pada Data Mining dan metode klasifikasi Fk-NN. Pembelajaran terhadap literatur-literatur tersebut dilakukan agar penelitian ini memiliki penunjang yang berdasar atau tidak sembarang dalam pelaksanaannya.

3.2 Tipe Penelitian

Penelitian yang dilakukan adalah mengkaji suatu masalah atau kejadian tertentu. Masalah dalam penelitian ini yaitu klasifikasikan jenis penyakit kanker berdasarkan susunan protein. Metode yang digunakan dalam penelitian ini yaitu *Fuzzy k-Nearest Neighbor* (Fk-NN). Penelitian ini bertujuan untuk menganalisis hubungan antar komponen penelitian tersebut. Dari hasil penelitian ini nantinya dapat diketahui apakah suatu metode yaitu *Fuzzy k-Nearest Neighbor* (Fk-NN) dapat diimplementasikan dan menghasilkan akurasi yang tinggi dalam pengklasifikasian jenis penyakit kanker berdasarkan susunan protein. Berdasarkan hal tersebut, maka penelitian yang dilakukan dapat dikategorikan ke dalam tipe penelitian *non implementatif - analitik*.

3.3 Strategi Penelitian

Strategi penelitian yang dilakukan pada penelitian ini yaitu berupa eksperimen, dimana eksperimen yang dilakukan adalah dengan mengimplementasikan algoritma *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk klasifikasi jenis penyakit kanker berdasarkan susunan protein. Algoritma tersebut diimplementasikan ke dalam sebuah program komputer dan kemudian dapat dilakukan pengujian agar dapat diketahui tingkat akurasi serta kesimpulan apakah metode tersebut cocok untuk diimplementasikan pada permasalahan tersebut.

3.4 Pengumpulan Data

Data yang digunakan merupakan data sekuens protein *human* TP53 *isoform* 1 dari <http://www.uniprot.org/> dan database TP53 dari <http://p53.free.fr/> . Contoh data yang digunakan dalam penelitian ini ditunjukkan seperti pada Tabel 3.1.

Tabel 3.1 Contoh Dataset

No.	Data Sekuens Protein	Kelas
1	MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDIEQ WFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQKTY QGSYGFRGLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDSTPPPGT RVRAMAIYKQSQHMTEVVRRCPPHHERCSDSDGLAPPQHILRVEGNLRVEYLD DRNTFRHSVVVPYEPPEVGSDCTTIHYNMCMSSCMGGMNRRPILTIITLEDSS SGNLLGRNSFEVRVCACPGRRRTEENLRKKGEPHHELPPGSTKRALPNNTS SSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALELKDAQAGKEPGGSRAHSS HLKSKKGQSTSRHKKLMFKTEGPDSD	0
2	MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDIEQ WFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQKTY QGSYGFRGLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDSTPPPGT RVRAMAIYKQSQHMTEVVRRCPPHHERCSDSDGLAPPQHILRVEGNLRVEYLD DRNTFRHSVVVPYEPPEVGSDCTTIHYNMCMSSCMGGMNRRPILTIITLEDSS SGNLLGRNSFEVRDCACPGRRRTEENLRKKGEPHHELPPGSTKRALPNNTS SSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALELKDAQAGKEPGGSRAHSS HLKSKKGQSTSRHKKLMFKTEGPDSD	1
3	MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDIEQ WFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQKTY QGSYGFRGLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDSTPPPGT RVRAMAIYKQSQHMTEVVRCLHHERCSDSDGLAPPQHILRVEGNLRVEYLD DRNTFRHSVVVPYEPPEVGSDCTTIHYNMCMSSCMGGMNRRPILTIITLEDSS SGNLLGRNSFEVRVCACPGRRRTEENLRKKGEPHHELPPGSTKRALPNNTS SSPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALELKDAQAGKEPGGSRAHSS HLKSKKGQSTSRHKKLMFKTEGPDSD	2
4	MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLPSQAMDDLMLSPDDIEQ WFTEDPGPDEAPRMPEAAPPVAPAPAAPTPAAPAPAPSWPLSSSVPSQKTY QGSYGFRGLGFLHSGTAKSVTCTYSPALNKMFCQLAKTCPVQLWVDSTPPPGT RVRAMAIYKQSQHMTEVVRRCPPHHERCSDSDGLAPPQHILRVEGNLRVEYLD DRNTFRHSVVVPYEPPEVGSDCTTIHYSYMCNSSCMGGMNRRPILTIITLEDSS GNLLGRNSFEVRVCACPGRRRTEENLRKKGEPHHELPPGSTKRALPNNTSS SPQPKKKPLDGEYFTLQIRGRERFEMFRELNEALELKDAQAGKEPGGSRAHSS HLKSKKGQSTSRHKKLMFKTEGPDSD	3

3.5 Perancangan

Algoritma yang digunakan pada penelitian klasifikasi jenis kanker berdasarkan susunan protein adalah *Fuzzy k-Nearest Neighbor* (Fk-NN). Algoritma Fk-NN

menggabungkan logika fuzzy dengan teknik klasifikasi k-NN. Fk-NN membantu menempatkan suatu atau beberapa data ke dalam kelas tertentu berdasarkan kelas data tetangga terdekat yang memiliki nilai derajat keanggotaan tertinggi. Pada bab perancangan akan dijelaskan mengenai perancangan algoritma Fk-NN, perancangan antarmuka serta perancangan pengujian.

3.6 Implementasi

Klasifikasi jenis kanker berdasarkan susunan protein dilakukan dengan mengimplementasikan metode Fk-NN ke dalam program komputer menggunakan bahasa pemrograman *Java*. Aplikasi yang digunakan adalah NetBeans IDE 8.1. Hasil yang dikeluarkan merupakan hasil penentuan kelas setiap data uji serta tingkat akurasi yang dihasilkan dari pengimplementasian metode Fk-NN.

3.7 Pengujian

Pengujian yang dilakukan yaitu menguji pengaruh nilai k terhadap tingkat akurasi dengan menggunakan *10-fold-cross-validation*. Pengujian dilakukan menggunakan perangkat pendukung meliputi perangkat keras dengan spesifikasi seperti di bawah ini :

1. Sistem Operasi : 64-bit *Operating System*, x64-based processor
2. *Installed memory* (RAM) : 4,00 GB
3. Prosesor : Intel(R) Core(TM) i3-5005U CPU @ 2.00GHz 2.00 GHz

Perangkat pendukung lainnya yang digunakan yaitu perangkat lunak NetBeans IDE 8.1.

3.8 Penarikan Kesimpulan dan Saran

Penarikan kesimpulan dilakukan setelah penelitian selesai dilakukan. Dari hasil implementasi serta pengujian yang telah dilakukan, dapat ditarik kesimpulan terhadap hasil penelitian yang nantinya dapat dimanfaatkan dan dikembangkan pada bidang yang bersangkutan. Saran ditujukan untuk memberi pandangan dan acuan kepada penelitian selanjutnya.

BAB 4 PERANCANGAN

Bab ini berisi penjelasan tentang perancangan yang dilakukan dalam implementasi metode *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk klasifikasi penyakit kanker berdasarkan susunan protein ke dalam program komputer. Perancangan penelitian ini dibagi menjadi 3 bagian yaitu perancangan algoritma, perancangan antarmuka dan perancangan pengujian. Dalam perancangan algoritma akan ditunjukkan diagram alir proses implementasi metode Fk-NN dan contoh perhitungan manual metode Fk-NN. Dalam perancangan pengujian akan ditunjukkan skenario pengujian yang dilakukan terhadap hasil implementasi.

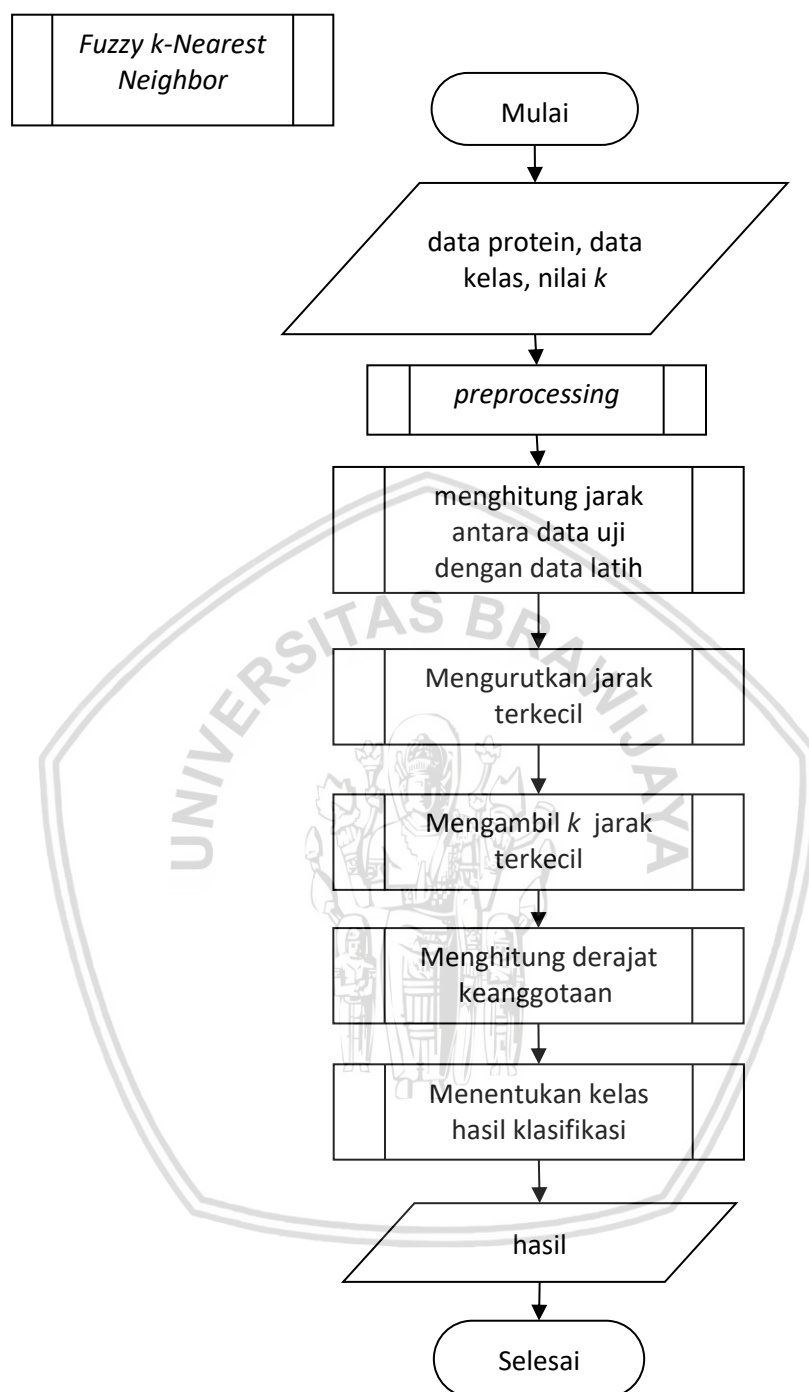
4.1 Perancangan Algoritma

Metode Fk-NN digunakan untuk mengklasifikasi penyakit kanker berdasarkan susunan protein. Metode Fk-NN memetakan data baru ke dalam suatu kelas tertentu yang merupakan kelas yang tepat. Seperti yang dilakukan metode *k-Nearest Neighbor* (k-NN) pada umumnya, konsep utama metode Fk-NN juga mencari tetangga terdekat data yang akan diklasifikasi sehingga data tersebut dapat masuk ke dalam kelas data yang menjadi tetangga terdekatnya. Proses yang dilakukan yaitu menghitung jarak antara data baru dengan data yang telah terklasifikasi ke dalam kelas tertentu. Kelas suatu data yang memiliki jarak terdekat dengan data baru akan menjadi kelas untuk data baru tersebut. Yang membedakan metode Fk-NN dengan k-NN pada umumnya adalah metode Fk-NN menentukan derajat keanggotaan masing-masing kelas berdasarkan k data terpilih. Derajat keanggotaan tertinggi menunjukkan kelas yang paling tepat.

4.1.1 Diagram Alir Fk-NN

Diagram alir yang akan dibuat, menunjukkan jalannya proses algoritma Fk-NN. Gambaran proses algoritma akan membantu dalam proses implementasi algoritma Fk-NN ke dalam program komputer. Berikut langkah-langkah yang dilakukan dalam proses keseluruhan algoritma Fk-NN sesuai dengan diagram alir pada Gambar 4.1.

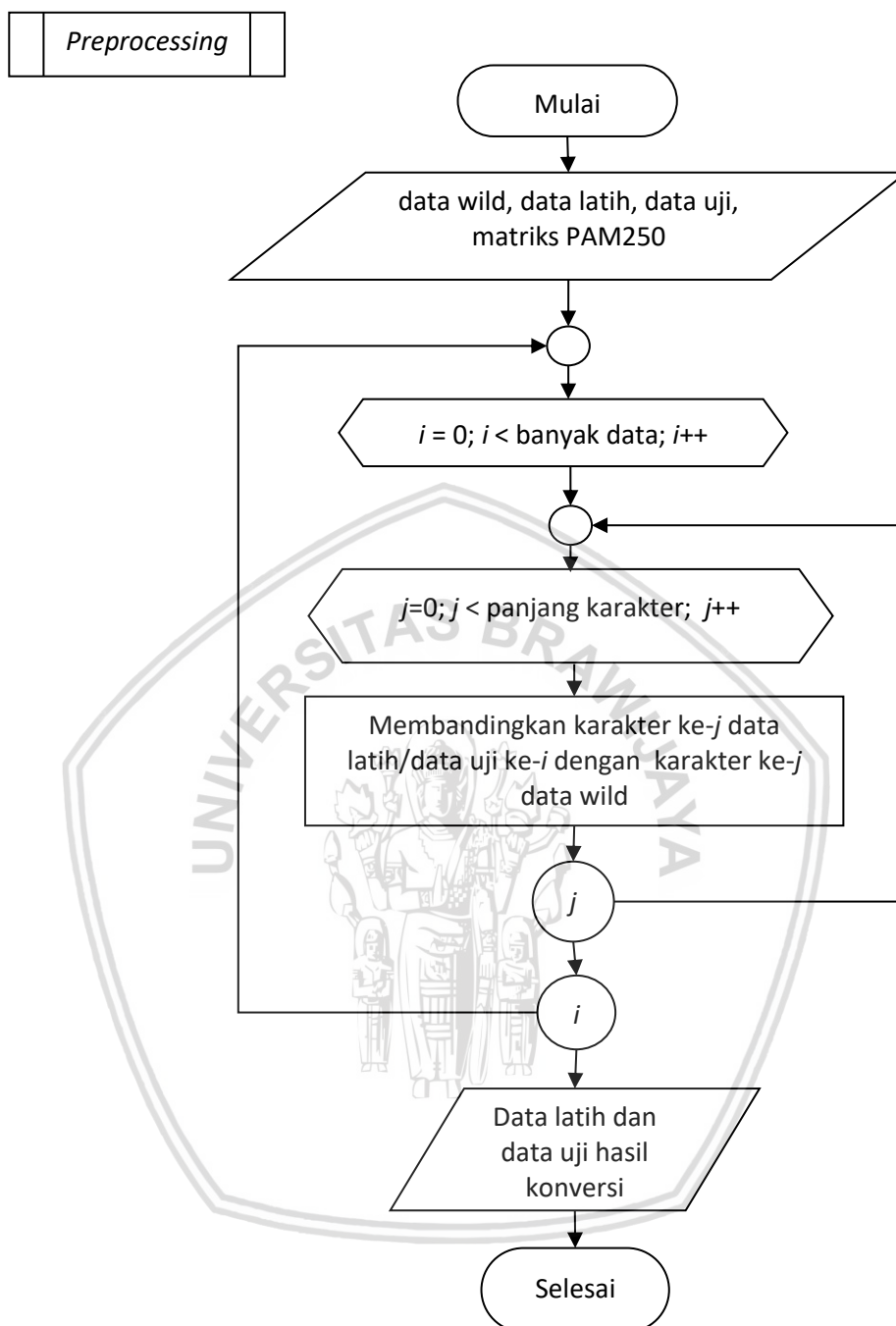
1. Memasukkan dataset protein dan nilai k
2. Melakukan *preprocessing*
3. Menghitung jarak data uji terhadap data latih
4. Mengambil k jarak terkecil
5. Mengurutkan nilai jarak terkecil
6. Menghitung derajat keanggotaan
7. Memilih derajat keanggotaan terbesar
8. Mengeluarkan hasil klasifikasi



Gambar 4.1 Diagram Alir Metode Fk-NN

4.1.1.1 Proses Preprocessing

Pada proses *Preprocessing*, Dataset Protein yang masuk akan dikonversi menggunakan Matriks PAM250 seperti yang ditunjukkan Tabel 2.3. Konversi tersebut dilakukan untuk mengganti tipe isi data dari huruf menjadi angka. Data protein dibandingkan dengan data wild yang telah dipilih. Kemudian, dari hasil perbandingan disesuaikan nilainya dengan yang ada pada Matriks PAM250.



Gambar 4.2 Diagram alir proses *preprocessing*

1. Memasukkan data latih, data uji, data wild dan data matriks PAM250
2. Memasukkan data latih/uji ke- i yang dikonversi sebagai nilai parameter baris matriks PAM250 dan data wild ke- i sebagai nilai parameter kolom matriks PAM250
3. Memanggil matriks PAM pada indeks baris dan kolom yang ditentukan
4. Mengeluarkan data latih dan data uji yang telah terkonversi

Menghitung jarak data uji terhadap data latih

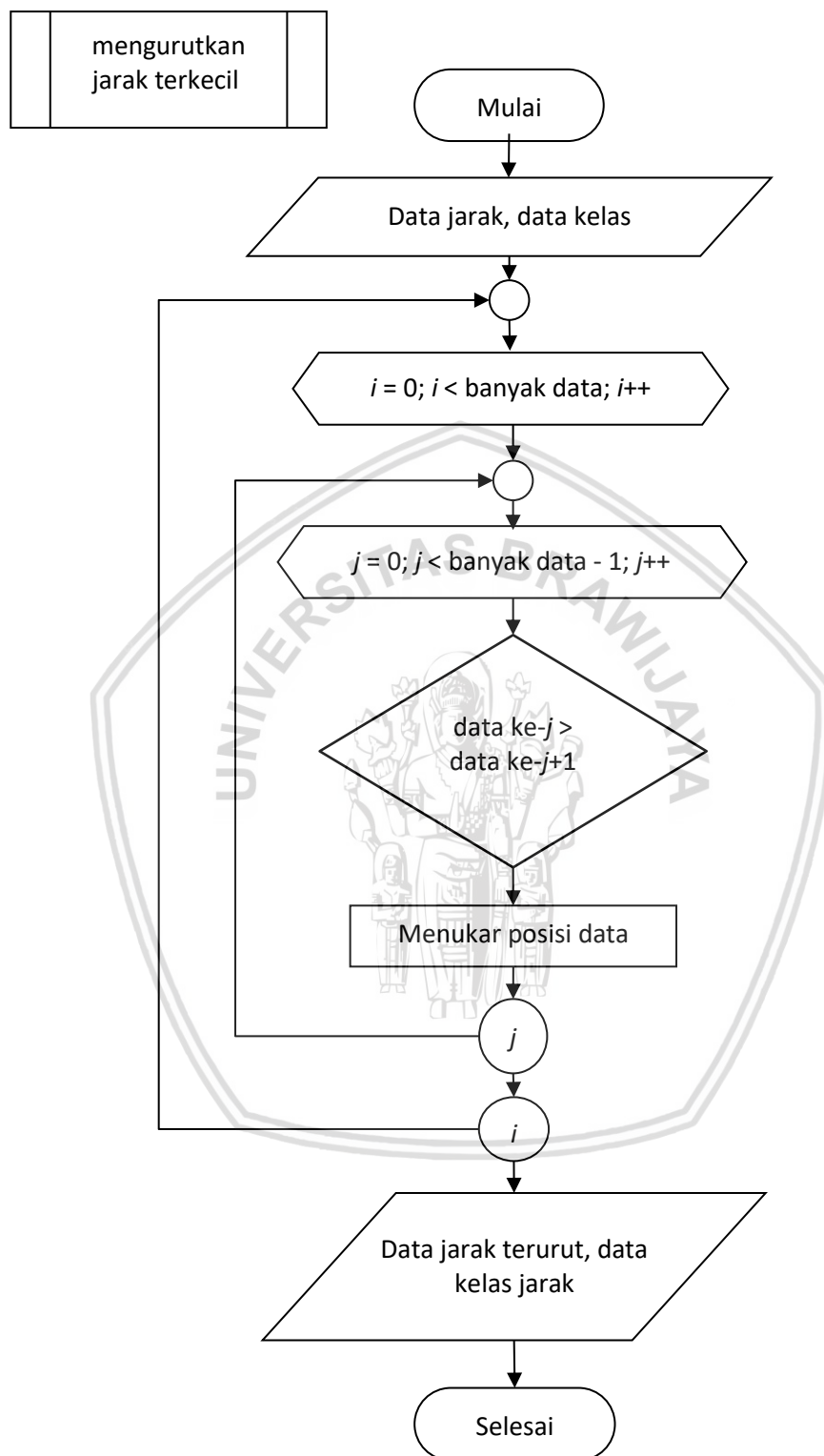
```

graph TD
    Start([Mulai]) --> Input[/Data latih dan data uji hasil konversi/]
    Input --> J1(( ))
    J1 --> LoopI{i = 0; i < banyak data latih; i++}
    LoopI --> J2(( ))
    J2 --> LoopJ{j = 0; j < banyak atribut; j++}
    LoopJ --> Process[
$$d(\text{data latih}(i), \text{data uji}(i)) = \sqrt{\sum_{j=0}^{\text{banyak atribut}} (\text{data latih}_j - \text{data uji}_j)^2}$$
]
    Process --> J3((j))
    J3 --> J4((i))
    J4 --> Output[/data jarak/]
    Output --> End([Selesai])
    J4 --> J2
  
```

Proses perhitungan menggunakan rumus *Euclidean Distance* seperti yang ditunjukkan pada Persamaan 2.1. Berikut langkah-langkah yang dilakukan.

1. Memasukkan data latih hasil konversi, data uji hasil konversi dan nilai k
2. Menghitung jumlah dari hasil kuadrat selisih nilai parameter ke- j data uji ke- i dengan data latih ke- i kemudian hasil penjumlahan tersebut diakarkan
3. Menghasilkan nilai jarak data uji dengan masing-masing data latih

4.1.1.3 Proses Mengurutkan Jarak Terkecil

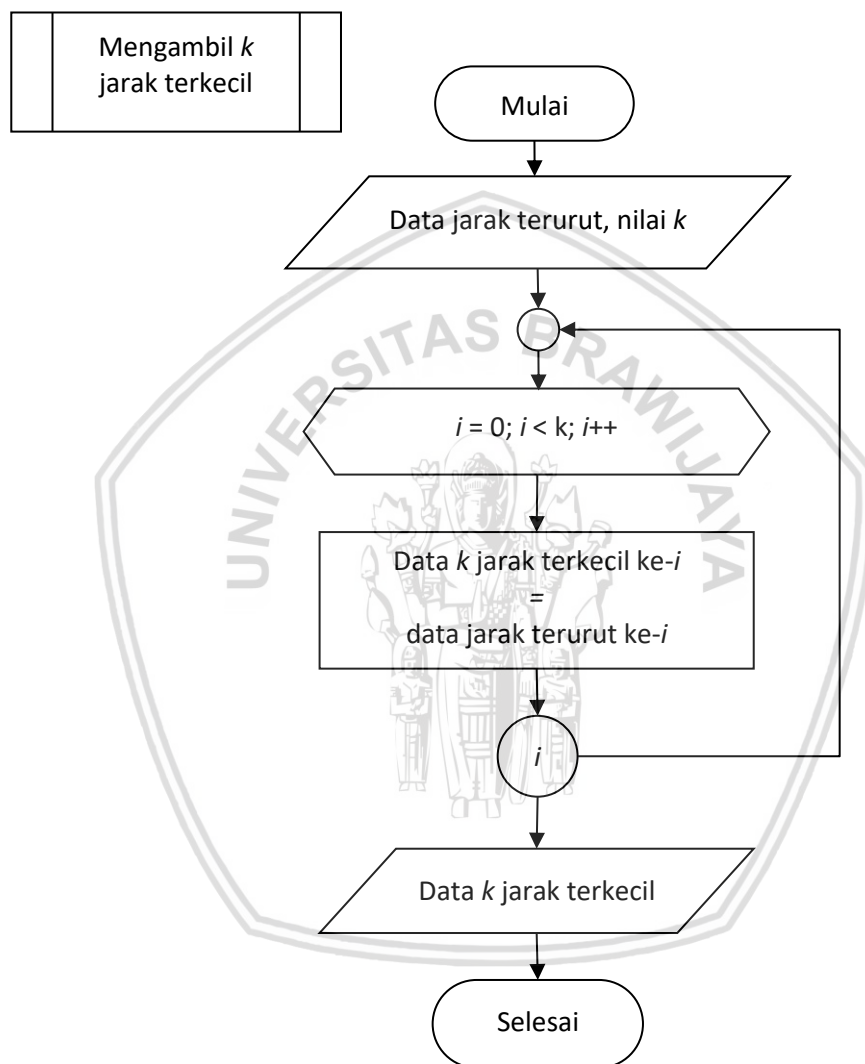


Gambar 4.4 Diagram alir proses mengurutkan jarak terkecil

Setelah mendapatkan hasil perhitungan jarak data uji dengan seluruh data latih, kemudian diurutkan berdasarkan hasil jarak terkecil. Berikut langkah-langkah yang dilakukan sesuai diagram alir pada Gambar 4.4.

1. Memasukkan hasil perhitungan jarak dan data kelas
2. Melakukan perulangan, jika data ke- j lebih besar dari data ke- $(j+1)$ maka data akan ditukar sehingga urutan data terkecil berada di indeks yang lebih kecil
3. Mengeluarkan jarak terurut

4.1.1.4 Proses mengambil k jarak terkecil

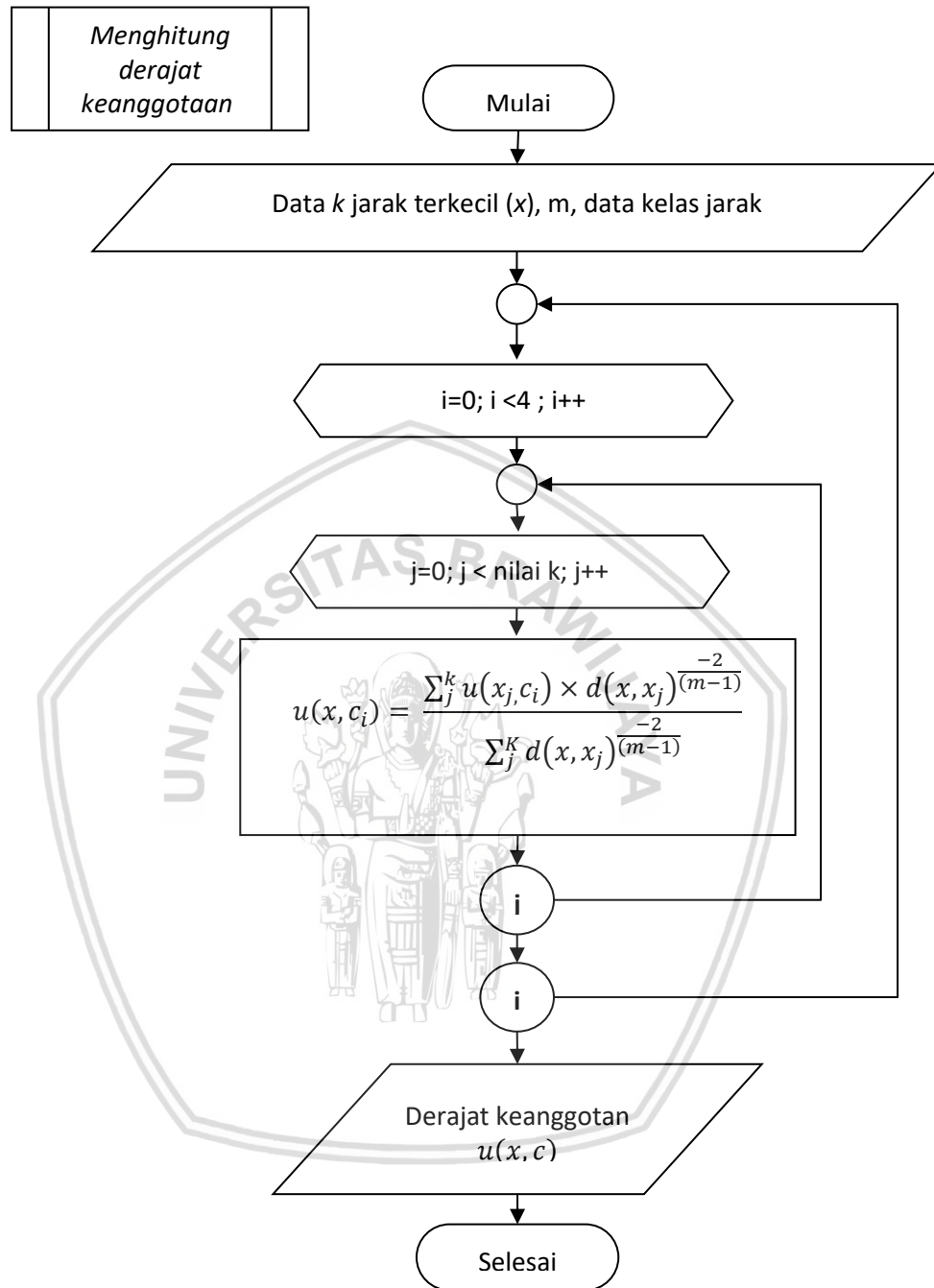


Gambar 4.5 Diagram alir proses mengambil k jarak terkecil

Langkah-langkah mengambil k jarak terkecil sesuai dengan Gambar 4.5 adalah sebagai berikut :

1. Memasukkan data hasil mengurutkan jarak terkecil dan nilai k
2. Melakukan perulangan untuk mengambil jarak terkecil dimulai dari data ke 1 sampai dengan data ke k
3. Mengeluarkan k jarak terkecil

4.1.1.5 Proses Menghitung Derajat Keanggotaan

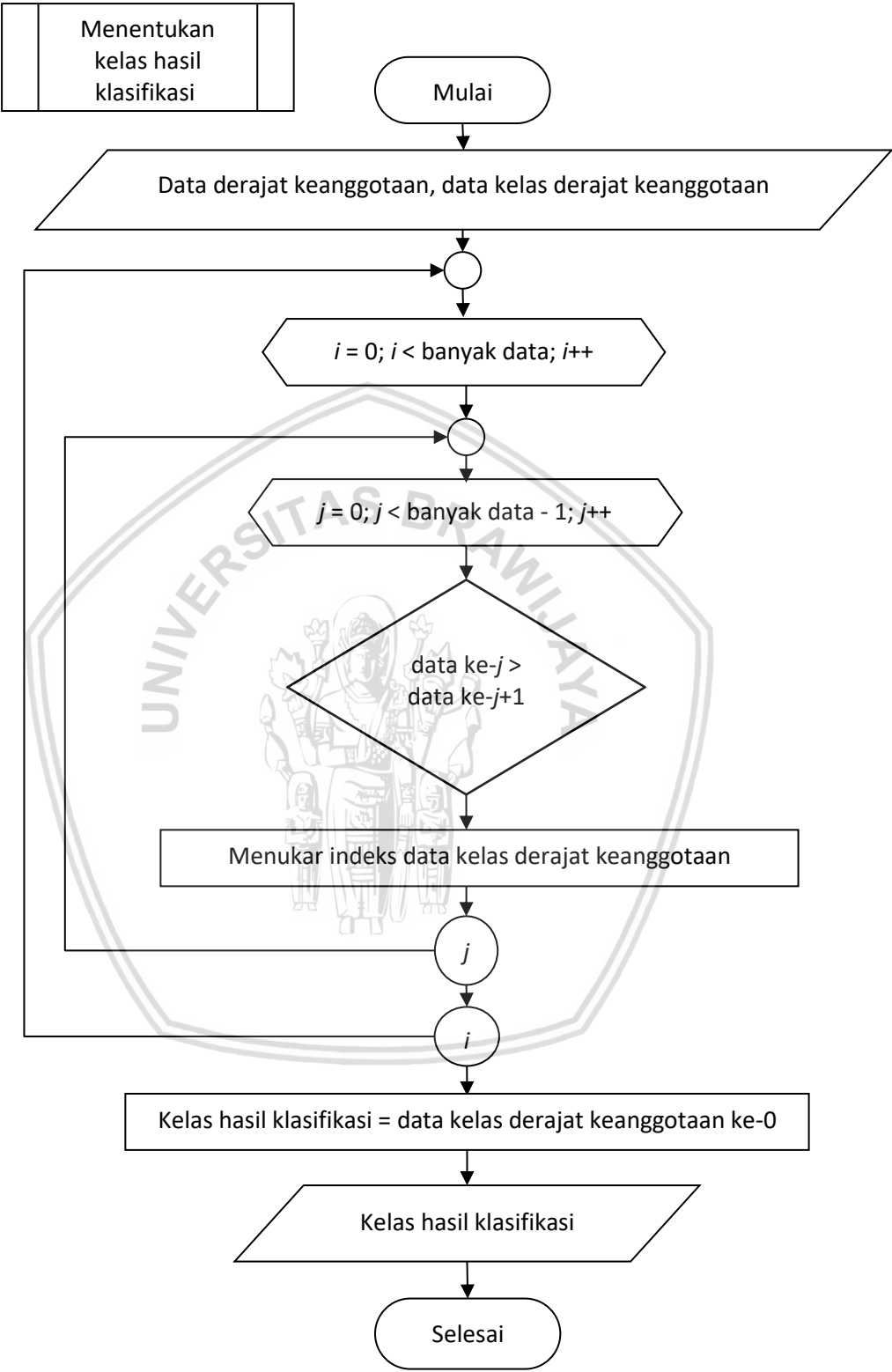


Gambar 4.6 Diagram alir proses Fk-NN

Langkah-langkah yang dilakukan adalah sebagai berikut.

1. Memasukkan sejumlah k data dengan nilai jarak terkecil dan nilai $m = 4$
2. Melakukan perulangan proses perhitungan derajat keanggotaan setiap kelas dengan nilai keanggotaan setiap data kelas (x_j) tersebut adalah 1, sedangkan nilai keanggotaan setiap data kelas lain adalah 0
3. Mengeluarkan hasil derajat keanggotaan

4.1.1.6 Proses menentukan kelas hasil klasifikasi



Gambar 4.7 Diagram alir proses menentukan kelas hasil klasifikasi

Proses ini merupakan proses terakhir dalam implementasi metode Fk-NN. Pada proses ini, derajat keanggotaan yang dihasilkan proses sebelumnya diurutkan berdasarkan nilai yang terbesar. Kelas dengan derajat keanggotaan terbesar menjadi kelas terklasifikasi. Berikut langkah-langkah pada proses yang ditunjukkan Gambar 4.7.

1. Memasukkan derajat keanggotaan masing-masing kelas
2. Melakukan perulangan, jika derajat keanggotaan ke- j lebih kecil dari derajat keanggotaan ke- $j+1$ maka posisi derajat keanggotaan akan ditukar sehingga derajat keanggotaan berada di indeks yang lebih kecil
3. Mengambil kelas derajat keanggotaan terbesar
4. Mengeluarkan kelas hasil klasifikasi

4.1.2 Perhitungan Manual

Dalam melakukan perhitungan manual, data yang digunakan merupakan data sampel yang diambil dari dataset protein yang akan digunakan dalam pengimplementasian metode Fk-NN. Data sampel berupa 1 data wild, 20 data latih dan 1 data uji yang berisi susunan 10 asam amino dari dataset protein. Data latih telah terbagi menjadi beberapa kelas yaitu kelas NC (*Non Cancer*), BC (*Breast Cancer*), CC (*Colorectal Cancer*) dan LC (*Lung Cancer*). Data wild yang digunakan ditunjukkan pada Tabel 4.1, data latih pada Tabel 4.2 dan data uji pada Tabel 4.3.

Tabel 4.1 Data Wild

Data Wild	Sekuens Protein ke-										kelas
	175	176	177	178	179	180	181	182	183	184	
1	R	C	P	H	H	E	R	C	S	D	NC

Tabel 4.2 Data latih

Data Protein	Sekuens Protein ke-										Kelas
	175	176	177	178	179	180	181	182	183	184	
1	R	C	P	H	H	E	R	C	S	D	NC
2	R	C	P	H	H	E	P	C	P	D	CC
3	R	C	P	H	H	E	R	C	L	D	BC
4	R	C	P	H	H	E	R	C	S	D	CC'
5	R	C	P	H	H	E	R	C	S	D	LC
6	R	C	P	H	R	E	R	C	S	D	BC
7	R	C	P	H	H	E	R	C	S	D	CC
8	R	C	P	H	H	E	R	Y	S	D	CC
9	R	C	P	H	H	E	R	A	S	D	NC
10	G	C	P	H	H	E	R	C	S	D	NC

11	R	C	P	H	H	E	R	C	S	D	BC
----	---	---	---	---	---	---	---	---	---	---	----

Tabel 4.2 Data latih (lanjutan)

Data Protein	Sekuens Protein ke-										Kelas
	175	176	177	178	179	180	181	182	183	184	
12	R	C	P	N	H	E	R	C	S	D	LC
13	R	C	L	H	H	E	R	C	S	D	LC
14	R	S	P	H	H	E	R	C	S	D	NC
15	R	F	P	H	H	E	R	C	S	D	CC
16	C	C	P	H	H	E	R	C	S	D	LC
17	R	C	P	H	H	E	R	C	S	D	NC
18	R	C	P	H	H	E	R	C	S	D	BC
19	R	C	P	H	H	K	R	C	S	D	LC
20	R	C	R	H	H	E	R	C	S	D	BC

Tabel 4.3 Data uji

Data Wild	Sekuens Protein ke-										Kelas
	175	176	177	178	179	180	181	182	183	184	
1	R	C	P	H	H	E	R	C	S	Y	?

Kemudian dilakukan proses *preprocessing* yaitu mengkonversi data latih dan data uji ke dalam bentuk angka dengan cara membandingkan data latih dan data uji masing-masing dengan data wild menggunakan tabel matriks PAM250 seperti yang ditunjukkan pada Tabel 2.3. Misal karakter Hasil proses *preprocessing* ditunjukkan pada Tabel 4.4 untuk hasil konversi data latih dan Tabel 4.5 untuk hasil konversi data uji.

Tabel 4.4 Hasil konversi data latih

Data Protein	Sekuens Protein ke-										kelas
	175	176	177	178	179	180	181	182	183	184	
1	6	12	6	6	6	4	6	12	2	4	NC
2	6	12	6	6	6	4	0	12	2	4	CC
3	6	12	6	6	6	4	6	12	-2	4	BC
4	6	12	6	6	6	4	6	12	2	4	CC
5	6	12	6	6	6	4	6	12	2	4	LC
6	6	12	6	6	1	4	6	12	2	4	BC
7	6	12	6	6	6	4	6	12	2	4	CC
8	6	12	6	6	6	4	6	1	2	4	CC
9	6	12	6	6	6	4	6	-2	2	4	NC
10	-3	12	6	6	6	4	6	12	2	4	NC

Tabel 4.4 Hasil konversi data latih (lanjutan)

Data Protein	Sekuens Protein ke-										kelas
	175	176	177	178	179	180	181	182	183	184	
11	6	12	6	6	6	4	6	12	2	4	BC
12	6	12	6	1	6	4	6	12	2	4	LC
13	6	12	-2	6	6	4	6	12	2	4	LC
14	6	1	6	6	6	4	6	12	2	4	NC
15	6	-4	6	6	6	4	6	12	2	4	CC
16	-3	12	6	6	6	4	6	12	2	4	LC
17	6	12	6	6	6	4	6	12	2	4	NC
18	6	12	6	6	6	4	6	12	2	4	BC
19	6	12	6	6	6	0	6	12	2	4	LC
20	6	12	0	6	6	4	6	12	2	4	BC

Tabel 4.5 Hasil konversi data uji

Data Wild	Sekuens Protein ke-										kelas
	175	176	177	178	179	180	181	182	183	184	
1	6	12	6	6	6	4	6	12	2	-4	?

Hasil konversi data latih dan data uji kemudian digunakan di dalam proses perhitungan metode *k-Nearest Neighbor*. Dalam proses ini dilakukan perhitungan jarak antara data uji terhadap data latih menggunakan rumus *Euclidean Distances* seperti yang ditunjukkan pada Persamaan 2.2. Berikut contoh perhitungan jarak antara data uji dengan data latih baris ke 1.

$$\begin{aligned}
 d(y, x) &= \sqrt{\sum_i^n (y_i - x_i)^2} \\
 &= \sqrt{(6 - 6)^2 + (12 - 12)^2 + (6 - 6)^2 + (6 - 6)^2 + (6 - 6)^2 + (4 - 4)^2 + (6 - 6)^2 + (12 - 12)^2 + (2 - 2)^2 + ((-4) - 4)^2} \\
 &= \sqrt{(-8)^2} \\
 &= \sqrt{64} \\
 &= 8
 \end{aligned}$$

Hasil perhitungan jarak data uji terhadap data latih ditunjukkan pada Tabel 4.6.

Tabel 4.6 Hasil perhitungan jarak

Data Protein	Jarak	Kelas
1	8,00	NC
2	8,00	CC
3	8,94	BC
4	8,00	CC
5	8,00	LC
6	9,43	BC
7	8,00	CC
8	13,60	CC
9	16,12	NC
10	12,04	NC
11	8,00	BC
12	9,43	LC
13	11,31	LC
14	13,60	NC
15	17,89	CC
16	12,04	LC
17	8,00	NC
18	8,00	BC
19	8,94	LC
20	10,00	BC

Dari seluruh hasil perhitungan jarak data uji terhadap data latih, diambil sejumlah k data yang memiliki nilai jarak terbesar. Dalam perhitungan manual ini, digunakan $k = 5$, sehingga data hasil perhitungan adalah sebagai berikut.

Tabel 4.7 Nilai jarak terdekat

Data Protein	Jarak	Kelas
1	8	NC
2	8	CC
4	8	CC
5	8	LC
7	8	CC

Data hasil proses perhitungan *k-Nearest Neighbor* kemudian masuk ke dalam proses *Fuzzy k-Nearest Neighbor* yaitu menghitung nilai keanggotaan tiap kelas. Menghitung nilai keanggotaan menggunakan rumus seperti pada Tabel 2.3. Berikut contoh perhitungan nilai keanggotaan (U_i) dengan misal nilai $m=2$. Nilai keanggotaan U_1 untuk kelas *Non Cancer* (NC), U_2 untuk kelas *Breast Cancer* (BC), U_3 untuk kelas *Colorectal Cancer* (CC), U_i untuk kelas *Lung Cancer* (LC).

$$U_1 = \frac{1(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}})}{(8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}})} = 0,2$$

$$U_2 = \frac{0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}})}{(8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}})} = 0,0$$

$$U_3 = \frac{0(8^{\frac{-2}{(2-1)}}) + 1(8^{\frac{-2}{(2-1)}}) + 1(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 1(8^{\frac{-2}{(2-1)}})}{(8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}})} = 0,75$$

$$U_4 = \frac{0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}}) + 1(8^{\frac{-2}{(2-1)}}) + 0(8^{\frac{-2}{(2-1)}})}{(8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}}) + (8^{\frac{-2}{(2-1)}})} = 0,25$$

Tabel 4.8 Derajat keanggotaan kelas NC

Data Protein	Jarak	Kelas	U(NC)	Nilai Keanggotaan
1	8	NC	1	0,200
2	8	CC	0	
4	8	CC	0	
5	8	LC	0	
7	8	CC	0	

Tabel 4.9 Derajat keanggotaan kelas BC

Data Protein	Jarak	Kelas	U(BC)	Nilai Keanggotaan
1	8	NC	0	0,000
2	8	CC	0	
4	8	CC	0	
5	8	LC	0	
7	8	CC	0	

Tabel 4.10 Derajat keanggotaan kelas CC

Data Protein	Jarak	Kelas	U(CC)	Nilai Keanggotaan
1	8	NC	0	0,750
2	8	CC	1	
4	8	CC	1	
5	8	LC	0	
7	8	CC	1	

Tabel 4.11 Derajat keanggotaan kelas LC

Data Protein	Jarak	Kelas	U(LC)	Nilai Keanggotaan
1	8	NC	0	0,250
2	8	CC	0	
4	8	CC	0	
5	8	LC	1	
7	8	CC	0	

Dari hasil perhitungan derajat keanggotaan masing-masing kelas, selanjutnya dapat dipilih satu nilai derajat keanggotaan terbaik sebagai hasil klasifikasi. Berdasarkan hasil perhitungan manual tersebut, data uji telah terklasifikasi masuk ke dalam kelas CC (*Colorectal Cancer*) dengan nilai derajat keanggotaan sebesar 0.75.

4.2 Perancangan Antarmuka

Antarmuka dibangun untuk memudahkan penelitian terutama dalam menampilkan keluaran dan melakukan pengujian terhadap hasil implementasi metode Fk-NN ke dalam program komputer menggunakan bahasa pemrograman *Java*. Rancangan antarmuka pada pengimplementasian metode Fk-NN untuk klasifikasi penyakit kanker berdasarkan susunan protein ditunjukkan pada Gambar 4.7. Berikut elemen-elemen yang digunakan dalam perancangan antarmuka :

1. *TextField* untuk memasukkan angka 1-8 yang digunakan untuk menentukan *fold* ke berapa yang akan dieksekusi
2. *TextField* untuk memasukkan nilai *k* yang digunakan untuk melakukan proses perhitungan metode Fk-NN
3. *Button Submit* untuk menampilkan hasil proses perhitungan metode Fk-NN serta tingkat akurasi yang dihasilkan
4. *JTabbedPane* Data yang berisi tab-tab panel untuk menampilkan data-data yang digunakan dalam melakukan proses perhitungan metode Fk-NN seperti data wild, data latih, data uji, kelas data latih dan kelas data uji
5. *JTabbedPane* hasil yang berisi tab-tab panel untuk menampilkan hasil normalisasi data latih, normalisasi data uji, perhitungan jarak, mengurutkan jarak terdekat, mengambil *k* jarak terdekat, derajat keanggotaan masing-masing kelas dan kelas hasil klasifikasi
6. *TextField* untuk menampilkan hasil perhitungan tingkat akurasi terhadap hasil perhitungan metode Fk-NN
7. *Button Reset* untuk mengembalikan isi seluruh elemen *Jframe* ke keadaan kosong sehingga dapat dioperasikan kembali

Gambar 4.8 Perancangan antarmuka

4.3 Perancangan Pengujian

Pengujian dilakukan untuk menghasilkan tingkat akurasi. Teknik pengujian dilakukan dengan menggunakan dataset sebanyak 752 data. Pengujian yang dilakukan yaitu untuk menguji pengaruh nilai k terhadap tingkat akurasi. Pengujian pengaruh nilai k dilakukan menggunakan *k-fold-validation* yang mana data set dibagi menjadi k *fold* data. Dalam pengujian, secara bergantian salah satu *fold* dijadikan sebagai data uji sedangkan *fold* lainnya digabung menjadi data latih. Dari 752 dataset, ditentukan nilai k pada *k-fold-validation* yaitu 8 sehingga data set akan dibagi menjadi 8 *fold*. Nilai k yang akan diuji pada implementasi metode Fk-NN ini adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.

Tabel 4.12 Perancangan pengujian pengaruh nilai k

Nilai k	Fold					Rata-rata Akurasi
	1	2	3	...	8	
1						
2						
...						
15						
Rata-rata Akurasi						

Gambar 4.8 Perancangan antarmuka

4.3 Perancangan Pengujian

Pengujian dilakukan untuk menghasilkan tingkat akurasi. Teknik pengujian dilakukan dengan menggunakan dataset sebanyak 752 data. Pengujian yang dilakukan yaitu untuk menguji pengaruh nilai k terhadap tingkat akurasi. Pengujian pengaruh nilai k dilakukan menggunakan *k-fold-validation* yang mana data set dibagi menjadi k fold data. Dalam pengujian, secara bergantian salah satu *fold* dijadikan sebagai data uji sedangkan *fold* lainnya digabung menjadi data latih. Dari 752 dataset, ditentukan nilai k pada *k-fold-validation* yaitu 8 sehingga data set akan dibagi menjadi 8 *fold*. Nilai k yang akan diuji pada implementasi metode Fk-NN ini adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.

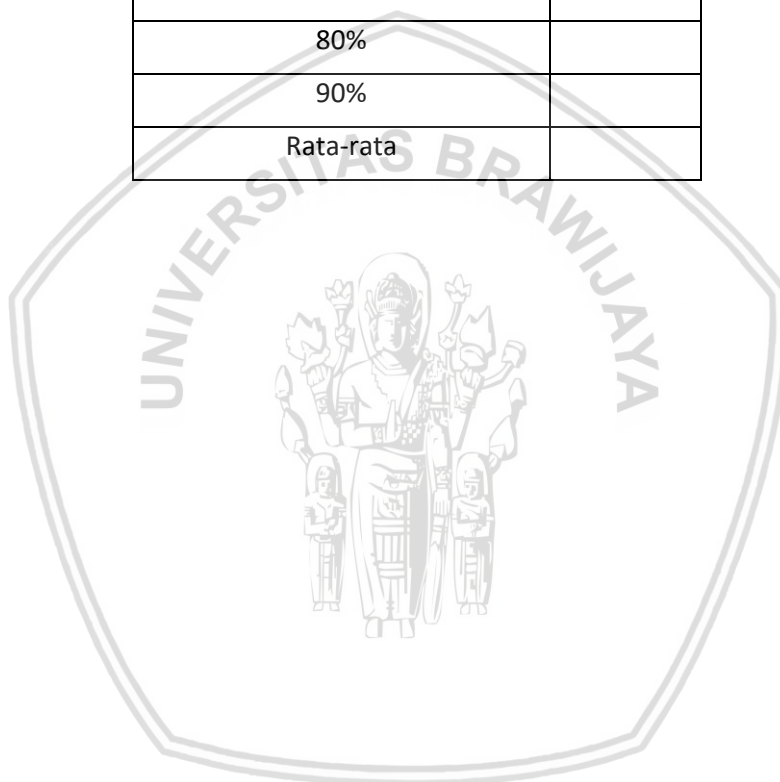
Tabel 4.12 Perancangan pengujian pengaruh nilai k

Nilai k	Fold					Rata-rata Akurasi
	1	2	3	...	8	
1						
2						
...						
15						
Rata-rata Akurasi						

Setelah dilakukan pengujian pengaruh nilai k , kemudian dilakukan pengujian pengaruh variasi data latih terhadap tingkat akurasi dengan nilai k yang digunakan adalah nilai k optimal dari hasil pengujian sebelumnya. Variasi data latih yang digunakan adalah 60%, 70%, 80% dan 90% dari jumlah dataset. Jumlah data uji yang digunakan dalam melakukan pengujian ini yaitu 20% dari dataset atau sebesar 150 data. Perancangan pengaruh variasi data latih terhadap tingkat akurasi ditunjukkan pada Tabel 4.13.

Tabel 4.13 Perancangan pengujian pengaruh variasi data

Variasi data latih dan data uji	Rata-rata
60%	
70%	
80%	
90%	
Rata-rata	



BAB 5 IMPLEMENTASI

Bab ini berisi implementasi metode Fuzzy k-Nearest Neighbor (Fk-NN) untuk klasifikasi jenis kanker berdasarkan susunan protein sesuai dengan perancangan yang telah dilakukan pada Bab 4. Implementasi kode program metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein terdiri dari beberapa proses berdasarkan perancangan yang telah dilakukan.

5.1 Implementasi Kode Program

5.1.1 Kode program *preprocessing*

Proses *preprocessing* merupakan proses mengkonversi data latih dan data uji menggunakan matriks PAM250 dengan membandingkan data-data tersebut dengan data *wild* (normal). Implementasi kode program yang akan dijabarkan pada proses ini meliputi proses konversi data latih dan data uji menggunakan matriks PAM250.

Kode Program 5.1 Proses *preprocessing*

Baris	Kode Sumber
1	//MEMBENTUK MATRIKS PAM
2	Double[][] PAM = new Double[200][200];
3	PAM['A']['A'] = PAM['N']['N'] = PAM['S']['S'] = PAM['T']['T'] = PAM['N']['D'] = PAM['D']['N'] = PAM['D']['Q'] = PAM['Q']['D'] = PAM['L']['I'] = PAM['I']['L'] = PAM['I']['M'] = PAM['M']['I'] = PAM['L']['F'] = PAM['F']['L'] = PAM['A']['S'] = PAM['S']['A'] = PAM['N']['S'] = PAM['S']['N'] = PAM['G']['S'] = PAM['S']['G'] = PAM['P']['S'] = PAM['S']['P'] = PAM['R']['W'] = PAM['W']['R'] = PAM['M']['V'] = PAM['V']['M'] = PAM['L']['V'] = PAM['V']['L'] = PAM['E']['Q'] = PAM['Q']['E'] = 2.0;
4	PAM['R']['R'] = PAM['H']['H'] = PAM['L']['L'] = PAM['M']['M'] = PAM['P']['P'] = 6.0;
5	PAM['D']['D'] = PAM['Q']['Q'] = PAM['E']['E'] = PAM['I']['I'] = PAM['V']['V'] = PAM['L']['M'] = PAM['M']['L'] = PAM['I']['V'] = PAM['V']['I'] = 4.0;
6	PAM['C']['C'] = 12.0;
7	PAM['G']['G'] = PAM['K']['K'] = 5.0;
8	PAM['F']['F'] = 9.0;
9	PAM['W']['W'] = 17.0;
10	PAM['Y']['Y'] = 10.0;
11	PAM['A']['R'] = PAM['R']['A'] = PAM['A']['C'] = PAM['C']['A'] = PAM['R']['I'] = PAM['I']['R'] = PAM['N']['I'] = PAM['I']['N'] = PAM['D']['I'] = PAM['I']['D'] = PAM['I']['C'] = PAM['C']['I'] = PAM['Q']['I'] = PAM['I']['Q'] = PAM['E']['I'] = PAM['I']['E'] = PAM['L']['A'] = PAM['A']['L'] = PAM['Q']['L'] = PAM['L']['Q'] = PAM['L']['H'] = PAM['H']['L'] = PAM['G']['K'] = PAM['K']['G'] = PAM['I']['K'] = PAM['K']['I'] = PAM['N']['M'] = PAM['M']['N'] = PAM['E']['M'] = PAM['M']['E'] = PAM['H']['M'] = PAM['M']['H'] = PAM['H']['F'] =

12	PAM['F']['H'] = PAM['I']['P'] = PAM['P']['I'] =
	PAM['L']['P'] = PAM['P']['L'] = PAM['M']['P'] =
	PAM['P']['M'] = PAM['S']['L'] = PAM['L']['S'] =
	PAM['F']['S'] = PAM['S']['F'] = PAM['L']['T'] =
	PAM['T']['L'] = PAM['H']['T'] = PAM['T']['H'] =
	PAM['T']['Q'] = PAM['Q']['T'] = PAM['R']['T'] =
	PAM['T']['R'] = PAM['S']['W'] = PAM['W']['S'] =
	PAM['M']['Y'] = PAM['Y']['M'] = PAM['N']['Y'] =
	PAM['Y']['N'] = -2.0;
	PAM['K']['V'] = PAM['V']['K'] = PAM['H']['V'] =
	PAM['V']['H'] = PAM['G']['V'] = PAM['V']['G'] =
	PAM['E']['V'] = PAM['V']['E'] = PAM['Q']['V'] =
13	PAM['V']['Q'] = PAM['C']['V'] = PAM['V']['C'] =
	PAM['D']['V'] = PAM['V']['D'] = PAM['N']['V'] =
	PAM['V']['N'] = PAM['R']['V'] = PAM['V']['R'] = -2.0;
	PAM['A']['N'] = PAM['N']['A'] = PAM['R']['N'] =
	PAM['N']['R'] = PAM['A']['D'] = PAM['D']['A'] =
	PAM['Q']['A'] = PAM['A']['Q'] = PAM['E']['A'] =
	PAM['A']['E'] = PAM['N']['G'] = PAM['G']['N'] =
	PAM['E']['G'] = PAM['G']['E'] = PAM['E']['H'] =
	PAM['H']['E'] = PAM['D']['K'] = PAM['D']['K'] =
	PAM['K']['E'] = PAM['E']['K'] = PAM['H']['K'] =
	PAM['K']['H'] = PAM['M']['F'] = PAM['F']['M'] =
	PAM['N']['P'] = PAM['P']['N'] = PAM['D']['P'] =
14	PAM['P']['D'] = PAM['P']['Q'] = PAM['Q']['P'] =
	PAM['E']['P'] = PAM['P']['E'] = PAM['G']['P'] =
	PAM['P']['G'] = PAM['H']['P'] = PAM['P']['H'] =
	PAM['H']['S'] = PAM['S']['H'] = PAM['S']['Q'] =
	PAM['Q']['S'] = PAM['A']['T'] = PAM['T']['A'] =
	PAM['N']['T'] = PAM['T']['N'] = PAM['P']['T'] =
	PAM['T']['P'] = PAM['H']['Y'] = PAM['Y']['H'] =
	PAM['W']['Y'] = PAM['Y']['W'] = PAM['T']['V'] =
	PAM['V']['T'] = PAM['A']['V'] = PAM['V']['A'] =
	PAM['P']['R'] = PAM['R']['P'] = 0.0;
	PAM['R']['D'] = PAM['D']['R'] = PAM['G']['Q'] =
	PAM['Q']['G'] = PAM['D']['C'] = PAM['A']['H'] =
15	PAM['H']['A'] = PAM['I']['A'] = PAM['A']['I'] =
	PAM['K']['A'] = PAM['A']['K'] = PAM['A']['M'] =
	PAM['M']['A'] = PAM['R']['M'] = PAM['M']['R'] =
	PAM['M']['Q'] = PAM['Q']['M'] = PAM['P']['C'] =
	PAM['C']['P'] = PAM['K']['P'] = PAM['P']['K'] =
	PAM['I']['S'] = PAM['S']['I'] = PAM['M']['S'] =
	PAM['S']['M'] = PAM['M']['T'] = PAM['T']['M'] =
	PAM['K']['T'] = PAM['T']['K'] = PAM['I']['T'] =
	PAM['T']['I'] = PAM['G']['T'] = PAM['T']['G'] =
	PAM['E']['T'] = PAM['T']['E'] = PAM['T']['D'] =
	PAM['D']['T'] = PAM['L']['Y'] = PAM['Y']['L'] =
	PAM['I']['Y'] = PAM['Y']['I'] = PAM['S']['V'] =
15	PAM['V']['S'] = PAM['P']['V'] = PAM['V']['P'] =
	PAM['F']['V'] = PAM['V']['F'] = -1.0;
	PAM['C']['R'] = PAM['R']['C'] = PAM['G']['R'] =
	PAM['R']['G'] = PAM['G']['C'] = PAM['C']['G'] =
	PAM['H']['C'] = PAM['C']['H'] = PAM['H']['G'] =
	PAM['G']['H'] = PAM['G']['I'] = PAM['I']['G'] =
	PAM['H']['I'] = PAM['I']['H'] = PAM['R']['L'] =
	PAM['L']['R'] = PAM['N']['L'] = PAM['L']['N'] =
	PAM['L']['E'] = PAM['E']['L'] = PAM['L']['K'] =
	PAM['K']['L'] = PAM['M']['D'] = PAM['D']['M'] =
	PAM['G']['M'] = PAM['M']['G'] = PAM['A']['F'] =

16	PAM['F']['A'] = PAM['D']['F'] = PAM['F']['D'] = PAM['P']['C'] = PAM['C']['P'] = PAM['C']['T'] = PAM['T']['C'] = PAM['T']['Y'] = PAM['Y']['T'] = PAM['S']['Y'] = PAM['Y']['S'] = PAM['A']['Y'] = PAM['Y']['A'] = PAM['Y']['V'] = PAM['V']['Y'] = -3.0; PAM['N']['C'] = PAM['C']['N'] = PAM['L']['D'] = PAM['D']['L'] = PAM['G']['L'] = PAM['L']['G'] = PAM['R']['F'] = PAM['F']['R'] = PAM['C']['F'] = PAM['F']['C'] = PAM['F']['Q'] = PAM['Q']['F'] = PAM['G']['F'] = PAM['F']['G'] = PAM['F']['P'] = PAM['P']['F'] = PAM['F']['T'] = PAM['T']['F'] = PAM['K']['W'] = PAM['W']['K'] = PAM['E']['Y'] = PAM['Y']['E'] = PAM['Y']['Q'] = PAM['Q']['Y'] = PAM['D']['Y'] = PAM['Y']['D'] = -4.0;
17	PAM['D']['C'] = PAM['C']['D'] = PAM['Q']['C'] = PAM['C']['Q'] = PAM['E']['C'] = PAM['C']['E'] = PAM['C']['K'] = PAM['K']['C'] = PAM['C']['M'] = PAM['M']['C'] = PAM['D']['F'] = PAM['F']['D'] = PAM['E']['F'] = PAM['F']['E'] = PAM['K']['F'] = PAM['F']['K'] = PAM['T']['W'] = PAM['W']['T'] = PAM['H']['W'] = PAM['W']['H'] = PAM['N']['W'] = PAM['W']['N'] = PAM['P']['Y'] = PAM['Y']['P'] = PAM['K']['Y'] = PAM['Y']['K'] = PAM['G']['Y'] = PAM['Y']['G'] = PAM['R']['Y'] = PAM['Y']['R'] = -5.0;
18	PAM['R']['Q'] = PAM['Q']['R'] = PAM['N']['Q'] = PAM['Q']['N'] = PAM['R']['E'] = PAM['E']['R'] = PAM['N']['E'] = PAM['E']['N'] = PAM['G']['A'] = PAM['A']['G'] = PAM['D']['G'] = PAM['G']['D'] = PAM['R']['H'] = PAM['H']['R'] = PAM['H']['N'] = PAM['N']['H'] = PAM['D']['H'] = PAM['H']['D'] = PAM['Q']['I'] = PAM['N']['K'] = PAM['K']['N'] = PAM['K']['Q'] = PAM['Q']['K'] = PAM['K']['M'] = PAM['M']['K'] = PAM['I']['F'] = PAM['F']['I'] = PAM['A']['P'] = PAM['P']['A'] = PAM['R']['S'] = PAM['S']['R'] = PAM['D']['S'] = PAM['S']['D'] = PAM['C']['S'] = PAM['S']['C'] = PAM['E']['S'] = PAM['S']['E'] = PAM['K']['S'] = PAM['S']['K'] = PAM['S']['T'] = PAM['T']['S'] = PAM['F']['W'] = PAM['W']['F'] = PAM['C']['Y'] = PAM['Y']['C'] = 1.0;
19	PAM['E']['D'] = PAM['D']['E'] = PAM['H']['Q'] = PAM['Q']['H'] = PAM['R']['K'] = PAM['K']['R'] = 3.0;
20	PAM['C']['L'] = PAM['L']['C'] = PAM['W']['P'] = PAM['P']['W'] = PAM['M']['W'] = PAM['W']['M'] = PAM['I']['W'] = PAM['W']['I'] = PAM['W']['Q'] = PAM['Q']['W'] = PAM['A']['W'] = PAM['W']['A'] = PAM['W']['V'] = PAM['V']['W'] = -6.0;
21	PAM['L']['W'] = PAM['W']['L'] = PAM['G']['W'] = PAM['W']['G'] = PAM['E']['W'] = PAM['W']['E'] = PAM['C']['W'] = PAM['W']['C'] = PAM['D']['W'] = PAM['W']['D'] = -7.0;
22	PAM['F']['Y'] = PAM['Y']['F'] = 7.0;
23	//KONVERSI DATA : membandingkan data latih dan data uji terhadap data wild menggunakan matriks PAM
24	//KONVERSI DATA LATIH
25	Double[][] konvLatih = new Double[dataLatih.length][393];
26	for (int i = 0; i < konvLatih.length; i++) {
27	for (int j = 0; j < konvLatih[0].length;

```

28      j++) {
29          String latih = dataLatih[i];
          konvLatih[i][j] =
PAM[latih.charAt(j)][dataWild.charAt(j)];
30      }
31  }
32
33
34      //KONVERSI DATA UJI
35      Double[][] konvUji = new
Double[dataUji.length][393];
36      for (int i = 0; i < konvUji.length; i++) {
37          for (int j = 0; j < konvUji[0].length;
j++) {
38              String uji = dataUji[i];
39              konvUji[i][j] =
PAM[uji.charAt(j)][dataWild.charAt(j)];
40          }
41      }
42

```

Berikut ini adalah penjelasan Kode Program 5.1:

1. Baris 1-22 membangun matriks *PAM* dengan tipe data indeks *Array* berupa *char* dan berisi angka-angka dengan tipe data *double*.
2. Baris 25 menginisialisasi variabel *Array konvLatih* bertipe data *double* dengan panjang baris sebanyak baris data latih dan panjang kolom sebanyak 393.
3. Baris 26 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
4. Baris 27 menginisialisasi perulangan terhadap variabel *j* dimulai dengan *j* bernilai 0.
5. Baris 28 data latih ke-*i* dimasukkan ke dalam variabel *latih* bertipe data *String*.
6. Baris 29 karakter ke-*j* dari variabel *latih* dijadikan indeks baris data *PAM* dan karakter ke-*j* data *wild* dijadikan indeks kolom data *PAM* sehingga nilai dari data *PAM* pada indeks tersebut dimasukkan ke dalam variabel *Array konvLatih* pada indeks baris ke-*i* dan indeks kolom ke-*j*.
7. Baris 35 menginisialisasi variabel *Array konvUji* bertipe data *double* dengan panjang baris sebanyak baris data uji dan panjang kolom sebanyak 393.
8. Baris 36 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
9. Baris 37 menginisialisasi perulangan terhadap variabel *j* dimulai dengan *j* bernilai 0.
10. Baris 38 data uji ke-*i* dimasukkan ke dalam variabel *uji* bertipe data *String*,
11. Baris 39 karakter ke-*j* dari variabel *uji* dijadikan indeks baris data *PAM* dan karakter ke-*j* data *wild* dijadikan indeks kolom data *PAM* sehingga nilai dari

data *PAM* pada indeks tersebut dimasukkan ke dalam variabel *Array konvUji* pada indeks baris ke-*i* dan indeks kolom ke-*j*.

5.1.2 Kode program *k-Nearest Neighbor* (k-NN)

Setelah didapatkan hasil konversi data latih dan data uji, data tersebut digunakan untuk proses selanjutnya yaitu proses perhitungan k-NN. Proses implementasi k-NN terdiri dari beberapa proses seperti proses menghitung jarak, mengurutkan jarak terkecil serta mengambil *k* jarak terkecil.

5.1.2.1 Proses Menghitung Jarak

Proses menghitung jarak dilakukan dengan menggunakan rumus *Euclidean Distance* seperti yang ditunjukkan pada Persamaan 2.2. Proses menghitung jarak dilakukan untuk menghitung jarak data uji terhadap data latih yang telah dikonversi menggunakan matriks *PAM*.

Kode Program 5.2 Proses menghitung jarak data uji terhadap data latih

Baris	Kode Sumber
1	Double[] jarak = new Double[konvLatih.length];
2	for (int i = 0; i < konvLatih.length; i++)
3	{
4	double total = 0.0;
5	for (int j = 0; j <
6	konvLatih[0].length; j++) { total += Math.pow((konvUji[j] -
7	konvLatih[i][j]), 2); }
8	Jarak[i]=Math.sqrt(total);
9	}

Berikut adalah penjelasan kode program 5.2:

1. Baris 1 menginisialisasi variabel *Array jarak* bertipe data *double* dengan panjang sebesar panjang baris *Array konvLatih*.
2. Baris 2 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
3. Baris 4 menginisialisasi variabel *total* bertipe data *double* bernilai 0
4. Baris 5 menginisialisasi perulangan terhadap variabel *j* dimulai dengan *j* bernilai 0.
5. Baris 6 menjumlahkan hasil kuadrat dari selisih nilai *konvUji* indeks ke-*j* dengan nilai *konvLatih* indeks baris ke-*i* dan indeks kolom ke-*j* kemudian dimasukkan ke dalam variabel *total*
6. Baris 7 batas perulangan terhadap variabel *j*
7. Baris 8 nilai *total* diakarkan dan dimasukkan ke dalam variabel *Array jarak* pada indeks ke-*i*.

5.1.2.2 Proses Mengurutkan Jarak Terkecil

Hasil jarak yang didapatkan dari proses perhitungan jarak kemudian diurutkan. Proses mengurutkan jarak terkecil dilakukan untuk menampilkan jarak dimulai dari jarak terkecil hingga jarak terbesar. Jarak terkecil nantinya digunakan dalam proses klasifikasi data uji. Kelas data latih juga dimasukkan ke dalam proses ini agar urutan kelas data latih sesuai.

Kode Program 5.3 Proses mengurutkan jarak terkecil

Baris	Kode Sumber
1	<code>for (int i = 0; i < jarak.length; i++) {</code>
2	<code> for (int j = 0; j < jarak.length - 1;</code>
3	<code> j++) {</code>
4	<code> if (jarak[j] > jarak[j + 1]) {</code>
5	<code> double temp = jarak[j];</code>
6	<code> String temp1 = dataKelas[j];</code>
7	<code> jarak[j] = jarak[j + 1];</code>
8	<code> dataKelas[j] = dataKelas[j +</code>
9	<code> 1];</code>
10	<code> jarak[j + 1] = temp;</code>
11	<code> dataKelas[j + 1] = temp1;</code>
12	<code> } } }</code>

Berikut adalah penjelasan Kode Program 5.3:

1. Baris 1 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
2. Baris 2 menginisialisasi perulangan terhadap variabel *j* dimulai dengan *j* bernilai 0.
3. Baris 3-9 melakukan percabangan terhadap kondisi benar *jarak* ke-*j* lebih besar dari *jarak* ke-*j*+1, maka *jarak* ke-*j* dimasukkan ke variabel *temp* sebagai variabel penyimpan nilai sementara, *dataKelas* ke-*j* dimasukkan ke variabel *temp1* yang bertipe data sama dengan variabel *temp* yaitu *double*. Nilai variabel *jarak* ke-*j*+1 dimasukkan ke *jarak* ke-*j* dan *dataKelas* ke-*j*+1 dimasukkan ke *dataKelas* ke-*j*. Nilai variabel *temp* dimasukkan ke variabel *jarak* ke-*j*+1 dan *temp1* dimasukkan ke variabel *dataKelas* ke-*j*+1. Jika kondisi salah, maka proses tidak menjalankan apapun dan kembali pada perulangan sampai perulangan selesai.

5.1.2.3 Proses Mengambil *k* Jarak Terkecil

Setelah hasil jarak diurutkan, diambil *k* jarak terkecil.

Kode Program 5.4 Proses mengambil *k* jarak terkecil

Baris	Kode Sumber
1	<code>int k = Integer.parseInt(tf_nilaiK.getText());</code>
2	<code>Double[] kJarak = new Double[k];</code>
3	<code>String[] kelask = new String[k];</code>
4	<code>for (int i = 0; i < k; i++) {</code>

5	<code>kJarak[i] = jarak[i];</code>
6	<code>kelask[i] = dataKelas[i];</code>
7	<code>}</code>

Berikut penjelasan Kode Program 5.4:

1. Baris 1 menginisialisasi variabel *k* bertipe data *integer* dengan mengkonversi nilai bertipe *String* yang dimasukkan melalui *TextField* *tf_nilaiK*
2. Baris 2 menginisialisasi variabel *Array kJarak* sepanjang *k* bertipe data *double*.
3. Baris 3 menginisialisasi variabel *Array kelask* sepanjang *k* bertipe data *String*.
4. Baris 4 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
5. Baris 5 Nilai variabel jarak ke-*i* dimasukkan ke variabel *kJarak* ke-*i*.
6. Baris 6 Nilai variabel data kelas ke-*i* dimasukkan ke variabel *kelask* ke-*i*.

5.1.3 Kode program Fuzzy k-Nearest Neighbor (Fk-NN)

Setelah didapatkan hasil *k* jarak terkecil, dilakukan proses implementasi Fk-NN. Dalam prose Fk-NN, dilakukan perhitungan derajat keanggotaan setiap kelas pada masing-masing data uji.

Kode Program 5.5 Proses menghitung derajat keanggotaan

Baris	Kode Sumber
1	<code>Double[] dkeanggotaan = new Double[4];</code>
2	<code>String setKelas[] = {"0", "1", "2", "3"};</code>
3	<code>double pembilang = 0;</code>
4	<code>double penyebut = 0;</code>
5	<code>int m = 4;</code>
6	<code>double pangkat = -2 / (m - 1);</code>
7	<code>for (int i = 0; i < setKelas.length; i++)</code>
8	<code>{</code>
9	<code>for (int j = 0; j < kJarak.length;</code>
10	<code>j++) {</code>
11	<code>penyebut += Math.pow(kJarak[j],</code>
12	<code>pangkat);</code>
13	<code>if (kelask[j].equals(setKelas[i]))</code>
14	<code>{</code>
15	<code>pembilang</code>
16	<code>Math.pow(kJarak[j], pangkat);</code>
17	<code>}</code>
	<code>dkeanggotaan[i] = pembilang /</code>
	<code>penyebut;</code>
	<code>}</code>

Berikut penjelasan Kode Program 5.5:

1. Baris 1 menginisialisasi variabel *Array dKeanggotaan* sepanjang 4 bertipe data *double*.
2. Baris 2 menginisialisasi isi variabel *Array setKelas* bertipe data *double*.

3. Baris 3 menginisialisasi nilai 0 variabel *pembilang* bertipe data *double*.
4. Baris 4 menginisialisasi nilai 0 variabel *penyebut* bertipe data *double*.
5. Baris 5 menginisialisasi nilai 4 variabel *m* bertipe data *int*.
6. Baris 6 menginisialisasi hasil perhitungan $-2/(m-1)$ ke variabel *pangkat* bertipe data *double*.
7. Baris 7 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
8. Baris 8 menginisialisasi perulangan terhadap variabel *j* dimulai dengan *j* bernilai 0.
9. Baris 10 mengisi nilai variabel *penyebut* dengan menjumlahkan hasil pangkat *pangkat* nilai variabel *kJarak* ke-*i* dengan nilai variabel *penyebut* sebelumnya
10. Baris 11-15 melakukan percabangan terhadap kondisi benar isi *kelas* ke-*j* sama dengan isi *setKelas* ke-*i*, mengisi nilai variabel *pembilang* dengan menjumlahkan hasil pangkat *pangkat* nilai variabel *kJarak* ke-*i* dengan nilai variabel *pembilang* sebelumnya. Jika kondisi salah, maka tidak dijalankan proses apapun dan kembali ke perulangan sampai batas perulangan tercapai.
11. Baris 16 memasukkan hasil proses perulangan variabel *j* berupa hasil perhitungan nilai variabel *pembilang* dibagi dengan nilai variabel *penyebut* ke variabel *dKeanggotaan* ke-*i*.

5.1.4 Implementasi perhitungan tingkat akurasi

Setelah didapatkan kelas klasifikasi masing-masing data uji yang diujikan, selanjutnya kelas tersebut dibandingkan dengan kelas data uji yang ada di dataset. Hasil tingkat akurasi didapatkan dengan menggunakan persamaan yang ditunjukkan pada Persamaan 2.3.

Kode Program 5.6 Proses perhitungan tingkat akurasi

Baris	Kode Sumber
1	for (int i = 0; i < keakuratan.length; i++) {
2	if
3	(kelasUji[i].equals(kelasKlasifikasi[i])) {
4	keakuratan[i] = "Akurat";
5	} else {
6	keakuratan[i] = "Tidak Akurat";
7	}
8	}
9	
10	double akurasi;
11	int total = 0;
12	for (int i = 0; i < keakuratan.length; i++) {
13	if (keakuratan[i].equals("Akurat")) {
14	total += 1;
15	}
16	}

17	<code>akurasi = total / keakuratan.length * 100;</code>
----	---

Berikut penjelasan Kode Program 5.5:

1. Baris 1 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
2. Baris 2-8 melakukan percabangan terhadap kondisi benar isi *kelasUji* ke-*i* sama dengan isi *kelasKlasifikasi* ke-*i*, mengisi nilai variabel *String keakuratan* ke-*i* dengan "Akurat". Jika kondisi salah, maka mengisi nilai variabel *String keakuratan* ke-*i* dengan "Tidak Akurat".
3. Baris 10 menginisialisasi variabel *akurasi* bertipe data *double*.
4. Baris 11 menginisialisasi variabel *total* bernilai 0 bertipe data *int*.
5. Baris 12 menginisialisasi perulangan terhadap variabel *i* dimulai dengan *i* bernilai 0.
6. Baris 13-16 melakukan percabangan terhadap kondisi benar isi *keakuratan* ke-*j* adalah "Akurat", mengisi nilai variabel *total* dengan menjumlahkan 1 dengan nilai variabel *total* sebelumnya. Jika kondisi salah, maka tidak dijalankan proses apapun dan kembali ke perulangan sampai batas perulangan tercapai.
7. Baris 17 memasukkan hasil nilai variabel *total* dibagi panjang *Array keakuratan* dikali 100 ke variabel *akurasi*.

5.2 Implementasi antarmuka

Berdasarkan perancangan yang telah dibuat di bab 4, berikut merupakan hasil implementasi antarmuka:

5.2.1 Memasukkan nilai k dan $fold$

Gambar 5.1 Antarmuka *input* nilai k dan $fold$

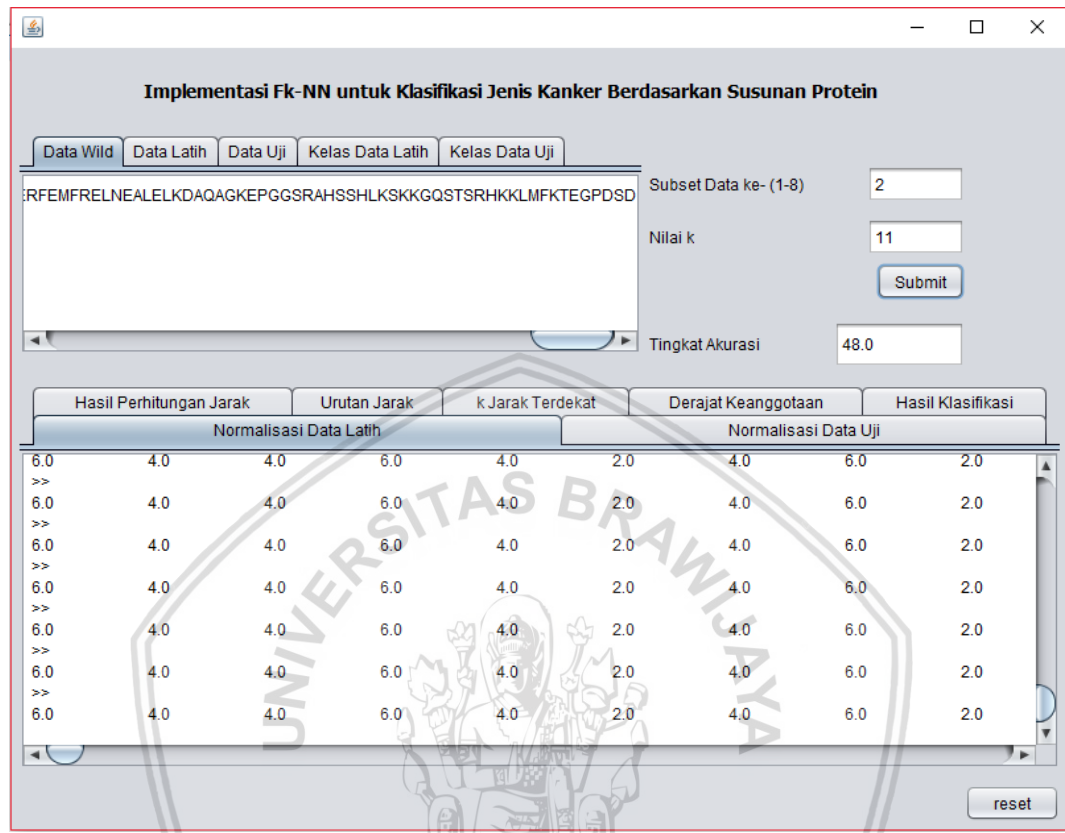
Antarmuka di atas menunjukkan pengisian 2 *TextField* dengan pada label *fold* dan nilai k seperti contoh 2 dan 11. Setelah itu dapat diklik *button* Submit.

5.2.2 Menampilkan data-data

Data-data yang digunakan pada implementasi ditampilkan melalui tab-tab yang berada dalam 1 *JTabbedPane* dalam *Jframe Java*. Tab-tab tersebut antara lain tab Data Wild, Data Latih, Data Uji, Kelas Data Latih, Kelas Data Uji.

1. Tab data wild digunakan untuk menampilkan data wild yang digunakan dalam implementasi yang berisi 1 baris sekuens protein normal ditunjukkan pada Gambar 5.2.
2. Tab Data Latih digunakan untuk menampilkan data latih yang digunakan dalam implementasi yang berisi 658 baris data ditunjukkan pada Gambar 5.3.
3. Tab Data Uji digunakan untuk menampilkan data uji yang digunakan untuk menampilkan data uji yang digunakan dalam implementasi yang berisi 94 baris data ditunjukkan pada Gambar 5.4.
4. Tab Kelas Data Latih digunakan untuk menampilkan 658 baris kelas data latih 0-4 yang digunakan dalam implementasi ditunjukkan pada Gambar 5.5.

5. Tab Kelas Data Uji digunakan untuk menampilkan 94 baris kelas data uji 0-4 yang digunakan dalam implementasi ditunjukkan pada Gambar 5.6.



Gambar 5.2 Tab data wild

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih			Normalisasi Data Uji	
Hasil Perhitungan Jarak	Urutan Jarak	k Jarak Terdekat	Derajat Keanggotaan	Hasil Klasifikasi
Data Uji	Derajat Keanggotaan	Kelas	Keakuratan	
0	0.363636363636365	0	Tidak Akurat	
1	0.454545454545453	0	Tidak Akurat	
2	0.545454545454544	0	Akurat	
3	0.72727272727273	0	Akurat	
4	0.63636363636364	0	Tidak Akurat	
5	0.545454545454544	0	Tidak Akurat	
6	0.545454545454544	0	Tidak Akurat	
7	0.363636363636365	0	Akurat	
8	0.545454545454544	0	Akurat	
9	0.63636363636364	0	Akurat	
10	0.454545454545453	0	Tidak Akurat	
11	0.72727272727273	0	Tidak Akurat	
12	0.545454545454544	0	Akurat	

reset

Gambar 5.3 Tab data latih

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT
MEEPQSDPSVEPPLSQETFSDLWKLLPENNVLSPLSQAMDDLMLSPDDIEQWFT

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih			Normalisasi Data Uji	
Hasil Perhitungan Jarak	Urutan Jarak	k Jarak Terdekat	Derajat Keanggotaan	Hasil Klasifikasi
Data Uji	Derajat Keanggotaan	Kelas	Keakuratan	
0	0.363636363636365	0	Tidak Akurat	
1	0.454545454545453	0	Tidak Akurat	
2	0.545454545454544	0	Akurat	
3	0.72727272727273	0	Akurat	
4	0.63636363636364	0	Tidak Akurat	
5	0.545454545454544	0	Tidak Akurat	
6	0.545454545454544	0	Tidak Akurat	
7	0.363636363636365	0	Akurat	
8	0.545454545454544	0	Akurat	
9	0.63636363636364	0	Akurat	
10	0.454545454545453	0	Tidak Akurat	
11	0.72727272727273	0	Tidak Akurat	
12	0.545454545454544	0	Akurat	

reset

Gambar 5.4 Tab Data Uji

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih			Normalisasi Data Uji	
Hasil Perhitungan Jarak	Urutan Jarak	k Jarak Terdekat	Derajat Keanggotaan	Hasil Klasifikasi
Data Uji	Derajat Keanggotaan	Kelas	Keakuratan	
0	0.363636363636365	0	Tidak Akurat	
1	0.454545454545453	0	Tidak Akurat	
2	0.545454545454544	0	Akurat	
3	0.727272727272723	0	Akurat	
4	0.636363636363636	0	Tidak Akurat	
5	0.545454545454544	0	Tidak Akurat	
6	0.545454545454544	0	Tidak Akurat	
7	0.363636363636365	0	Akurat	
8	0.545454545454544	0	Akurat	
9	0.636363636363636	0	Akurat	
10	0.454545454545453	0	Tidak Akurat	
11	0.727272727272723	0	Tidak Akurat	
12	0.545454545454544	0	Akurat	

reset

Gambar 5.5 Tab kelas data latih

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih			Normalisasi Data Uji	
Hasil Perhitungan Jarak	Urutan Jarak	k Jarak Terdekat	Derajat Keanggotaan	Hasil Klasifikasi
Data Uji	Derajat Keanggotaan	Kelas	Keakuratan	
0	0.363636363636365	0	Tidak Akurat	
1	0.454545454545453	0	Tidak Akurat	
2	0.545454545454544	0	Akurat	
3	0.727272727272723	0	Akurat	
4	0.636363636363636	0	Tidak Akurat	
5	0.545454545454544	0	Tidak Akurat	
6	0.545454545454544	0	Tidak Akurat	
7	0.363636363636365	0	Akurat	
8	0.545454545454544	0	Akurat	
9	0.636363636363636	0	Akurat	
10	0.454545454545453	0	Tidak Akurat	
11	0.727272727272723	0	Tidak Akurat	
12	0.545454545454544	0	Akurat	

reset

Gambar 5.6 Tab kelas data uji

5.2.3 Menampilkan hasil perhitungan

Hasil perhitungan yang dihasilkan pada implementasi ditampilkan melalui tab-tab yang berada dalam 1 *JTabbedPane* dalam *Jframe Java*.

1. Tab Normalisasi Data Latih digunakan untuk menampilkan hasil konversi nilai data latih menggunakan Matriks PAM250 ditunjukkan pada Gambar 5.7.
2. Tab Normalisasi Data Uji digunakan untuk menampilkan hasil konversi nilai data uji menggunakan Matriks PAM250 ditunjukkan pada Gambar 5.8.
3. Tab Hasil Perhitungan Jarak digunakan untuk menampilkan tabel hasil perhitungan jarak data uji terhadap data latih ditunjukkan pada Gambar 5.9.
4. Tab Urutan Jarak digunakan untuk menampilkan tabel hasil mengurutkan jarak dari yang terkecil sampai terbesar ditunjukkan pada Gambar 5.10.
5. Tab k Jarak Terdekat digunakan untuk menampilkan tabel hasil mengambil k jarak terkecil berdasarkan hasil mengurutkan nilai jarak ditunjukkan pada Gambar 5.11.
6. Tab Derajat Keanggotaan digunakan untuk menampilkan tabel hasil menghitung drajat keanggotaan antar kelas ditunjukkan pada Gambar 5.12.
7. Tab Hasil Klasifikasi digunakan untuk menampilkan tabel hasil klasifikasi beserta keakuratan hasil klasifikasi ditunjukkan pada Gambar 5.13.

Normalisasi Data Latih					Normalisasi Data Uji				
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									
6.0	4.0	4.0	6.0	4.0	2.0	4.0	6.0	2.0	
>>									

Gambar 5.7 Tab normalisasi data latih



Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

RFEMFRELNEALELKDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKLMFKTEGPDSD

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih Normalisasi Data Uji

Hasil Perhitungan Jarak Urutan Jarak k Jarak Terdekat Derajat Keanggotaan Hasil Klasifikasi

Data Uji - Latih	Jarak	Kelas
0 - 0	0.0	1
0 - 1	0.0	3
0 - 2	0.0	0
0 - 3	6.0	2
0 - 4	6.0	0
0 - 5	6.0	0
0 - 6	6.0	0
0 - 7	6.0	0
0 - 8	6.0	3
0 - 9	6.0	2
0 - 10	6.0	3
0 - 11	6.0	0
0 - 12	6.0	2

reset

Gambar 5.10 Tab urutan jarak

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

RFEMFRELNEALELKDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKLMFKTEGPDSD

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih Normalisasi Data Uji

Hasil Perhitungan Jarak Urutan Jarak k Jarak Terdekat Derajat Keanggotaan Hasil Klasifikasi

Data Uji - Latih	Jarak	Kelas
0 - 0	0.0	1
0 - 1	0.0	3
0 - 2	0.0	0
0 - 3	6.0	2
0 - 4	6.0	0
0 - 5	6.0	0
0 - 6	6.0	0
0 - 7	6.0	0
0 - 8	6.0	3
0 - 9	6.0	2
0 - 10	6.0	3
1 - 0	0.0	1
1 - 1	0.0	3

reset

Gambar 5.11 Tab k jarak terdekat

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

RFEMFRELNEALELKDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKKLMFKTEGPDSD

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih Normalisasi Data Uji

Hasil Perhitungan Jarak Urutan Jarak k Jarak Terdekat Derajat Keanggotaan Hasil Klasifikasi

Data Uji - U	Derajat Keanggotaan
0 - 0	0.363636363636365
0 - 1	0.181818181818182
0 - 2	0.27272727272727
0 - 3	0.25
1 - 0	0.454545454545453
1 - 1	0.318181818181818
1 - 2	0.303030303030304
1 - 3	0.25
2 - 0	0.545454545454544
2 - 1	0.363636363636365
2 - 2	0.27272727272727
2 - 3	0.25
3 - 0	0.72727272727273

reset

Gambar 5.12 Tab derajat keanggotaan

Implementasi Fk-NN untuk Klasifikasi Jenis Kanker Berdasarkan Susunan Protein

Data Wild Data Latih Data Uji Kelas Data Latih Kelas Data Uji

RFEMFRELNEALELKDAQAGKEPGGSRAHSSHLKSKKGQSTSRHKKLMFKTEGPDSD

Subset Data ke- (1-8) 2

Nilai k 11

Submit

Tingkat Akurasi 48.0

Normalisasi Data Latih Normalisasi Data Uji

Hasil Perhitungan Jarak Urutan Jarak k Jarak Terdekat Derajat Keanggotaan Hasil Klasifikasi

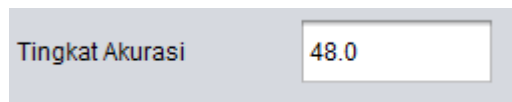
Data Uji	Derajat Keanggotaan	Kelas	Keakuratan
0	0.363636363636365	0	Tidak Akurat
1	0.454545454545453	0	Tidak Akurat
2	0.545454545454544	0	Akurat
3	0.72727272727273	0	Akurat
4	0.63636363636364	0	Tidak Akurat
5	0.545454545454544	0	Tidak Akurat
6	0.545454545454544	0	Tidak Akurat
7	0.363636363636365	0	Akurat
8	0.545454545454544	0	Akurat
9	0.63636363636364	0	Akurat
10	0.454545454545453	0	Tidak Akurat
11	0.72727272727273	0	Tidak Akurat
12	0.545454545454544	0	Akurat

reset

Gambar 5.13 Tab hasil klasifikasi

5.2.4 Menampilkan tingkat akurasi

Tingkat akurasi ditunjukkan pada *TextField* dengan label Tingkat Akurasi setelah *button Submit* dieksekusi. Tampilan tingkat akurasi ditunjukkan seperti Gambar 5.14.



The image shows a rectangular box with a light gray background. On the left side of the box, the text "Tingkat Akurasi" is written in a dark gray font. To the right of this text is a white rectangular text field with a thin gray border. Inside the text field, the number "48.0" is displayed in a dark gray font.

Gambar 5.14 *TextField* tingkat akurasi

Setelah dihasilkan tingkat akurasi, dapat digunakan *button reset* untuk mengeset *Jfame* kembali ke tampilan awal sebelum dieksekusi *button Submit*.



BAB 6 PEMBAHASAN DAN ANALISIS

Bab ini menjabarkan hasil implementasi berupa tingkat akurasi berdasarkan pengujian pengaruh nilai k pada *8-fold-validation* dan pengaruh variasi jumlah data latih dan uji dari hasil implementasi metode *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk klasifikasi jenis kanker berdasarkan susunan protein. Setelah didapatkan tingkat akurasi, dilakukan pembahasan serta analisis mengenai hasil pengujian.

6.1 Pengujian nilai k pada *8-fold-validation*

Pengujian ini melakukan perhitungan tingkat akurasi menggunakan *k-fold-validation* dengan nilai k yaitu 8. Dataset sejumlah 752 data dibagi menjadi 8 *fold* data. Masing-masing *fold* data berisi 94 data dengan kelas yang bervariasi terdiri dari 4 kelas yaitu kelas *Non Cancer* (NC) sebagai kelas 0, kelas *Breast Cancer* (BC) sebagai kelas 1, kelas *Collorectal Cancer* (CC) sebagai kelas 2 dan kelas *Lung Cancer* (LC) sebagai kelas 3. Setelah dataset telah terbagi menjadi 8 *fold*, dalam setiap pengujian digunakan salah satu *fold* sebagai data uji dan *fold-fold* lainnya digabung menjadi data latih. Pengujian menggunakan *k-fold-validation* ditujukan agar data uji yang digunakan untuk pengujian bervariasi sehingga sistem telah teruji ke data yang berbeda-beda dan dapat diketahui keakuratan sistem dalam melakukan klasifikasi. Nilai k pada *k-Nearest Neighbor* yang digunakan dalam pengujian adalah 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15.

Skenario pengujian yang dilakukan yaitu membagi data menjadi 8 *fold* data seperti yang telah dijelaskan. Pada saat program dijalankan, dapat dimasukkan nilai *fold* berupa angka 1 sampai 8 dan nilai k . Sebelum mengeluarkan hasil tingkat akurasi, program menjalankan perhitungan metode Fk-NN menggunakan data latih dan data uji sesuai dengan *fold* yang diinginkan. Tingkat akurasi yang dihasilkan dari implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein ditunjukkan pada Tabel 6.1.

Tabel 6.1 Tingkat akurasi pengaruh nilai k

Nilai k	Fold								Rata- Rata
	1	2	3	4	5	6	7	8	
1	41.50	31.91	35.11	29.79	34.04	43.62	32.98	42.55	36.44
2	42.55	44.68	37.23	38.30	38.30	44.68	48.94	48.94	42.95
3	42.55	44.68	43.62	37.23	39.36	47.87	50.00	54.26	44.95
4	43.62	46.81	46.81	38.30	44.68	50.00	48.94	52.13	46.41
5	45.74	50.00	41.50	38.36	44.68	43.62	46.81	47.87	54.99
6	46.81	50.00	45.75	41.50	43.62	46.81	46.81	50.00	46.41
7	45.75	45.75	45.75	41.50	46.81	47.87	47.87	52.18	46.69

Tabel 6.1 Tingkat akurasi pengaruh nilai k (Lanjutan)

Nilai k	Fold								Rata-rata
	1	2	3	4	5	6	7	8	
8	46.81	50.00	47.87	45.75	46.81	48.94	48.94	53.19	48.54
9	45.75	48.94	48.94	45.75	45.75	48.94	52.13	53.19	48.70
10	44.68	50.00	50.00	47.87	46.81	50.00	51.06	54.26	49.34
11	46.81	48.94	48.94	45.75	47.87	47.87	50.00	55.32	48.94
12	46.81	50.00	47.87	47.87	47.87	51.06	50.00	56.38	49.73
13	47.87	48.94	48.94	47.87	47.87	51.06	51.06	55.32	49.87
14	47.87	48.94	48.94	47.87	47.87	48.94	50.00	56.38	49.60
15	48.94	48.94	47.87	48.94	47.87	50.00	48.94	56.38	50.00
Rata-rata	45.60	47.24	45.68	42.84	44.48	48.10	48.30	52.56	47.57

Pengujian pengaruh nilai k dilakukan untuk mengetahui bagaimana kehandalan metode Fk-NN dalam mengklasifikasi jenis kanker berdasarkan susunan protein. Selain itu, dapat diketahui pula nilai k yang terbaik menghasilkan tingkat akurasi terbesar.

6.2 Pengujian pengaruh variasi jumlah data latih

Pengujian pengaruh variasi jumlah data latih dan data uji dilakukan dengan membagi jumlah data latih secara bervariasi yaitu masing 90%, 80%, 70%, 60% dari jumlah dataset. Data uji yang digunakan sebesar 20% dari jumlah dataset. Nilai k yang digunakan dalam pengujian ini adalah nilai k dengan tingkat akurasi terbaik dari pengujian sebelumnya.

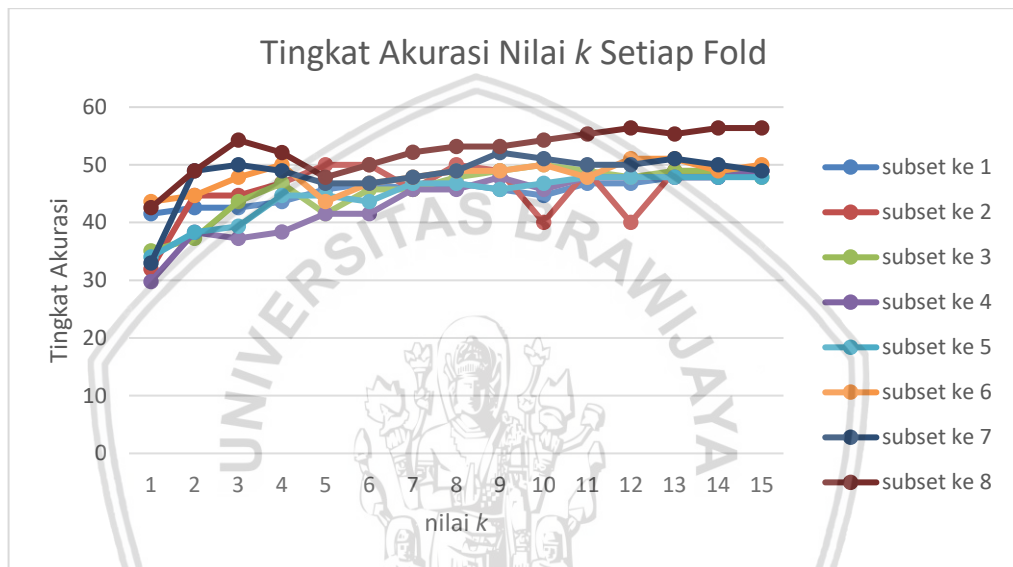
Skenario pengujian yaitu dataset dibagi menjadi 2 sebagai data latih dan data uji. Jumlah data latih dan data uji sesuai dengan persentase variasi jumlah data yang telah ditentukan. Kemudian dimasukkan nilai k optimal dari hasil pengujian sebelumnya yaitu 5 dengan tingkat akurasi 54.99%. Tingkat akurasi yang dihasilkan dari pengujian pengaruh variasi jumlah data latih dan data uji terhadap tingkat akurasi adalah sebagai berikut:

Tabel 6.2 Tingkat akurasi pengaruh variasi jumlah data latih dan data uji

Variasi jumlah data	60%	70%	80%	90%	Rata-rata
Tingkat akurasi	40.00	52.67	46.00	55.33	48.50

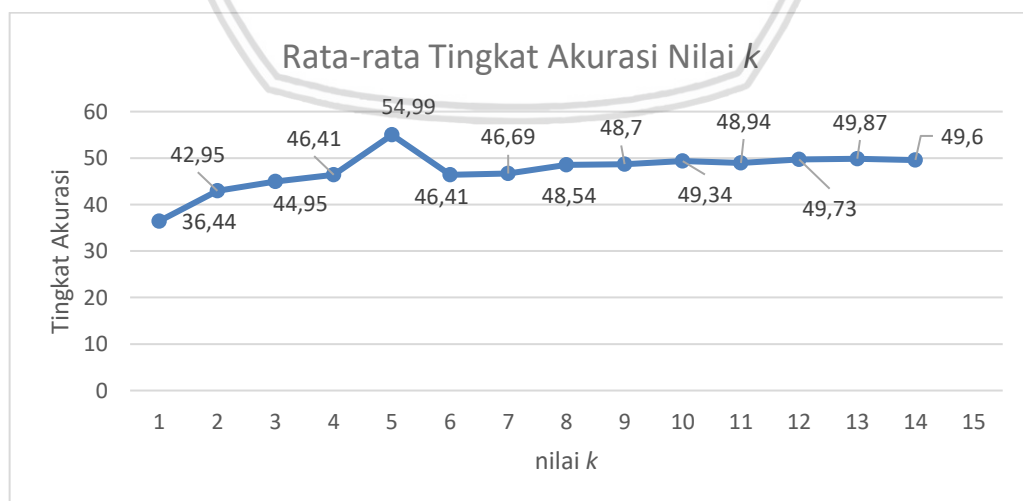
6.3 Analisis pengaruh nilai k pada 8-fold-validation

Berdasarkan Tabel 6.1, dapat diketahui bahwa implementasi metode Fk-NN menghasilkan tingkat akurasi yang berbeda-beda. Hal tersebut diketahui karena tingkat akurasi yang dihasilkan mengalami perubahan di beberapa nilai k yang dimasukkan. Namun, perubahan tingkat akurasi tidak tetap. Semakin besar nilai k tidak menjamin semakin besarnya tingkat akurasi. Contohnya dapat dilihat pada Tabel 6.1 *fold* ke 1 dan 2, tingkat akurasi naik sampai pada nilai k sama dengan 6. Pada nilai k sama dengan 7, tingkat akurasi turun dari 46.81% menjadi 45.75 dan dari 50.00% menjadi 45.75%. Pada *fold* lainnya, tingkat akurasi juga mengalami penurunan di nilai k tertentu.



Gambar 6.1 Grafik tingkat akurasi nilai k tiap *fold*

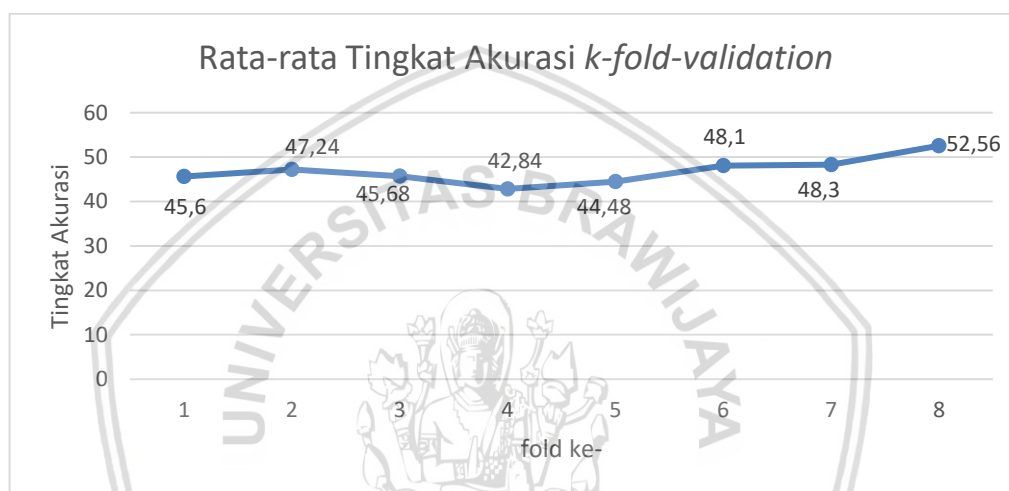
Tingkat akurasi tertinggi dihasilkan pada *fold* 8 dengan nilai k sama dengan 12, 14 dan 15 yaitu sebesar 56.38%.



Gambar 6.2 Grafik rata-rata tingkat akurasi nilai k

Pada Grafik 6.2 dapat diketahui rata-rata akurasi setiap nilai k pada keseluruhan tingkat akurasi yang dihasilkan di setiap *fold*. Tingkat akurasi tertinggi didapatkan pada nilai k sama dengan 5 dengan tingkat akurasi sebesar 54.99%. sedangkan rata-rata tingkat akurasi keseluruhan dari hasil pengujian nilai k pada 8-*fold-validation* yaitu sebesar 47.57%.

Semakin kecil nilai k menghasilkan tingkat akurasi yang lebih kecil dikarenakan jumlah tetangga terdekat yang terpilih lebih sedikit sehingga kurang cukup untuk menggambarkan kelas klasifikasi yang tepat. Sedangkan semakin besar nilai k menghasilkan tingkat akurasi yang lebih kecil dikarenakan jumlah tetangga terdekat yang terpilih lebih luas sehingga semakin banyak bias dalam menggambarkan kelas klasifikasi yang tepat.



Gambar 6.3 Grafik rata-rata tingkat akurasi k -fold

Pada Grafik 6.3 dapat diketahui rata-rata akurasi setiap *fold* pada keseluruhan tingkat akurasi yang dihasilkan di setiap nilai k . Tingkat akurasi rata-rata tertinggi didapatkan pada *fold* ke 8 dengan tingkat akurasi sebesar 52.56%. Dari hasil tersebut diketahui bahwa sistem hanya bisa mengklasifikasi sekitar setengah dari jumlah data uji yang masuk dengan akurat.

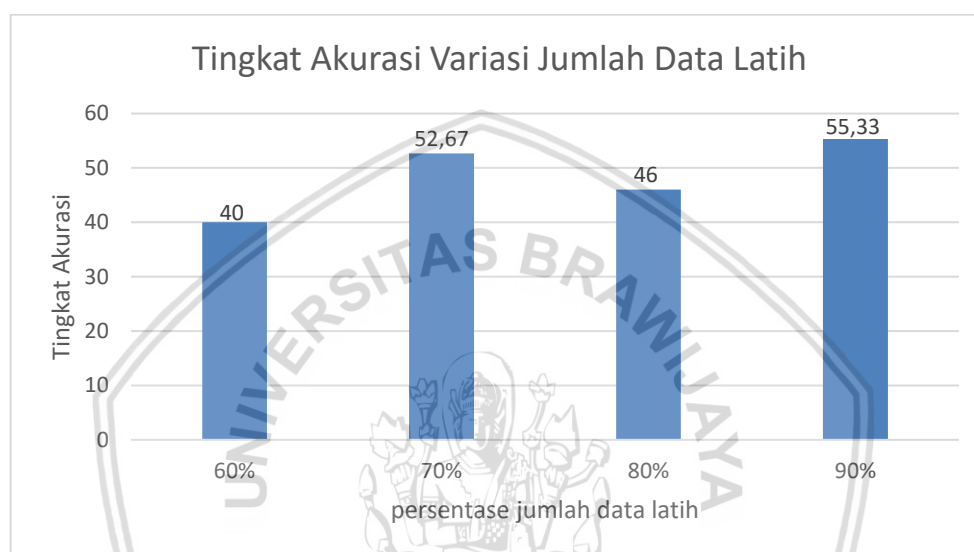
6.4 Analisis pengaruh variasi jumlah data latih

Berdasarkan Tabel 6.2, dapat diketahui bahwa variasi jumlah data latih mempengaruhi tingkat akurasi. Hal tersebut diketahui karena tingkat akurasi yang dihasilkan mengalami perubahan di beberapa variasi jumlah data latih yang ditentukan. Namun, perubahan yang terjadi tidak tetap. Tingkat akurasi tertinggi dihasilkan oleh jumlah data latih sebesar 90% sebesar 55.33%. Tingkat akurasi pada jumlah data latih yang paling besar lebih tinggi dibandingkan dengan tingkat akurasi pada jumlah data latih yang paling kecil.

Umumnya, jumlah data latih yang lebih besar menyebabkan terjadinya pembelajaran yang lebih baik sehingga memungkinkan tingkat akurasi akan semakin besar. Begitu juga sebaliknya, jumlah data latih yang semakin kecil menyebabkan pembelajaran yang kurang baik sehingga memungkinkan tingkat akurasi akan semakin kecil karena pengklasifikasian yang kurang tepat. Namun,

pada penelitian ini dihasilkan tingkat akurasi yang naik turun antar variasi jumlah data.

Hasil analisis lain terhadap pengaruh data terhadap tingkat akurasi yaitu oleh hasil konversi data yang sama antar satu data dengan data lainnya. Berdasarkan hasil konversi pada perhitungan manual yang ditunjukkan Tabel 4.2, dapat dilihat contohnya pada mutasi asam amino R menjadi C dengan kelas NC hasil konversinya sama dengan mutasi asam amino R menjadi G pada kelas yang berbeda yaitu LC sehingga dapat terjadi kesalahan klasifikasi.



Gambar 6.4 Grafik tingkat akurasi pengaruh variasi jumlah data latih

BAB 7 PENUTUP

Setelah dilakukan perancangan, implementasi dan pengujian hasil implementasi metode *Fuzzy k-Nearest Neighbor* (Fk-NN) untuk klasifikasi jenis kanker berdasarkan susunan protein, dapat ditarik kesimpulan dan saran. Bab ini menjabarkan kesimpulan dan saran dari hasil penelitian.

7.1 Kesimpulan

Berikut ini kesimpulan dari hasil penelitian implementasi metode Fk-NN untuk klasifikasi jenis kanker berdasarkan susunan protein.

1. Metode Fk-NN dinilai kurang cukup baik untuk diterapkan dalam klasifikasi jenis kanker berdasarkan susunan protein karena sistem hanya dapat mengklasifikasi sekitar setengah dari jumlah data uji dengan tepat berdasarkan rata-rata tingkat akurasi tertinggi yang dihasilkan berdasarkan pengujian menggunakan *8-fold-validation* sebesar 52.56% pada *fold* ke-8.
2. Nilai k terbaik untuk metode Fk-NN dalam klasifikasi jenis kanker berdasarkan susunan protein yaitu nilai $k = 5$ yang menghasilkan tingkat akurasi tertinggi sebesar 51.14%.
3. Variasi data latih mempengaruhi tingkat akurasi. Jumlah data latih paling besar (90% dari jumlah dataset) yang digunakan dalam pengujian pengaruh variasi data latih terhadap tingkat akurasi menghasilkan tingkat akurasi tertinggi yaitu sebesar 55.33%. Semakin besar data latih akan menghasilkan hasil pembelajaran yang lebih baik sehingga kelas klasifikasi lebih tepat.

7.2 Saran

Berdasarkan kesimpulan yang telah dijelaskan, berikut ini beberapa saran yang dapat digunakan dalam pelaksanaan penelitian sebelumnya.

1. Dalam melakukan klasifikasi jenis kanker berdasarkan susunan protein dapat digunakan metode lainnya agar dapat dihasilkan tingkat akurasi yang lebih baik.
2. Untuk penelitian selanjutnya, diharapkan dapat menganalisis kelas data dan menguji keseimbangan kelas data dalam setiap variasi jumlah data latih dan data uji.
3. Untuk penelitian selanjutnya, diharapkan dapat dilakukan perbaikan akurasi metode Fk-NN salah satunya dengan menggunakan metode *Fuzzy k-Nearest Neighbor in every Class* Fk-NNC.

DAFTAR PUSTAKA

- Candrarini, E., Soebroto, A. A. & Ridok, A., 2015. Sistem Pakar Diagnosa Penyakit Kulit pada Anak Menggunakan Metode Dempster-Sahfer. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Gorunescu, F., 2011. *Data Mining - Concepts, Models and Techniques*. Berlin: Springer - Verlag Berlin Heidelberg.
- Hermawati, F. A., 2013. *Data Mining*. Yogyakarta: Penerbit ANDI.
- Keller, J., Gray, M. & Givens, J., 1985. A Fuzzy k-Nearest Neighbor Algorithm. *IEEE Transactions on Systems Man and Cybernetics SMC-15(4)*.
- Kelvin, J. F. & Tyson, L. B., 2011. *100 Tanya Jawab mengenai Gejala Kanker dan Efek Samping Pengobatan Kanker*. 2nd penyunt. Jakarta: PT Indeks.
- King, R. J. B., 2000. *Cancer Biology*. 2nd penyunt. England: Pearson Education Limited.
- Klamerus, J. F., Brahmer, J. R. & Ettinger, D. S., 2012. *Patient's Guide to Lung Cancer*. Jakarta: PT. Indeks.
- Kurniati, R., M. & W., 2013. Penggunaan Metode Pengelompokan K-Means pada Klasifikasi KNN untuk Penentuan Jenis Kanker Berdasarkan Susunan Protein. *Repositori Jurnal Mahasiswa PTIIK UB*, Volume 1.
- Laxsmana, R. H., I. & Ratnawati, D. E., 2015. Pendeteksi Jenis Attention Deficit Hyperactivity Disorder (ADHD) pada Anak Usia Dini Menggunakan Metode Fuzzy k-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Nugraha, A., 2009. *Hubungan DNA, RNA, dan Protein*. [Online] Available at: <https://kantongkeresek.wordpress.com/2009/11/08/hubungan-dna-rna-dan-protein/> [Diakses 7 February 2018].
- Nugraha, S. D., Putri, R. R. M. & Wihandika, R. C., 2017. Penerapan Fuzzy K-Nearest Neighbor (FK-NN) Dalam Menentukan Status Gizi Balita. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Volume 1, pp. 925-932.
- Osterath, B., 2014. *Beginilah Terjadinya Kanker*. [Online] Available at: <http://www.dw.com/id/beginilah-terjadinya-kanker/a-17407704> [Diakses 7 February 2018].
- Pramadhani, A. E. & Setiadi, T., 2014. Penerapan Data Mining untuk Klasifikasi Prediksi Penyakit ISPA (Infeksi Saluran Pernapasan Akut) dengan Algoritma Decision Tree (ID3). *Jurnal Sarjana Teknik Informatika*, Volume 2.

- Prasetyo, E., 2012. *DATA MINING - Konsep dan Aplikasi Menggunakan MATLAB*. Yogyakarta: Penerbit ANDI.
- Puspitasari, D., Hidayat, N. & Marji, 2015. Pemodelan Sistem Pakar Diagnosa Penyakit Tanaman Cabai Merah dengan Metode Fuzzy k-Nearest Neighbor (Fk-NN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Retwitasari, A., M. & W., 2016. Penentuan Jenis Kanker Berdasarkan Struktur Protein Menggunakan Algoritma Modified k-Nearest Neighbor (MKNN). *Repositori Jurnal Mahasiswa PTIIK UB*, Volume 7.
- Rizby, L. P., M. & Muflikhah, L., 2018. Clustering Pasien Kanker Berdasarkan Struktur Protein Dalam Tubuh Menggunakan Metode K-Medoids. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, Volume 2, pp. 3810-3816.
- R., Soebroto, A. A. & Ridok, A., 2015. Sistem Pakar Diagnosa Penyakit pada Anak Menggunakan Metode Fuzzy k-Nearest Neighbor. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Setiati, E., 2009. *Waspada! 4 Kanker Ganas Pembunuh Wanita*. Yogyakarta: C.V. Andi Offset.
- Srimuddawamah, I., Soebroto, A. A. & Ridok, A., 2015. Sistem Pakar Diagnosa Penyakit Kulit pada Anak Menggunakan Metode Naive Bayes. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Sudiana, I. K., 2008. *Patobiologi Molekuler Kanker*. Jakarta Selatan: Salemba Medika.
- Syahiwa, N., Ratnawati, D. E. & Arwan, A., 2017. Implementasi Fuzzy k-Nearest Neighbor untuk Diagnosis Penyakit Tiroid. *Repositori Jurnal Mahasiswa PTIIK UB*, Volume 9.
- Wiraga, S., Hidayat, N. & Ratnawati, D. E., 2015. Diagnosa Penyakit Tanaman Cabai Merah Menggunakan Metode Modified k-Nearest Neighbor (Mk-NN). *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*.
- Wulandari, T., 2018. *Klasifikasi Jenis Kanker Berdasarkan Susunan Protein Menggunakan Algoritma Naive Bayes*, Malang: s.n.
- Zhang, J., Niu, Y. & Nie, H., 2009. Web Document Classification Based on Fuzzy K-NN Algorithm. *International Conference on Computational Intelligence and Security*.