TUGAS AKHIR – SS141501

ALGORITMA GENETIKA UNTUK OPTIMASI PARAMETER PADA *SUPPORT VECTOR MACHINE* DAN *FUZZY SUPPORT VECTOR MACHINE* : KASUS KLASIFIKASI DATA *MICROARRAY* KANKER KOLON

ELOK FAIQOH
NRP 062116 4500 0005

Dosen Pembimbing
Irhamah, M.Si., Ph.D

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

TUGAS AKHIR – SS141501

ALGORITMA GENETIKA UNTUK OPTIMASI PARAMETER PADA *SUPPORT VECTOR MACHINE* DAN *FUZZY SUPPORT VECTOR MACHINE* : KASUS KLASIFIKASI *MICROARRAY* DATA KANKER KOLON

ELOK FAIQOH
NRP 062116 4500 0005

Dosen Pembimbing
Irhamah, M.Si., Ph.D

PROGRAM STUDI SARJANA
DEPARTEMEN STATISTIKA
FAKULTAS MATEMATIKA, KOMPUTASI, DAN SAINS DATA
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

FINAL PROJECT – SS141501

GENETIC ALGORITHM FOR PARAMETER OPTIMIZATION IN SUPPORT VECTOR MACHINE AND FUZZY SUPPORT VECTOR MACHINE : CASE OF COLON CANCER MICROARRAY CLASSIFICATION

ELOK FAIQOH
SN 062116 4500 0005

Supervisor
Irhamah, M.Si., Ph.D

UNDERGRADUATE PROGRAMME
DEPARTMENT OF STATISTICS
FACULTY OF MATHEMATICS, COMPUTING, AND DATA SCIENCE
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA 2018

LEMBAR PENGESAHAN

ALGORITMA GENETIKA UNTUK OPTIMASI PARAMETER PADA *SUPPORT VECTOR MACHINE* DAN *FUZZY SUPPORT VECTOR MACHINE* : KASUS KLASIFIKASI *MICROARRAY* DATA KANKER KOLON

TUGAS AKHIR
Diajukan Untuk Memenuhi Salah Satu Syarat
Memperoleh Gelar Sarjana Sains
pada
Program Studi Sarjana Departemen Statistika
Fakultas Matematika, Komputasi, dan Sains Data
Institut Teknologi Sepuluh Nopember

Oleh :
Elok Faiqoh
NRP. 062116 4500 0005

Disetujui oleh Pembimbing:

Irhamah, M.Si., Ph.D ( )
NIP. 19780406 200112 2 002

Mengetahui,
Kepala Departemen

Dr. Suhartono
NIP. 19710929 199512 1 001

SURABAYA, JULI 2018

ALGORITMA GENETIKA UNTUK OPTIMASI PARAMETER PADA *SUPPORT VECTOR MACHINE* DAN *FUZZY SUPPORT VECTOR MACHINE* : KASUS KLASIFIKASI PADA DATA *MICROARRAY* KANKER KOLON

Nama Mahasiswa : Elok Faiqoh
NRP : 062116 4500 0005
Departemen : Statistika FMKSD ITS
Dosen Pembimbing : Irhamah, M.Si., Ph.D

Abstrak

Kanker usus besar adalah penyebab utama kedua kematian terkait kanker di dunia sehingga penelitian tentang kanker penting untuk dilakukan dalam upaya menurunkan jumlah penderita kanker kolon. Kemajuan terbaru dalam teknologi microarray memungkinkan pemantauan tingkat ekspresi dari sejumlah besar gen secara bersamaan. Data microarray adalah jenis data berdi-mensi tinggi dengan ratusan atau bahkan ribuan jumlah gen (fitur), sementara biasanya jumlah pasien yang diamati (peng-amatan) jauh lebih kecil daripada jumlah fitur. Penelitian ini akan menggunakan dataset microarray colon cancer yang berisi dua kelas gen, normal dan tumor. Tujuan dari penelitian ini adalah untuk mengembangkan model klasifikasi menggunakan fuzzy support vector machine (FSVM) hibridisasi dengan algoritma genetika (GA) untuk mengklasifikasikan individu berdasarkan ekspresi gen. Keanggotaan fuzzy akan diterapkan ke SVM untuk menangani kasus data microarray yang tidak seimbang. Sementara itu, peran algoritma genetika adalah untuk mengoptimalkan parameter SVM dan FSVM karena GA mampu menangani masalah optimasi nonlinier yang memiliki dimensi tinggi, mudah beradaptasi, dan mudah dikombinasikan dengan metode lain. Klasifikasi yang dilakukan adalah SVM dan FSVM baik grid search ataupun dengan optimasi GA. Metode klasifikasi menggunakan seleksi FCBF memiliki nilai akurasi yang lebih tinggi dibandingkan yang tanpa seleksi. Hasil penelitian menunjukkan bahwa FSVM yang telah dioptimasi menggunakan GA memiliki nilai akurasi tertinggi dibandingkan metode klasifikasi lainnya yang digunakan dalam penelitian ini.

Kata Kunci : Fuzzy SVM, Genetic Algorithm, Seleksi Variabel, SVM

(*Halaman ini sengaja dikosongkan*)

GENETIC ALGORITHM FOR PARAMETER OPTIMIZATION IN SUPPORT VECTOR MACHINE AND FUZZY SUPPORT VECTOR MACHINE : CASE OF COLON CANCER MICROARRAY CLASSIFICATION

Student Name : **Elok Faiqoh**
Student Number : **062116 4500 0005**
Department : **Statistics**
Supervisor : **Irhamah, M.Si., Ph.D**

Abstract

Colon cancer is the second leading cause of cancer-related deaths in the world hence research on that topic needs to be undertaken with improvement. Recent advancement in microarray technology allows the monitoring of the expression level of a large set of genes simultaneously. Microarray data is a type of high-dimensional data with hundreds or even thousands number of genes (features), while usually the number of patients observed (observations) is much smaller than the number of features. This study will use an open source colon cancer microarray dataset contains two class of genes, normal and tumor. The aims of this study is to develop a classification model using fuzzy support vector machine (FSVM) hybridized with genetic algorithm (GA) for classify individuals based on gene expression. Fuzzy memberships will be applied to SVM in order to dealing with the case of imbalanced microarray data. Meanwhile, the role of genetic algorithm is, firstly, to select the relevant genes as the features and, secondly, to optimize the parameter of FSVM as GA is able to handle the problem of nonlinear optimization that has a high dimension, adaptable, and easily combined with other methods. The classification is SVM and FSVM either grid search or GA optimization. The method of classification using FCBF selection has a higher accuracy value than the ones without the selection. The results show that FSVM that has been optimized using GA has the highest accuracy value compared to other classification methods used in this study.

Keywords **: *Feature Selection*, Fuzzy SVM, *Genetic Algorithm*, and SVM**

(This page intentionally left blank)

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas limpahan berkat rahmat, karunia, taufik, dan hidayahNya yang tidak pernah berhenti sehingga penulis dapat menyelesaikan penyusunan dengan baik Tugas Akhir yang berjudul "Algoritma Genetika Untuk Optimasi Parameter Pada *Support Vector Machine* dan *Fuzzy Support Vector Machine* : Kasus Klasifikasi Pada Data *Microarray* Kanker Kolon". Penulis menyadari bahwa dalam penyusunan Tugas Akhir ini tidak terlepas dari bantuan dan dukungan dari berbagai pihak. Oleh karena itu, pada kesempatan ini penulis mengucapkan terima kasih yang sebesar-besarnya kepada:

1. Ibu Irhamah, M.Si., Ph.D selaku dosen pembimbing yang telah dengan sukarela membimbing, memberikan motivasi dan informasi hingga penulis dapat menyelesaikan Tugas Akhir ini.
2. Ibu Wiwiek S. W., M.Si dan ibu Pratnya Paramitha O, M.Si selaku dosen penguji atas saran dan kritiknya yang membangun serta Bu Ni Luh Putu Satyaning, M.Sc yang senantiasa memberikan arahan dan wawasan tentang metode yang digunakan dalam Tugas Akhir ini.
3. Bapak Dr. Suhartono, M.Sc. selaku Ketua Jurasan Statistika ITS yang telah memberikan banyak fasilitas yang menunjang kelancaran penyelesaian Tugas Akhir.
4. Bapak Dr. Sutikno, S.Si., M.Si. selaku Ketua Prodi Sarjana Statistika ITS yang banyak membantu memberikan dukungan, bantuan, dan informasi kepada penulis sehingga dapat menyelesaikan tugas akhir ini
5. Bapak Prof. I Nyoman Budiantara selaku dosen wali yang selalu memberikan pengarahan dan motivasi selama perkuliahan penulis.

6. Seluruh civitas akademika Departemen Statistika FMKSD ITS yang telah membantu dalam melancarkan Tugas Akhir ini.
7. Bapak dan Ibu Sucipto atas segala doa, kasih sayang dan perjuangan nya sehingga penulis dapat menyelesaikan tugas akhir ini
8. Laily Badria, Nafisa Cahyani, Budi Cahyono, dan Raska Naufal yang selalu memberikan motivasi dan dukungan baik materi maupun non materi
9. Violita Pertiwi, Cicilia Ajeng, Jefri, Neni, Ageng sebagai rekan sesama *support vector machines* yang selalu membantu penulis dalam bertukar wawasan mengenai Tugas Akhir ini serta telah memberikan warna persahabatan dibangku perkuliahan yang singkat ini
10. Dyah Ayu, Sabella Dinna, Anggraeni Nur, Aprilia Dwi, Fabi'ayyi, dan Beti Kartika yang selalu mendengarkan keluh kesah penulis dan senantiasa memberikan semangat dalam mengerjakan tugas akhir
11. Teman-teman Lintas Jalur Statistika ITS 2016 khususnya teman selusin dan Semua pihak yang telah mendukung dan tidak dapat penulis sebutkan satu persatu.

Penulis sangat berharap hasil Tugas Akhir ini dapat bermanfaat bagi penulis pada khususnya dan pembaca pada umumnya. Penulis menyadari dalam penulisan laporan Tugas Akhir ini masih jauh dari sempurna. Oleh karena itu, peneliti mengharap adanya perbaikan dalam penulisan laporan di masa mendatang.

Surabaya, Juli 2018

Penulis

DAFTAR ISI

Halaman

DAFTAR GAMBAR

Halaman

(Halaman ini sengaja dikosongkan)

DAFTAR TABEL

Halaman

DAFTAR LAMPIRAN

Halaman

(*Halaman ini sengaja dikosongkan*)

BAB I
PENDAHULUAN

1.1 Latar Belakang

Kesehatan merupakan hal yang terpenting didalam hidup setiap orang. Namun setiap orang mempunyai sel kanker di dalam tubuh yang bisa sewaktu-waktu mengancam jiwa. Penyakit kanker merupakan salah satu penyebab kematian utama diseluruh dunia. Pada tahun 2012, sekitar 8,2 juta kematian disebabkan oleh kanker. Kanker paru, hati, perut, kolorektal, dan kanker payudara adalah penyebab terbesar kematian akibat kanker setiap tahunnya (Info Datin, 2015). Penyakit kanker sangatlah ganas karena akan menyerang pada pertahanan tubuh manusia dengan waktu yang sangat cepat. Sel-sel kanker tidak terlihat dalam tes standard hingga berkembang biak menjadi bermilyar-milyar. Ketika dokter mengatakan kepada pasien kanker bahwa tidak ada lagi sel kanker di tubuh mereka setelah perawatan, itu berarti bahwa tes yang dilakukan tidak mampu mendeteksi sel kanker karena sel kanker tersebut tidak sampai pada jumlah yang dapat diprediksi. *World Health Organization* (WHO) menyatakan bahwa diagnosa kanker di seluruh dunia diperkirakan akan menemukan 12 juta penderita setiap tahun. Sedangkan kematian akibat kanker secara global akan mencapai tujuh juta. Tren penyakit akan meningkat dua kali lipat dan lebih mematikan pada tahun 2030 dan kemungkinan 75 juta akan menderita penyakit kanker (Mujiarto, 2015).

National Cancer Institute (NCI) menyebutkan bahwa kanker usus besar atau kanker kolon adalah kanker kedua yang paling banyak merenggut nyawa. Kanker kolon akan tumbuh pada jaringan usus besar yang biasanya diawali dengan adanya gumpalan kecil sel jinak atau polip yang kemudian akan berubah menjadi kanker. Dalam pemeriksaan sangat dianjurkan untuk menemukan polip sebelum terjadi perubahan menjadi kanker. Kanker kolon diperkirakan membunuh lebih

dari 51.000 orang pada tahun 2010 dan jumlahnya akan meningkat setiap tahunnya (World, 2017). Kolon bekerja sebagai organ untuk menyimpan produk-produk limbah, mereabsorpsi air dari limbah dan menjaga keseimbangan air dalam tubuh. Selain ini, kolon berfungsi sebagai tempat bagi pertumbuhan bakteri menguntungkan dan mikroorganisme lain yang membantu memfermentasi bahan makanan serta membantu memelihara keseimbangan air dalam tubuh dan penyerapan beberapa vitamin penting dan elektrolit. Bakteri ramah yang berada pada kolon juga membantu untuk memeriksa pertumbuhan bakteri berbahaya, dan menjaga keseimbangan pH dalam tubuh, sehingga kolon atau usus besar merupakan salah satu bagian terpenting dari sistem pencernaan manusia (Indriani, 2018).

Diagnosis kanker dapat dilakukan berdasarkan struktur morfologisnya, namun menemui kesulitan karena perbedaan struktur morfologis yang sangat tipis antar jenis kanker yang berbeda (Golub, Slonim, Tamayo, Huard, Gaasenbeek, 1999). Pada tahun 1999, Alon melakukan penelitian mengenai ekpresi gen berkaitan dengan kanker kolon yang dimuat pada *microarray data*. *Microarray* data merupakan teknologi dalam bidang Biologi Molekuler dan Medis yang dapat digunakan untuk melihat perbedaan ekspresi gen. Informasi yang diperoleh dari hasil *microarray* telah dimanfaatkan untuk berbagai aplikasi spesifik seperti diagnosis penyakit, penemuan obat-obatan, pengelompokan ekspresi gen yang terlibat dalam organogenesis, rekaman dan interaksi dengan mikro organisme. *Microarray Data* merupakan jenis data yang dipakai dalam bioinformatika. Jenis data ini merupakan salah satu jenis data dengan dimensi yang sangat tinggi. Karakteristik *microarray data* adalah jumlah data sedikit dan jumlah *feature* yang sangat banyak. *Microarray data* terdiri dari ribuan spot (*feature*) dan dari masing-masing spot terdiri dari jutaan *copies* dari molekul DNA yang merespon ke suatu gen. Kumpulan-kumpulan gen akan digunakan untuk mengklasifikasikan ke dalam kelas suatu

penyakit (Babu, 2013). Oleh karena itu, penelitian dilakukan menggunakan *microarray data* kanker kolon yang akan dilakukan pengklasifikasian *gen* atau *features* dengan menggunakan metode *Support Vector Machines* (SVM) dan *Fuzzy Support Vector Machines* (FSVM).

SVM atau *Support Vector Machine* merupakan salah satu metode pengklasifikasian yang memberikan hasil terbaik (Chen, Lu dan Huang, 2009). SVM memiliki kemampuan yang lebih baik dalam klasifikasi daripada metode lain. Dengan adanya kemampuan generalisasi, SVM mampu menghasilkan akurasi yang tinggi dan tingkat kesalahan yang relatif kecil. Pada perkembangannya, SVM telah berhasil digunakan untuk menyelesaikan permasalahan dalam berbagai bidang, diantaranya adalah klasifikasi pada *microarray data* (Furey dkk, 2000). Namun pengaruh dari *imbalanced* data pada SVM akan menjadi kekurangan dalam paradigma memaksimalkan margin (Akbani dkk, 2004). Lin dan Wang (2002) mengusulkan *Fuzzy* SVM (FSVM) yang berlaku keanggotaan *fuzzy* untuk masing-masing sampel dan merumuskan SVM, sehingga *input sampel* yang berbeda memiliki kontribusi yang berbeda dan dapat menangani data *imbalanced.* Data *imbalanced* adalah sebuah data yang didominasi oleh kelas mayoritas atau kelas yang signifikan lebih banyak kejadiaan daripada kelas yang langka atau kelas minoritas (Canedo dkk, 2014).

Berbagai penelitian mengenai *microarray colon cancer* juga telah dilakukan dengan menggunakan *grid seacrh SVM*, performa klasifikasi yang diperoleh untuk klasifikasi menghasilkan akurasi sebesar 75%, sensitivitas sebesar 60%, spesifisitas sebesar 90%, *G-Mean* sebesar 73,8% dan AUC sebesar 75% (Kusumaningrum, 2017). Metode SVM dan K-NN (*K-Nearest Neighbor*) pernah dilakukan untuk klasifikasi berita *online* yang menghasilkan kesimpulan bahwa SVM kernel *polynomial* lebih baik dibandingkan dengan K-NN dengan menggunakan k=2 pada data *testing* dengan *world vector* (Aisyah, 2016). Penelitian dengan menggunakan meto-

de SVM dan K-NN juga pernah dilakukan untuk mengklasifikasikan *email spam* yang menghasilkan kesimpulan bahwa dengan menggunakan K-NN, penggunaan *k* berpengaruh terhadap ketepatan klasifikasi. Metode K-NN memberikan hasil pengukuran klasifikasi terbaik untuk semua *fold* saat k=3 dengan hasil ketepatan klasifikasi yang diberikan sebesar 92,28% dengan *error* sebesar 7.72% dari total 6000 *email*. Selain itu, dengan menggunakan metode SVM linier dan kernel RBF didapatkan bahwa penggunaan kernel yang berbeda juga mempengaruhi ketepatan klasifikasi. Jika dilakukan perbandingan metode K-NN dan SVM linier pada kasus ini, diketahui bahwa SVM dengan kernel linier memberikan hasil klasifikasi yang lebih baik dibandingkan dengan 3-NN (Pratiwi, 2016). Selain itu, *microarray data colon cancer* menggunakan metode SVM tingkat akurasi *Variabel selection* dengan CFS sebesar 95,42% dan FCBF sebesar 96,25% (Rusydina, 2016). Penelitian tentang klasifikasi citra satelit dengan menggunakan YUV FSVM diperoleh nilai akurasi sebesar 72,83% dan dengan menggunakan RGB FSVM nilai akurasi yang diperoleh adalah 72,25% (Supianto, 2013). Selain itu, penelitian mengenai FSVM juga pernah dilakukan untuk mengetahui klasifikasi *genre* musik berdasarkan Variabel audia dan diperoleh kesimpulan bahwa FSVM untuk klasifikasi genre musik memiliki akurasi yang lebih baik pada jumlah kelas yang lebih sedikit (Rata-rata akurasi pada jumlah kelas 5 adalah 40.8%, jumlah kelas 4 adalah 48,2% dan pada jumlah kelas 3 adalah 56,93%) (Cariadhi dkk, 2014).

Berdasarkan uraian tersebut maka dapat diketahui bahwa *fuzzy support vector machines* adalah salah satu metode yang dapat digunakan dalam klasifikasi data *imbalance.* Namun, selama ini, metode pengklasifikasian yang sering digunakan baik untuk data *balance* ataupun data *imbalance* adalah metode *support vector machines* karena nilai akurasi yang didapatkan lebih tinggi dibandingkan dengan metode klasifikasi lainnya. Oleh karena itu, pada penelitian ini akan dilakukan perban-

dingan analisis *Fuzzy Support Vector Machines* (FSVM) dan *support vector machines* (SVM) dengan seleksi variabel yaitu *fast correlation based filter* dan *genetic algorithm optimation* pada data *microarray* kanker kolon. *Genetic Algorithm* digunakan karena selain dapat digunakan karena dapat memiliki banyak kelebihan, antara lain mampu mengatasi berbagai jenis fungsi objektif dari berbagai kromosom, lebih adaptif dan mudah dikombinasikan dengan metode lain, mempunyai kemampuan untuk menangani permasalahan yang kompleks dan paralel, dapat digunakan untuk jumlah variabel yang besar, serta dapat menangani masalah optimasi nonlinear yang berdimensi tinggi. Penelitian ini akan membandingkan SVM maupun FSVM dengan menggunakan seleksi variabel dan tanpa menggunakan seleksi variabel. Hasil klasifikasi dinilai berdasarkan performa klasifikasi yang meliputi akurasi, sensitifitas, spesifisitas, *Gmeans*, dan AUC.

1.2 Rumusan Masalah

Microarray data merupakan data *high dimentional data* yang berisi ekspresi gen yang salah satunya adalah *microarray* kanker kolon. *Microarray data* kanker kolon adalah data ekpresi gen dari jaringan tumor dan normal dimana terdapat 62 observasi yang terdiri dari 40 observasi kelas tumor dan 22 observasi kelas normal dengan banyak variabel adalah 2000 variabel. Dominasi gen atau variabel bervariasi sehingga perlu diklasifikasikan dan metode yang sesuai adalah *Support Vector Machines* (SVM) dan *Fuzzy Support Vector Machines* (FSVM) dengan menggunakan seleksi variabel FCBF dan optimasi *genetic algorithm.*

1.3 Tujuan

Berdasarkan uraian rumusan masalah yang telah dijelaskan, maka tujuan yang ingin dicapai dalam penelitian ini adalah untuk mendapatkan hasil klasifikasi menggunakan metode *Support Vector Machine* (SVM) dan metode *Fuzzy*

Support Vector Machine (FSVM) dimana masing-masing metode akan dilakukan seleksi variabel menggunakan FCBF dan optimasi *genetic algorithm* untuk mendapatkan nilai akurasi dari masing-masing metode. Metode SVM akan dibagi menjadi SVM *grid search* dan SVM optimasi GA dimana masing-masing klasifikasi akan dibandingkan jika tanpa seleksi variabel dan dengan seleksi variabel FCBF. Begitu juga dengan metode FSVM dan FSVM optimasi GA yang akan membandingkan masing-masing metode menggunakan seleksi FCBF dan tanpa seleksi variabel.

1.4 Manfaat Penelitian

Manfaat yang diperoleh dari penelitian ini yaitu dapat membantu menyelesaikan permasalahan klasfikasi *high dimensional data* berupa *microarray data* kanker kolon. Manfaat lain yang diperoleh adalah mendapatkan hasil klasifikasi menggunakan metode SVM dan FSVM (*Fuzzy Support Vector Machine*) menggunakan seleksi variabel FCBF dan optimasi *genetic algorithm* yang dapat digunakan untuk mendeteksi kanker kolon sejak dini di Indonesia.

1.5 Batasan Masalah

Batasan masalah yang perlu diperhatikan dalam penelitian ini adalah seleksi variabel menggunakan FCBF dan optimasi parameter menggunakan *genetic algorithm* pada SVM dan *Fuzzy Supprot Vector Machines.* Selain itu, batasan lainnya yaitu kernel yang digunakan adalah kernel RBF (gaussian).

BAB II
TINJAUAN PUSTAKA

2.1 *Support Vector Machine*

Support Vector Machine (SVM) pertama kali diperkenalkan oleh Vapnik tahun 1992. SVM merupakan suatu teknik untuk menemukan fungsi pemisah dalam pengklasifikasian yang dapat memisahkan dua atau lebih kelompok data yang berbeda. Metode SVM merupakan metode *machine learning* yang bekerja atas prinsip *Structural Risk Minimazation* (SRM) dengan tujuan menemukan *hyperplane* terbaik yang memisahkan dua buah kelas pada *input space*. SVM dapat menemukan fungsi pemisah (*hyperplane*) terbaik diantara fungsi yang tidak terbatas jumlahnya untuk memisahkan obyek. *Hyperplane* terbaik terletak tepat di tengah antara dua set obyek dari dua kelas. Mencari *hyperplane* terbaik ekuivalen dengan memaksimalkan margin atau jarak antara dua set obyek dari dua kelas berbeda. SVM bekerja untuk menemukan suatu fungsi pemisah dengan margin yang maksimal (Vapnik, 1999). SVM merupakan teknik klasifikasi dengan proses pelatihan (*supervised learning*) untuk menemukan garis pemisah *hyperplane* terbaik dengan $f(x)$.

$$f(\mathbf{x}) = sign(g(\mathbf{x})) \tag{2.1}$$

dengan

$$g(\mathbf{x}) = \mathbf{x}_\mathbf{i}^T\mathbf{w} + b \tag{2.2}$$

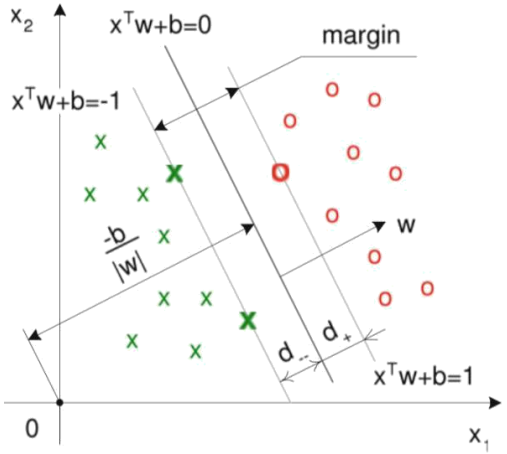
dimana $\boldsymbol{x}, w \in R^n$ dan $b \in R$

Apabila nilai g(x) negatif maka observasi masuk kedalam kelas negatif, begitupun jika g(x) bernilai positif maka observasi akan masuk kedalam kelas positif. Prinsip dasar SVM adalah *linear clasifier* yang kemudian dikembangkan agar dapat menyelesaikan permasalahan yang nolinear.

2.1.1 Klasifikasi Linear *Separable*

Klasifikasi Linier *separable* data adalah penerapan metode SVM pada data yang dapat dipisahkan secara linier.

Misal $x_i = \{x_i, x_{i+1}, \ldots, x_n\}$ adalah *dataset* dan $y_i = \{+1, -1\}$ adalah label kategori untuk *dataset*. Ilustrasi linier *separable* ditunjukkan oleh gambar 2.1.



Gambar 2.1 Konsep *Hyperplane* pada SVM (sumber : Hardle & Simar, 2015)

Pada gambar 2.1 margin sama dengan $d\text{-}+d_+$. Fungsi klasifikasi adalah *hyperplane* ditambah zona margin. Ini memisahkan poin dari kedua kelas dengan jarak "paling aman" tertinggi (margin) diantara kedua kelas. Pemisahan *hyperplane* didefinisikan hanya dengan *Support Vector* yang menahan *hyperplane* sejajar dengan pemisah.

$\mathbf{x_i}^T\mathbf{w} + b = 0$ adalah *hyperplane* pemisah, $d\text{-}+d_+$ akan menjadi jarak terpendek pada objek yang paling dekat dari kelas +1 (-1). Karena pemisahan dapat diselesaikan tanpa error, semua observasi *i = 1, 2, ..., n* harus memenuhi,

$$\begin{aligned} \mathbf{x_i}^T\mathbf{w} + b &\geq +1 \text{ untuk } y_i = +1 \\ \mathbf{x_i}^T\mathbf{w} + b &\geq -1 \text{ untuk } y_i = -1 \end{aligned} \qquad (2.3)$$

Canonical *hyperplane* $\mathbf{x_i}^T\mathbf{w} + b = \pm 1$ sejajar dan jarak antara masing-masing dan *hyperplanes* pemisah $d_+ = d_- = 1/\|\mathbf{w}\|$ dimana **w** adalah vector bobot (*weight vector*) yang berukuran

(*p x 1*), b adalah posisi bidang relatif terhadap pusat koordinat atau lebih dikenal dengan bias yang bernilai skalar.

$$X = \begin{bmatrix} x_{11} & x_{12} & \dots & x_{1p} \\ x_{21} & x_{22} & \dots & x_{2p} \\ \vdots & \vdots & \dots & \vdots \\ x_{n1} & x_{n2} & \dots & x_{np} \end{bmatrix} \quad \mathbf{x_i}^T = \begin{bmatrix} x_{i1} & x_{i2} & \cdots & x_{ip} \end{bmatrix}$$

$$\mathbf{w} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_p \end{bmatrix} \qquad y_i = \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix}$$

Gambar 2.1 menunjukkan $\frac{|b|}{\|\mathbf{w}\|}$ adalah jarak bidang pemisah yang tegak lurus dari titik pusat koordinat dan ||**w**|| adalah jarak *euclidean* (*norm euclidean*) dar **w**. Panjang vektor **w** adalah norm $\|\mathbf{w}\| = \sqrt{\mathbf{w}^T\mathbf{w}} = \sqrt{w_1^2 + w_2^2 + ... + w_p^2}$. Bidang batas pertama $\mathbf{x_i}^T\mathbf{w} + b = 1$ mempunyai bobot **w** dan jarak tegak lurus dari titik asal sebesar $\frac{|1-b|}{\|\mathbf{w}\|}$, sedangkan bidang pembatas kedua $\mathbf{x_i}^T\mathbf{w} + b = -1$ mempunyai bobot **w** dan jarak tegak lurus dari titik asal sebesar $\frac{|-1-b|}{\|\mathbf{w}\|}$. Nilai maksimum margin atau nilai jarak antar bidang pembatas adalah

$$\frac{1-b-(1-b)}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|} \qquad (2.4)$$

Secara sistematis, formulasi permasalahan optimasi SVM untuk klasifikasi *linear* dalam prial space adalah

$$\min \frac{1}{2}\|\mathbf{w}\|^2 \tag{2.5}$$

Persamaan (2.5) akan lebih mudah diselesaikan jika diubah kedalam formula *lagrange*. Lagrangian dari *primal problem* yang berhubungan dengan maksimalisasi margin adalah,

$$\min_{w,b} L_p(w,b) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \left[y_i \left(\mathbf{x_i}^T \mathbf{w} + b \right) - 1 \right] \tag{2.6}$$

Karush-Kuhn-Tucker (KTT) (Gale dkk, 1951) kondisi optimal order pertama adalah,

$$\begin{aligned}
&\frac{\partial L_p}{\partial \mathbf{w}} = 0 \; ; \; \mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0, i = 1,2,..,n \\
&\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{x}_i = 0, i = 1,2,..,n \\
&\frac{\partial L_p}{\partial b} = 0 \; ; \; \sum_{i=1}^{n} \alpha_i y_i = 0 \\
&y_i(\mathbf{x_i}^T \mathbf{w} + b) - 1 \geq 0, \; i = 1,2,...,n \\
&\alpha_i \geq 0 \\
&\alpha_i \left[y_i \left(\mathbf{x_i}^T \mathbf{w} + b \right) - 1 \right] = 0
\end{aligned} \tag{2.7}$$

Substitusi hasil KKT persamaan (2.7) pada persamaan (2.6) sehingga didapatkan lagrangian untuk dual problem :

$$L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x_i}^T \mathbf{x_j}^T \tag{2.8}$$

Solusi primal dan dual problem adalah :

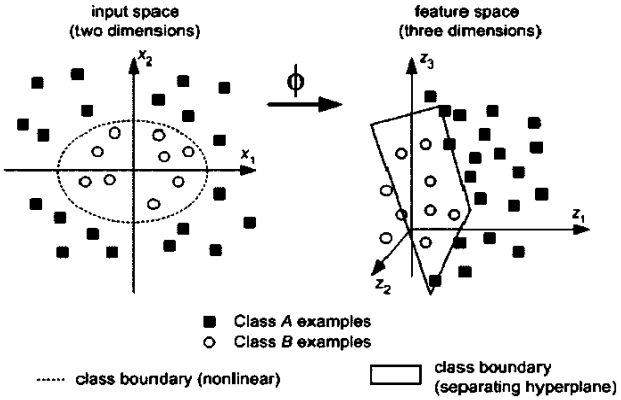
$$\min_{w,b} L_p(w,b)$$

$$\max_{\alpha} L_D(\alpha) \quad \text{s.t.} \quad \alpha_i \geq 0, \sum_{i=1}^{n} \alpha_i y_i = 0$$

Karena optimasi problem adalah konvex maka rumusan dual dan primal memberikan solusi yang sama.

2.1.2 Klasifikasi *Nonlinear* SVM

Pada umumnya, masalah dalam domain dunia nyata (*real world problem*) jarang yang bersifat *linear separable*, kebanyakan bersifat *nonlinear*. Metode untuk mengklasifikasikan data yang tidak dapat dipisahkan dengan fungsi linear adalah dengan mentransformasi data ke dalam dimensi ruang *fitur* (*feature space*) sehingga dapat dipisahkan secara linear pada *feature space. Input space* dengan 2 dimensi tidak dapat memisahkan data kedalam dua kelas secara linier. Oleh karena itu diperlukan pemetaan vektor input oleh fungsi $\Phi(x)$ ke ruang vektor baru yang berdimensi lebih tinggi (3 dimensi). Gambar berikut menunjukkan bahwa dengan 3 dimensi data dapat dipisahkan dalam dua kelas secara linier oleh sebuah *hyperplane*,



Gambar 2.2 *Hyperplane* pada *Non-Linear* SVM (sumber : Lessmann, 2004)

Cara untuk mengklasifikasikan data adalah dengan menggunakan fungsi transformasi $x_i \rightarrow \phi(x_i)$ kedalam *feature space*

sehingga terdapat bidang pemisah yang dapat memisahkan data sesuai dengan kategorinya. Dengan menggunakan fungsi transformasi $x_i \rightarrow \phi(x_i)$, sehingga dihasilkan,

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \phi(x_i)\phi(x_j) + b \qquad (2.9)$$

Feature space dalam prakteknya biasanya memiliki dimensi yang lebih tinggi dari vektor input (*input space*). Hal ini mengakibatkan komputasi pada *feature space* mungkin sangat besar, karena ada kemungkinan *feature space* dapat memiliki jumlah *feature* yang tidak terhingga. Selain itu, sulit untuk mengetahui fungsi transformasi yang tepat. Fungsi transformasi pada SVM adalah menggunakan *kernel trick* (Scholkopf & Simola, 2002). *Kernel trick* adalah menghitung *scalar product* dalam bentuk sebuah fungsi kernel.

$$K(\mathbf{x_i}, \mathbf{x_j}) = \phi\left(x_i\right)^T \phi\left(x_j\right) \qquad (2.10)$$

Maka fungsi transformasi pada persamaan (2.10) dapat digunakan tanpa perlu diketahui fungsi transformasi ϕ secara eksplisit. Dengan demikian fungsi yang dihasilkan adalah,

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i K\left(\mathbf{x_i}, \mathbf{x_j}\right) + b \qquad (2.11)$$

dengan $0 \le \alpha_i \le C \; ; i = 1, 2, ..., n$

Syarat perlu dan cukup pada fungsi simetrik $K(\mathbf{x_i}, \mathbf{x_j})$ untuk menjadi kernel diberikan oleh teorema *Mercer*. Diberikan sebuah kernel K dan data $x_1, x_2, ..., x_n \in X$ maka $K(\mathbf{x_i}, \mathbf{x_j})$ berukuran *nxn*. Sebuah syarat cukup dan perlu untuk matriks simetri K dengan $K_{ij} = K(\mathbf{x_i}, \mathbf{x_j}) = K(\mathbf{x}_j, \mathbf{x}_i) = K_{ji}$. Berikut beberapa fungsi kernel yang umum digunakan,

a. *Kernel Gaussian (RBF)*

$$K\left(\mathbf{x_i}, \mathbf{x_j}\right) = \exp\left(-\gamma\left(\left\|\mathbf{x_i} - \mathbf{x_j}\right\|^2\right)\right) \quad (2.12)$$

b. *Kernel Linear*

$$K\left(\mathbf{x_i}, \mathbf{x}_j\right) = \mathbf{x_i}^T \mathbf{x_j} \quad (2.13)$$

c. *Kernel Polinomial*

$$K\left(\mathbf{x_i}, \mathbf{x_j}\right) = \left(\delta \mathbf{x_i}^T \mathbf{x_j} + r\right)^p \quad (2.14)$$

Bermacam-macam jenis *kernel* lain dapat digunakan untuk pemetaan pada *feature space* namun ketiga *kernel* tersebut yang paling umum digunakan (Santosa, 2007). RBF merupakan fungsi kernel yang banyak digunakan karena RBF dapat mengatasi permasalah nonlinieritas pada data. Hsu, Chang, dan Lin (2003) merekomendasikan fungsi kernel RBF untuk digunakan karena kemampuannya dalam mengatasi nonlinieritas dan RBF memiliki kesulitan numerik yang lebih sedikit dibandingkan fungsi kernel lainnya.

2.2 *Fuzzy Support Vector Machines* (FSVM)

Fuzzy Support Vector Machine (FSVM) merupakan pengembangan *Suport Vector Machine* untuk permasalahan *multiclass*. Dengan menggunakan *decision function* yang diperoleh dari SVM untuk sebuah pasangan kelas, untuk setiap kelas didefinisikan sebuah *polyhedral pyramidal* fungsi keanggotaan. FSVM menggunakan fungsi keanggotaan untuk mengklasfikasikan daerah yang tidak dapat diklasifikasikan oleh *decision function* (Singo & Inoue, 2002). Terdapat beberapa penerapan yang hanya ingin fokus pada akurasi untuk klasifikasi suatu kelas. Untuk tujuan tersebut dapat ditentukan keanggotaan *fuzzy* sebagai fungsi dari masing-masing kelas. Misalkan diberikan rangkaian *training* (Lin & Wang, 2002)

$$(y_1, x_1, s_1), ..., (y_n, x_n, s_n) \quad (2.15)$$

Keanggotaan *fuzzy* s_i menjadi fungsi pada kelas y_i

s_i = 1 jika y_i = 1 dan s_i = 0.1 jika y_i = -1

Untuk menemukan keputusan yang optimal maka perlu untuk menyederhanakan masalah optimasi kuadrat berikut,

$$\min \ \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i=1}^{n} s_i\xi_i \qquad (2.16)$$

$$y_i(\mathbf{w}^T\varphi(\mathbf{x_i}) + b) \geq 1 - \xi_i \ ; \xi_i \geq 0, \ i = 1, 2, .., n$$

dimana ***w*** adalah vektor pembobot pada daerah keputusan, *b* menyatakan bias, $\varphi(\mathbf{x_i})$ merupakan fungsi nonlinear yang memetakan kedalam ruang *feature high dimensional* di mana daerah keputusan yang lebih baik dapat ditemukan, C adalah parameter regularisasi yang dipilih terlebih dahulu untuk mengontrol *trade-off* antara margin klasifikasi dan biaya kesalahan klasifikasi. Variabel ξ_i **(**nonnegatif) menyatakan variabel *slack* dari $\mathbf{x_i}$ pada SVM, sedangkan $s_i\xi_i$ adalah ukuran *error* dengan bobot yang berbeda sesuai dengan s_i .Untuk mengatasi optimasi kuadratik, persamaan 2.16 dinyatakan kedalam *lagragian* sebagai berikut,

$$L_p(\mathbf{w}, b, \xi, \alpha, \mu) = \frac{1}{2}\mathbf{w^T w} + C\sum_{i=1}^{n} s_i\xi_i \qquad (2.17)$$

$$-\sum_{i=1}^{n}\alpha_i(y_i(\mathbf{w}^T\varphi(\mathbf{x_i}) + b) - 1 + \xi_i) - \sum_{i=1}^{n}\mu_i\xi_i$$

Untuk mendapatkan *saddle point* dari $L_p(\mathbf{w}, b, \xi, \alpha, \mu)$, parameter harus memenuhi kondisi KKT.

$$\frac{\partial L_p(\mathbf{w}, b, \xi, \alpha, \mu)}{\partial w} = 0 \rightarrow \mathbf{w} = \sum_{i=1}^{n}\alpha_i y_i\varphi(x_i) \qquad (2.18)$$

$$\frac{\partial L_p(\mathbf{w},b,\xi,\alpha,\mu)}{\partial b} = 0 \rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0 \qquad (2.19)$$

$$\frac{\partial L_p(\mathbf{w},b,\xi,\alpha,\mu)}{\partial \xi_i} = 0 \rightarrow s_i C - \alpha_i - \mu_i = 0 \qquad (2.20)$$

Selanjutnya yaitu menerapkan kondisi KKT tersebut pada *lagrangian* persamaan 2.17 sehingga didapatkan *dual problem* yaitu,

$$\max L_D(\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j K(x_i, x_j) \qquad (2.21)$$

$$\begin{aligned} \sum_{i=1}^{n} \alpha_i y_i &= 0 \\ 0 \le \alpha_i &\le s_i C \\ i &= 1,2,...,n \end{aligned} \qquad (2.22)$$

Fungsi keputusan untuk FSVM dinyatakan sebagai berikut,

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \mathbf{K(x_i, x_j)} + b \qquad (2.23)$$

dimana $0 \le \alpha_i \le s_i C \; ; i = 1,2,...,n$

2.3 *Pre-Processing Data*

Data mentah yang diperoleh perlu dilakukan *pre-processing* terlebih dahulu sebelum dianalisis menggunakan *data mining*. *Pre-processing data* merupakan proses yang dilakukan untuk meningkatkan kualitas data mentah, sehingga dapat meningkatkan akurasi dan efisiensi untuk proses *data mining* selanjutnya. Apabila *input* data berkualitas akan menghasilkan analisis data yang berkualitas pula. Pada penelitian ini *pre-processing data* dilakukan menggunakan trans-

formasi. Transformasi data adalah mengubah data lama menjadi data baru dengan menggunakan prosedur tertentu sehingga analisis data menjadi lebih efisien dan pola yang diperoleh lebih mudah dipahami (Han dkk, 2012). Salah satu metode transformasi adalah *scalling* dimana salah satu keuntungan *scalling* adalah dapat menghindari fitur dengan *range* nilai yang lebih besar mendominasi fitur dengan *range* nilai yang lebih kecil. Setiap variabel secara linier ditransformasi menjadi *range* [0, 1] menggunakan persamaan berikut,

$$v' = \frac{v - \min_a}{\max_a - \min_a} \tag{2.24}$$

dimana

v' = nilai hasil transformasi
v = nilai awal
max_a = nilai maksimum pada variable ke-a
min_a = nilai minimum pada variable ke-a

2.4 *Fast Correlation Based Filter* (FCBF)

Fast Correlation Based Filter atau FCBF adalah salah satu metode seleksi variabel yang dikembangkan oleh Yu dan Liu (2003). Secara umum, sebuah variabel dikatakan bagus jika variabel tersebut relevan dengan konsep kelas namun tidak redunden pada variabel yang sama lainnya. Jika diterapkan korelasi antara dua variabel sebagai ukuran kebaikan, maka definisi dari sebuah variabel dikatakan bagus untuk klasifikasi jika berkorelasi sangat tinggi dengan kelas namun tidak berkorelasi dengan variabel lainnya. Dengan kata lain, jika korelasi antara variabel dan kelas cukup tinggi untuk membuat relevan dengan (atau prediksi) kelas dan korelasi antara variabel dan variabel yang sama lainnya tidak mencapai level sehingga dapat diprediksi oleh salah satu variabel relevan lainnya, itu akan dianggap sebagai variabel yang baik untuk melakukan klasifikasi. Artinya, dalam pemilihan variabel bertujuan untuk menemukan ukuran korelasi yang sesuai antara

variabel dan prosedur yang sama untuk memilih variabel berdasarkan ukuran korelasi.

Terdapat dua pendekatan untuk mengukur korelasi antara dua variabel acak. Pertama, didasarkan pada korelasi linier klasik dan kedua didasarkan pada teori informasi. Ukuran korelasi yang paling dikenal adalah koefisien korelasi linear.

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \qquad (2.25)$$

dimana $\bar{x}$ adalah *mean* dari *X*, dan $\bar{y}$ adalah mean dari *Y*. Nilai *r* terletak antara -1 dan 1, inklusif. Jika *X* dan *Y* benar-benar berkorelasi, *r* mengambil nilai 1 atau -1; Jika *X* dan *Y* benar-benar independen, *r* adalah nol.

Namun pengukuran dengan korelasi tidak mampu menangkap korelasi yang tidak linear, selain itu korelasi tidak mengharuskan semua variabel dan kelas mengandung nilai numerik. Untuk mengatasi kekurangan ini, Yu dan Liu (2003) menerapkan pendekatan lain yaitu memilih ukuran korelasi berdasarkan konsep *information theoretical entropy. Entropy* dari variabel X didefinikan sebegai berikut,

$$H(X) = -\sum_{i}^{n} P(x_i)\log(P(x_i)),\ i = 1, 2, ..., n \qquad (2.26)$$

Entropy dari variabel X jika diketahui variabel Y didefinisikan pada persamaan,

$$H(X \mid Y) = -\sum_{i}^{n} P(y_i)\sum_{i=1}^{n} P(x_i|y_i)\log_2(P(x_i|y_i) \qquad (2.27)$$

$$i = 1, 2, ..., n$$

dimana $P(x_i)$ adalah prior probabilities untuk semua nilai X dan $P(x_i \text{ dan } y_i)$ adalah posterior probabilities dari X jika Y diketahui. Jumlah dimana entropi X menurun mencerminkan

informasi tambahan tentang X yang disediakan oleh Y dan disebut Information Gain (Quinlan, 1993). Dari *entropy* tersebut maka diperoleh *information Gain* adalah

$$IG(X \mid Y) = H(X) - H(X \mid Y) \tag{2.28}$$

dengan H(X) merupakan nilai *etropy* dari variabel X dan H(X|Y) merupakan nilai *entropy* dari variabel X jika Y diketahui. Untuk mengukur *information gain* maka digunakan *symmetrical uncertainty. Information gain* adalah *symmetrical* dari 2 variabel *random* X dan Y. *Symmetrical uncertainty* dirumus-kan sebagai berikut,

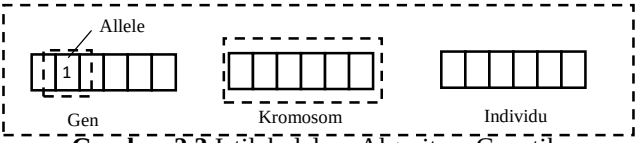
$$SU(X \mid Y) = 2\frac{IG(X \mid Y)}{H(X) + H(Y)} \tag{2.29}$$

Nilai *symmetrical uncertainty* berkisar pada rentang 0 sampai dengan 1 dengan nilai 1 yang menunjukkan bahwa penge-tahuan tentang nilai salah satu sepenuhnya memprediksi nilai yang lain atau X dan Y dependen dan nilai 0 yang menunjukkan bahwa X dan Y mandiri atau independen. Langkah-langkah berbasis entropi memerlukan variabel-variabel nominal, tetapi dapat diterapkan untuk mengukur korelasi antara variabel-variabel kontinu juga, jika nilai-nilai di diskritasikan dengan benar sebelumnya (Fayyad & Irani, 1993; Liu et al., 2002a). Oleh karena itu, peneliti menggunakan ketidakpastian simetris dalam hal ini.

2.5 *Genetic Algorithm*

Genetic algorithm (GA) pertama kali ditemukan oleh John Holand pada tahun 1975. Konsep GA didasarkan pada teori evolusi dengan prinsip seleksi alam yang dikembangkan oleh Darwin. GA merupakan teknik identifikasi pendekatan solusi untuk permasalahan optimasi. Optimasi dengan GA menggunakan kriteria kinerja (*fitness*) untuk mendapatkan solusi optimum. Dalam GA, solusi optimum diperoleh melalui proses seleksi, mutasi dan persilangan yang dilakukan berulang. Konsep *genetic algorithm* yang didasarkan pada ilmu genetik me-

nyebabkan istilah-istilah yang digunakan dalam *genetic algorithm* banyak diadaptasi dari ilmu tersebut. Gambar 2.3 merupakan ilustrasi istilah-istilah yang digunakan dalam algoritma genetika.



Gambar 2.3 Istilah dalam Algoritma Genetika

Keterangan:

Gen = nilai yang menyatakan satuan dasar yang membentuk suatu arti tertentu dalam satu kesatuan gen yang dinamakan kromosom. Dalam algoritma genetika, gen ini bisa berupa nilai biner, integer maupun karakter.

Allele = nilai dari gen

Kromosom = gabungan gen yang membentuk nilai tertentu.

Individu = menyatakan satu nilai atau keadaan yang menyatakan salah satu solusi yang mungkin dari permasalahan yang diangkat

Posisi yang ditempati oleh gen pada kromosom disebut lokus. Nilai yang terkan-dung dalam gen disebut *allele*. Tipe data *allele* bisa berupa biner, *floating point*, atau *integer* tergantung representasi genetic yang digunakan. Sementara gabungan *allele* bisa memberi nilai pada kromosom yang disebut fenotip (Gunawan dkk, 2012),

Langkah-langkah yang dilakukan pada metode GA adalah sebagai berikut :

Langkah 1 : *Define*, yaitu mendefinisikan operator pada GA yang sesuai dengan permasalahan. Pada penelitian ini dilakukan *Variabel selection* dan optimasi dengan menggunakan *genetic algorithm.*

Langkah 2 : *Initialize*, yaitu membentuk populasi awal yang terdiri atas N buah kromosom. N yang digunakan sebesar 100.

Langkah 3 : *Fitness*, yaitu mengevaluasi *fitness* setiap kromosom pada populasi.

Langkah 4 : *Selection*, yaitu menerapkan metode seleksi roulette wheel yang memberikan suatu set populasi perkawinan M dengan ukuran N.

Langkah 5 : *Crossover*, yaitu proses persilangan. Proses ini memasangkan semua kromosom pada M secara acak sehingga membentuk N/2 pasang. Apabila bilangan acak [0,1] kurang dari Pc, maka terjadi pindah silang.

Langkah 6 : *Mutation*, yaitu menggunakan peluang mutasi (Pm) untuk melakukan proses mutasi keturunan.

Langkah 7 : *Replace*, yaitu mengganti populasi yang lama dengan populasi baru. Populasi baru diperoleh dengan memilih N kromosom terbaik yang diperoleh dengan cara mengevaluasi nilai fitness dari orang tua dan keturunan baru.

Langkah 8 : *Test*, yaitu apabila kriteria telah terpenuhi, maka proses berhenti dan kembali ke solusi terbaik dari populasi saat ini. Apabila kriteria belum terpenuhi, maka kembali ke langkah 2. Selanjutnya, elitisme merupakan salah satu teknik yang dilakukan untuk mempertahankan suatu individu terbaik yang memiliki nilai fitness tertinggi untuk dapat bertahan hidup untuk generasi yang selanjutnya.

2.6 Evaluasi Performansi Klasifikasi

Nilai akurasi hasil klasifikasi dapat dihitung menggunakan *confusion matrix. Confusion matrix* adalah alat yang digunakan untuk menganalisis seberapa baik *classifier* mengenali data dari kelas yang berbeda. Pada penelitian ini, nilai ketepatan klasifikasi diketahui dari nilai AUC. Semakin tinggi nilai AUC berarti ketepatan klasifikasi tinggi. Nilai AUC digunakan untuk memperoleh parameter terbaik dimana nilai parameter yang memiliki AUC tertinggi merupakan parameter

optimal serta untuk mengetahui performa klasifikasi. Nilai AUC juga dapat diperoleh dari tabel klasifikasi. Berikut tabel klasifikasi pada penelitian ini,

Tabel 2.1 Tabel Klasifikasi

Aktual	Prediksi	
	Kanker	Normal
Kanker	n_{11}	n_{12}
Normal	n_{21}	n_{22}

Ketepatan klasifikasi dapat diukur melalui *confusion matrix* yang menghasilkan nilai akurasi, sensitivitas, dan spesifitas. Jumlah prediksi kelas ke-j yang tepat diklasifikasian ke kelas ke-i dinotasikan dengan n_{ij}. Akurasi menunjukkan efektifitas *classifier* secara menyeluruh dimana semakin besar nilai akurasi, maka kinerja *classifier* semakin baik. Nilai *fitness* merupakan nilai AUC. Sensifitas digunakan untuk mengukur efektifitas sebuah *classifier* untuk mengidentifikasi kelas positif, sedangkan spesifitas mengukur efektifitas *classifier* dalam mengidentifikasi kelas negatif.

$$accuracy = \frac{n_{11} + n_{22}}{n_{11} + n_{12} + n_{21} + n_{22}} \tag{2.30}$$

$$sensitivity = \frac{n_{11}}{n_{11} + n_{12}} \tag{2.31}$$

$$specifity = \frac{n_{22}}{n_{21} + n_{22}} \tag{2.32}$$

Perhitungan akurasi klasikasi dengan *confusion matrix* dapat digunakan untuk komposisi data yang *balance*. Akurasi klasifikasi pada data *imbalance* dapat dihitung menggunakan *geometric mean* (*G-mean*) dan *Area Under* ROC *Curve* (AUC). *G-mean* untuk kasus mengelompokkan dengan jumlah kategori lebih dari dua dapat menggunakan *Expanding G-mean* dengan rumus seperti persamaan (2.33) dan (2.34) dengan I adalah

jumlah kelas klasifikasi yang terbentuk (Bekkar, Djemaa & Alitouch, 2013).

$$G-mean=\left(\prod_{i=1}^{I}R_i\right)^{\frac{1}{I}}=\sqrt{sensitivity\ \text{x}\ specifity} \quad (2.33)$$

$$AUC=\frac{1}{m}\sum_{i=1}^{I}R_i=\frac{1}{2}(sensitivity+specifity) \quad (2.34)$$

dimana

$$R_i=\frac{n_{ii}}{\sum_{l=1}^{g}n_{il}};i=1,2,....,I \quad (2.35)$$

2.7 *K-folds Cross Validation* (KCV)

Metode *K-folds Cross Validation* (KCV) merupakan suatu metode yang dapat diandalkan (*reliable*) untuk memprediksi kesalahan dalam suatu klasifikasi. Metode ini banyak digunakan oleh peneliti untuk mengurangi bias yang terjadi karena pengambilan sampel data yang akan digunakan. KCV secara berulang-ulang membagi data menjadi data training dan data testing, dimana setiap data berkesempatan menjadi data testing (Gokgoz & Subasi, 2015). Tahapan yang dilakukan dalam menggunakan KCV adalah sebagai berikut :

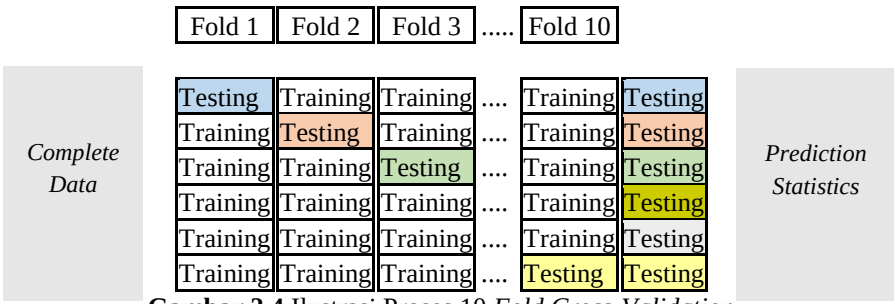
1. Menentukan nilai *K* atau banyak *fold*

   Langkah pertama adalah menentukan nilai *K* atau banyak fold yang akan digunakan. Nilai *K* yang umum digunakan adalah 5 atau 10 dan pada penelitian ini menggunakan 10 fold. 10 *fold cross validation* adalah salah satu K fold CV yang direkomendasikan untuk pemilihan model terbaik karena cenderung memberikan estimasi akurasi yang kurang bias dibandingkan dengan *cross validation* biasa, *leave one out cross validation* dan bootstrap. Dalam 10 fold CV, data dibagi menjadi 10 fold berukuran kira-kira sama,

sehingga kita memiliki 10 subset data untuk mengevaluasi kinerja model atau algoritma (Wibowo, 2017).

2. Melakukan pengambilan data untuk data *training*
   Selanjutnya akan dilakukan pengambilan data secara random sebanyak $K-1$ K n, dimana n adalah banyak data, data yang terambil ini akan disebut data *training*
3. Membuat model dengan menggunakan data *training*
   Kemudian akan dibuat model sesuai dengan metode yang digunakan, dimana data yang digunakan untuk membangun model adalah data training yang telah didapat sebelumnya
4. Menghitung performa model
   Pada tahapan ini akan dihitung performa dari model yang didapat, baik pada data *training* maupun data *testing*. Untuk metode klasifikasi, performa model dapat dilihat menggunakan *accuracy*, *sensitivity*, dan *specificity*
5. Mengulangi langkah 2 hingga langkah 4 sebanyak K.
   Mengulangi langkah 2 hingga langkah 4 sebanyak K-kali, sehingga semua data yang digunakan berkesempatan menjadi data *testing*
6. Menghitung rata-rata performa metode
   Setelah didapat sebanyak K-performa untuk data *training* dan K-performa untuk data *testing*, langkah terakhir adalah menghitung akurasi dari metode yang digunakan secara keseluruhan (CVA).

Berikut ilustrasi 10 *fold validation*,



Gambar 2.4 Ilustrasi Proses 10 *Fold Cross-Validation*

2.8 *Microarray Data*

Microarray data merupakan salah satu teknologi yang digunakan untuk mengukur tingkat ekspresi dari ribuan gen secara bersamaan dalam satu pengamatan dan muncul sebagai perangkat dari *microarray* tersebut yang biasanya dirangkum dalam daftar gen dan dinyatakan dalam dua kondisi atau klasifikasi berdasarkan fenotipnya. *Microarray data* merupakan jenis *high dimensional data* karena memiliki jumlah gen (variabel) ratusan bahkan ribuan, sedangkan jumlah pengamatan yang biasanya tidak mencapai 100 atau jauh lebih kecil dari jumlah Variabel (Yu dan Liu, 2003). Dua metode umum yang dilakukan untuk menganalisis *microarray data* adalah *clustering* dan klasifikasi (Selvaraj dan Natarajan, 2011). Berdasarkan informasi yang dimiliki, *microarray* memiliki peranan penting dalam penelitian biomedis sebagai alat untuk identifikasi dan klasifikasi penyakit, khususnya kanker. Pada penelitian ini, akan dilakukan klasifikasi terhadap *microarray.*

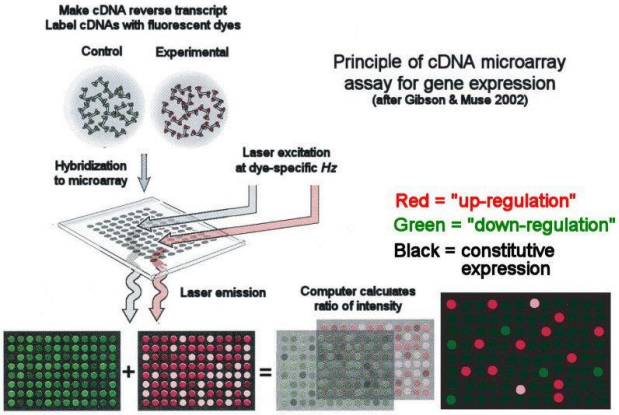
2.9 *Colon Cancer*

Data *colon cancer* pada penelitian ini merupakan data *microarray* yang berasal dari penelitian yang dilakukan oleh Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D, Levine AJ pada tahun 1999. Data *colon* menyimpan informasi berupa ekspresi gen yang diperoleh dari pengamatan pada jaringan usus yang terkena tumor (*tumor colon tissues*) dan jaringan usus normal atau tanpa tumor (*normal colon tissues*). Data *colon* terdiri dari 62 pengamatan, 40 diantaranya merupakan pengamatan kelas tumor dan 22 lainnya merupakan pengamatan kelas normal. Jumlah Variabel yang terdapat pada data *colon* sebanyak 2000 Variabel. Langkah-langkah dari *microarray experiment* agar mendapatkan *microarray colon cancer data* adalah sebagai berikut (Kusumaningrum, 2016),

1. Mendapatkan mRNA dari sel yang diamati (misalkan pada kasus tumor, maka sampel yang diamati adalah sel yang terkena tumor.

2. mRNA dikonversikan menjadi cDNA menggunakan enzim *reverse tranciptase.*
3. Menandai cDMA dari sel tumor dengan warna merah dan cDNA dari sel normal dengan warna hijau.
4. Sampel mengalami hibridisasi, yaitu cDNA saling mengikat terhadap DNA.
5. Sampel dipindai untuk mengukur ekspresi setiap gen melalui *fluorescence* yang terkandung (*fluorescence* berhubungan dengan jumlah cDNA dalam sampel untuk gen tersebut).
6. Titik yang bersinar merah terang adalah *gen* yang sangat diekspresikan dalam sel tumor, sedangkan titik yang bersinar hijau terang adalah *gen* yang sangat diekspresikan dalam sel tumor Apabila gen diekspresikan pada kedua sampel (tumor dan normal), maka warna yang dihasilkan adalah kuning terang.

Dari proses tersebut diperoleh data akhir yang terdiri dari ribuan titik yang memiliki warna berbeda dan perlu di interpretasikan. Titik-titik warna harus dirubah menjadi sebuah nilai tertentu untuk selanjutnya dapat dianalisis. Berikut adalah ilustrasi gambar untuk mendapatkan data *microarray,*



Gambar 2.5 Proses *Microarray data* (sumber : Gibson & Muse, 2002)

(*halaman ini sengaja dikosongkan*)

BAB III
METODOLOGI PENELITIAN

3.1 Deskripsi Data

Data yang digunakan dalam penelitian ini merupakan data jenis *microarray data* kanker kolon yang dilakukan oleh U.Alon, N. Barkai, D.A. Notterman, K. Gish, S. Ybarra, D. Mack, dan A.J. Levine pada tahun 1999. Data diambil dari jaringan usus besar manusia. Ekspresi gen tersebut disimpan dalam 2000 variabel. Data ini terdiri dari 62 pengamatan, yaitu 40 pengamatan *tumor colon tissue* (Tumor/kanker) dan 22 pengamatan *normal colon tissue* (Normal). Berikut adalah variabel penelitian yang digunakan,

Tabel 3.1 Variabel Penelitian

Variabel	Keterangan	Skala
Dependent (Y)	Kelas kanker kolon	Nominal
	0 = kelas normal	
	1 = kelas tumor	
Independent (X)	Ekpresi gen	Rasio

3.2 Struktur Data

Struktur data pada kanker kolon ditunjukkan oleh tabel 3.1 berikut,

Tabel 3.2 Struktur data

Pengamatan ke	Y	X_1	X_2	...	X_{2000}
1	0	8589416	5468241	...	28701
2	0	9164254	671953	...	16774
3	0	3825705	6970361	...	15156
:	:	:	:	...	:
61	1	6234623	40053	...	23265
62	1	737201	3653934	...	39631

3.3 Langkah Analisis

Langkah analisis pada penelitian ini yaitu sebagai berikut :

1. Mendeskripsikan *microarray* kanker kolon
2. Melakukan *pre-processing data* dengan menggunakan transformasi *scalling* seperti pada persamaan 2.24
3. Membagi data kedalam data *training* dan data *testing* menggunakan 10 *fold cross validation* untuk semua variabel sesuai dengan gambar 2.4
4. Melakukan seleksi variabel FCBF pada data kanker kolon
5. Setelah diseleksi, maka data akan dibagi kedalam data *training* dan data *testing* menggunakan 10 *fold cross validation* untuk variabel yang telah diseleksi sehingga data yang digunakan ada 2, yaitu data *cross validation* tanpa seleksi dan data *cross validation* dengan seleksi FCBF.
6. Melakukan analisis klasifikasi menggunakan metode SVM *grid search* untuk mencari nilai C dan γ optimal
7. Melakukan analisis klasifikasi menggunakan metode SVM optimasi GA pada *microarray* kanker kolon dengan data yang tanpa seleksi FCBF dan data dengan seleksi FCBF. Langkah-langkah klasifikasi SVM adalah sebagai berikut,
   a. Menentukan *fitness* atau akurasi sesuai persamaan 2.21 dan nilai AUC sesua persamaan 2.25, nilai P_c, P_m dan *stopping criteria* dimana nilai P_c yang digunakan adalah 0.8 dan nilai P_m yang digunakan adalah 0.01
   b. Menyusun kromosom dengan membangkitkan 100 kromosom dimana kromoson yang dibangkitkan terdiri dari 2 gen yang menunjukkan *hyperparameter* SVM, yaitu C dan γ
   c. Mengevaluasi kromosom berdasarkan nilai *fitness*
   d. Melakukan proses seleksi *roulette wheel* sebanyak 100 kromosom dari 100 induk yang berasal dari populasi
   e. Melakukan proses etilisme
   f. Melakukan pergantian populasi lama dengan generasi baru dengan cara memilih sejumlah kromosom dengan

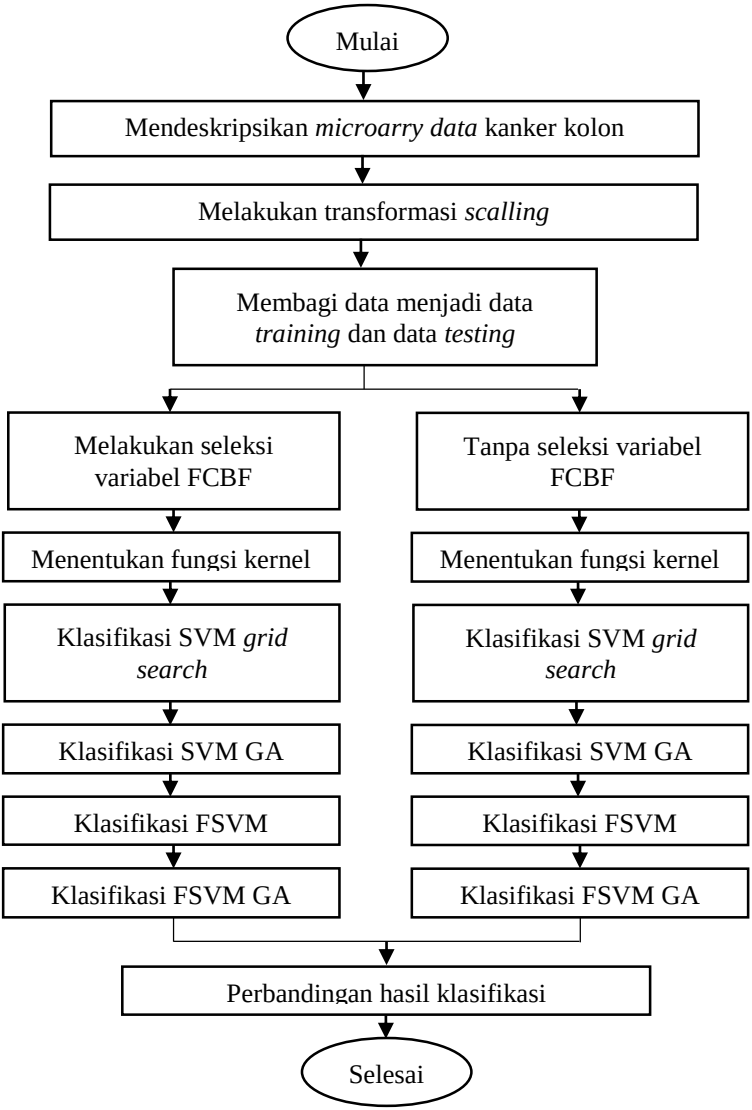
nilai *fitness* terbaik yang telah melalui proses seleksi, pindah silang dan etilisme.

g. Melakukan pengecekan setiap solusi yang telah didapatkan, apabila salah satu *stopping criteria* belum terpenuhi maka kembali melakukan evaluasi kromosom dan apabila semua salah satu *stopping criteria* terpenuhi maka dilanjutkan ke langkah selanjutnya
h. Apabila terdapat kombinasi nilai P_c dan P_m yang belum dilakukan maka kembali ke langkah 5c, yaitu mengevaluasi kromosom. Namun apabila semua kombinasi sudah dilakukan maka dilanjutkan ke langkah selanjutnya
i. Menentukan nilai parameter C dan γ yang paling optimal dengan melihat nilai AUC dan akurasi
j. Menarik kesimpulan berdasarkan hasil analisis yang diperoleh

8. Melakukan analisis klasifikasi menggunakan FSVM dengan data tanpa seleksi dan data dengan seleksi FCBF untuk menentukan C dan γ optimal dengan langkah-langkah sebagai berikut,
   a. Membagi data kedalam data *training* dan data *testing* menggunakan 10 *fold cross validation* sesuai dengan gambar 2.4
   b. Menentukan *fuzzy membership* pada data *training* menggunakan persamaan 2.15
9. Melakukan analisis klasifikasi menggunakan metode FSVM optimasi GA pada *microarray* kanker kolon dengan langkah-langkah sama seperti SVM optimasi GA.
10. Menarik kesimpulan berdasarkan hasil analisis yang diperoleh pada SVM dan FSVM menggunakan optimasi GA untuk data tanpa seleksi dan dengan seleksi FCBF.

3.4 Diagram Alir

Berikut diagram alir menggambarkan langkah analisis yang dilakukan menggunakan *microarray data* kanker kolon,



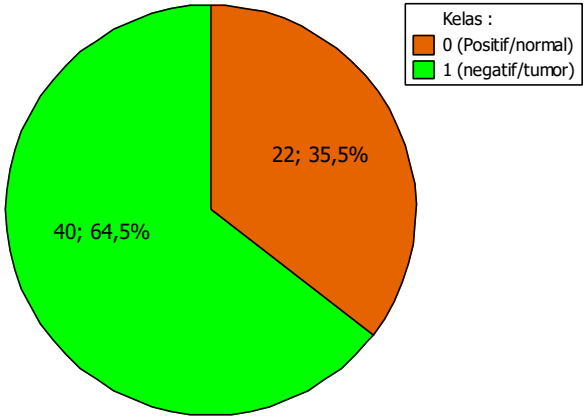
Gambar 3.1 Diagram alir penelitian

BAB IV
ANALISIS DAN PEMBAHASAN

Pada bab analisis dan pembahasan akan dilakukan analisis karakteristik data mengenai kanker kolon dan variabel-variabel yang berupa gen yang selanjutnya akan dilanjutkan analisis menggunakan FSVM (*Fuzzy Support Vector Machine*). Sebelum melakukan analisis FSVM, dilakukan seleksi variabel menggunakan metode FCBF (*Fast Correlation Based Fiter*).

4.1 Karakteristik Data

Data yang digunakan merupakan pengamatan yang dilakukan pada 62 sampel yang diambil dari jaringan usus manusia. Data tersebut dibagi ke dalam dua kelas yaitu kelas tumor dan kelas normal. Deskripsi dari kedua kelas kanker kolon ditunjukkan pada Gambar 4.1,

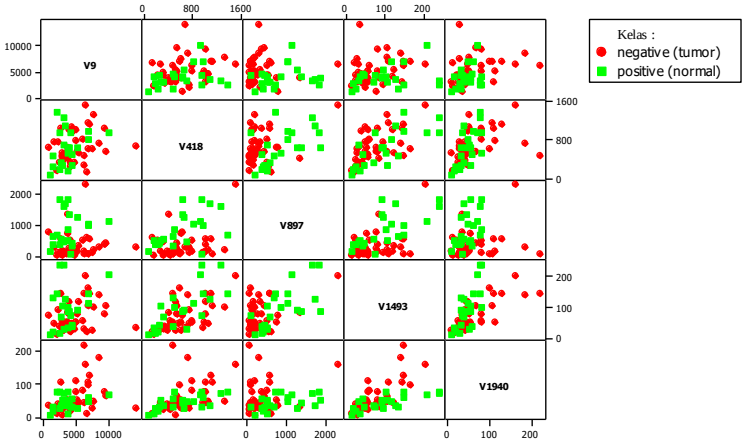


Gambar 4.1 Persentase Kelas Normal dan Tumor pada Data Kanker Kolon

Pada pengamatan ini, sampel positif dinyatakan sampel normal dengan kode 0 dan kelas negatif dinyatakan sampel tumor dengan kode 1. Gambar 4.1 menunjukkan bahwa dari 62 sampel yang diambil terdapat 64,5% dari total sampel dinyatakan tumor dan sisanya yaitu 35,5% dinyatakan normal. Berdasarkan proporsi

sampel kelas normal dan kelas tumor dapat diketahui bahwa data kanker kolon merupakan data *imbalanced* sehingga metode yang sesuai untuk mengklasifikasikan kelas Kanker kolon adalah metode *Fuzzy Support Vector Machine* (FSVM).

Data kanker kolon memiliki jumlah gen atau variabel sebanyak 2000 variabel sehingga jika akan menyulitkan dalam klasifikasi karena pola persebaran data akan menjadi sangat kompleks. Berikut pola persebaran data yang ditunjukkan pada gambar 4.2,



Gambar 4.2 Pola Persebaran beberapa variabel di Data kanker kolon

Matrix plot menunjukkan bahwa pola persebaran data kanker kolon yang ditunjukkan dengan warna merah adalah kelas negatif atau tumor dan warna hijau adalah kelas positif atau normal tersebar merata sehingga menyulitkan dalam klasifikasi. Fungsi pemisah atau *hyperplane* diharapkan mampu membantu mengatasi permasalahan klasifikasi pada data kanker kolon sehingga pada penelitian ini akan dilakukan analisis menggunakan metode klasifikasi *fuzzy support vector machine* (FSVM).

4.2 *Pre-Processing* Data Kanker Kolon

Data yang digunakan dalam penelitian ini merupakan data hasil dengan jumlah variabel sebanyak 2000 sehingga diperlukan *pre-processing* data agar menghasilkan klasifikasi yang tepat. *Pre-processing data* merupakan proses yang dilakukan untuk mening-

katkan kualitas data mentah, sehingga dapat meningkatkan kualitas data dengan menghasilkan nilai akurasi yang tepat dan hasil klasifikasi yang efisien. Pada penelitian ini, *pre-processing* yang dilakukan menggunakan transformasi dan seleksi variabel.

4.2.1 Transformasi Data

Pada penelitian ini *pre-processing data* yang pertama dilakukan dengan menggunakan transformasi. Transformasi data adalah mengubah data lama menjadi data baru dengan menggunakan prosedur tertentu sehingga analisis data menjadi lebih efisien dan pola yang diperoleh lebih mudah dipahami (Han dkk, 2012). Salah satu metode transformasi adalah *scalling* dimana salah satu keuntungan *scalling* adalah dapat menghindari fitur dengan *range* nilai yang lebih besar mendominasi fitur dengan *range* nilai yang lebih kecil. Transformasi *scalling* diperoleh dari pembagian antara (data ke-*i* dikurangi nilai minimal dari variabel ke-*a*) dibagi dengan *range* (nilai maksimal dikurangi nilai minimal) pada variabel ke *a*. Setiap variabel secara linier ditransformasi menjadi *range* [0, 1] menggunakan persamaan 2.24 dan diperoleh hasil sebagai berikut,

Tabel 4.1 Hasil transformasi *scalling* data kanker kolon

Variabel	Nama Variabel	Sebelum Transformasi		Setelah Transformasi	
		Mean	Varians	Mean	Varians
X_1	H55933	7016	9566467	0.3936	0.0569
X_2	R39465	4967	4791241	0.4087	0.0623
X_3	R39465_1	4095	3305418	0.3851	0.0614
X_4	R85482	3988	4076712	0.2784	0.0403
X_5	U14973	2937	1841267	0.2556	0.0384
...	...	...	...	...	...
...	...	...	...	...	...
X_{1999}	R77780	53.25	1479.39	0.2477	0.0404
X_{2000}	T49647	42.97	806.28	0.3070	0.0551

Tiap variabel memiliki nilai pengamatan dengan sebaran yang besar seperti yang dapat diketahui dari nilai varians seperti pada Tabel 4.1. Setelah dilakukan transformasi menggunakan *scalling*, nilai rata-rata setiap variabel menjadi lebih kecil dan

berkisar antara [0,1]. Nilai varians dari variabel yang sudah ditransformasi juga kecil yang menunjukkan bahwa nilai pengamatan dengan sebaran data kecil atau keragaman antara pengamatan satu dengan pengamatan lainnya cukup kecil.

4.2.2 Seleksi Variabel

Variabel yang terdapat pada *microarray data* kanker kolon sebanyak 2000 variabel. Variabel-variabel tersebut diperoleh dari hasil eksperimen pada jaringan manusia berkaitan tentang kanker kolon sehingga dari 2000 variabel yang diperoleh mempunyai kemungkinan ada yang tidak relevan atau signifikan terhadap klasifikasi kanker kolon. Oleh karena itu, diperlukan seleksi variabel untuk mendapatkan variabel-variabel yang sesuai untuk klasifikasi kelas kanker kolon serta untuk mempercepat proses komputasi. Pada penelitian ini, seleksi variabel yang digunakan adalah FCBF (*Fast Correlation Baser Filter*) dan berikut adalah hasil seleksi yang diperoleh,

Tabel 4.2 Hasil Seleksi Variabel Data Kanker kolon

Sebelum seleksi	Sesudah seleksi
2000	15

Variabel yang diperoleh setelah melakukan seleksi variabel dengan FCBF adalah 15 variabel dari 2000 variabel. 15 variabel yang dihasilkan akan dilakukan analisis lebih lanjut menggunakan klasifikasi SVM dan FSVM. Variabel yang terseleksi terdapat pada lampiran 2.

4.3 Klasifikasi Kanker Kolon Menggunakan SVM

Support Vector Machine (SVM) merupakan salah satu metode klasifikasi yang sudah banyak digunakan. Metode SVM pada data kanker kolon menggunakan kombinasi nilai parameter C yaitu 2^{-5}, 2^{-4}, 2^{-3},...,2^{15} dan nilai parameter γ yaitu 2^{-15}, 2^{-14}, 2^{-13},..,2^{3} berdasarkan penelitian yang telah dilakukan oleh Hsu Lin tahun 2003. Banyak *fold* yang digunakan pada penelitian ini yaitu 10 *fold* sehingga untuk mendapatkan nilai parameter *cost* dan γ optimal dilakukan percobaan dengan menggunakan data *training* untuk masing-masing *fold* dengan kombinasi *range* parameter *cost* yaitu

$2^{-5} – 2^{-1}$, $2^{-1} – 2^{3}$, $2^{3} – 2^{7}$, $2^{7} – 2^{11}$, dan $2^{11} – 2^{15}$ serta kombinasi *range* nilai parameter γ adalah $2^{-15} – 2^{-9}$, $2^{-9} – 2^{-3}$, dan $2^{-3} – 2^{3}$. Untuk menentukan *cost* dan γ optimal, pada penelitian ini menggunakan nilai akurasi dan nilai AUC. Sedangkan untuk mengetahui performa klasifikasi menggunakan nilai akurasi, sensitifitas, spesifisitas, *G-means,* dan AUC. Klasifikasi SVM terbagi menjadi 2, yaitu klasifikasi SVM tanpa seleksi dan klasifikasi SVM dengan seleksi FCBF.

4.3.1 Klasifikasi SVM Tanpa Seleksi Variabel

Klasifikasi SVM tanpa seleksi yaitu melakukan klasifikasi kelas kanker kolon dengan menggunakan semua variabel *independent* sebanyak 2000 variabel. Langkah pertama dalam melakukan klasifikasi yaitu mencari nilai *cost* dan γ optimal pada data *training* dan selanjutnya mencari performa klasifikasi dengan *cost* an γ optimal pada data testing. Berikut hasil dari rata-rata akurasi dari setiap kombinasi *range* parameter *cost* dan γ pada data *training* yang digunakan untuk menentukan parameter optimal,

Tabel 4.3 Hasil kombinasi *range* nilai *cost* dan γ data Kanker Kolon Tanpa Seleksi

Range parameter		**Rata-rata akurasi (%)**	**Rata-rata AUC (%)**
C	γ		
$2^{-5} – 2^{-1}$	$2^{-15} – 2^{-9}$	64.40	64.52
	$2^{-9} – 2^{-3}$	64.43	64.52
	$2^{-3} – 2^{3}$	64.47	64.52
$2^{-1} – 2^{3}$	$2^{-15} – 2^{-9}$	83.53	100
	$2^{-9} – 2^{-3}$	85.53	100
	$2^{-3} – 2^{3}$	64.40	64.52
$2^{3} – 2^{7}$	$2^{-15} – 2^{-9}$	87.07	100
	$2^{-9} – 2^{-3}$	86.43	100
	$2^{-3} – 2^{3}$	64.40	100
$2^{7} – 2^{11}$	**$2^{-15} – 2^{-9}$**	**87.43**	**100**
	$2^{-9} – 2^{-3}$	85.30	100
	$2^{-3} – 2^{3}$	64.40	100
$2^{11} – 2^{15}$	$2^{-15} – 2^{-9}$	83.27	100
	$2^{-9} – 2^{-3}$	85.30	100
	$2^{-3} – 2^{3}$	64.40	100

Hasil kombinasi *range cost* dan γ menunjukkan bahwa rata-rata akurasi tertinggi yaitu 87.433% serta nilai AUC sebesar 100% berada pada *range cost* $2^7 – 2^{11}$ dan *range* γ $2^{-15} – 2^{-9}$. Hal ini menunjukkan bahwa *cost* optimal berada pada *range cost* $2^7 – 2^{11}$ dan γ optimal berada pada *range* $2^{-15} – 2^{-9}$ karena memiliki nilai rata-rata akurasi dan AUC tertinggi dibandingkan *range cost* dan *range* γ lainnya. Rata-rata nilai akurasi sebesar 87.433% dan rata-rata nilai AUC sebesar 100% diperoleh dari rata-rata nilai akurasi pada 10 *fold* dengan menggunakan *range cost* dan *range* γ yang sama. Setelah mendapatkan *range cost* dan *range* γ optimal maka selanjutnya mencari nilai parameter *cost* dan γ optimal pada *range cost* dan *range* γ yang optimal tersebut yaitu pada *range cost* $2^7 – 2^{11}$ dan *range* γ $2^{-15} – 2^{-9}$. Untuk mendapatkan nilai *cost* dan γ optimal diperoleh menggunakan nilai akurasi dan AUC pada 10 *fold* dimana masing-masing *fold* akan menghasilkan nilai *cost* dan γ optimal dengan nilai akurasi yang berbeda-beda. Berikut nilai *cost* dan γ optimal pada masing-masing *fold*,

Tabel 4.4 Nilai *cost* dan γ optimal pada masing-masing *fold* Tanpa Seleksi

***Fold* ke-**	**Parameter Optimal**		**Akurasi (%)**	**AUC (%)**
	C	**γ**		
1	2^9	2^{-15}	85.33	100
2	2^{10}	2^{-15}	85.33	100
3	2^{10}	2^{-15}	85.33	100
4	2^7	2^{-9}	86.00	100
5	2^{10}	2^{-15}	89.33	100
6	2^{10}	2^{-15}	87.33	100
7	2^9	2^{-15}	87.67	100
8	**2^7**	**2^{-9}**	**91.00**	**100**
9	2^{10}	2^{-15}	89.99	100
10	2^{10}	2^{-15}	87.67	100

Berdasarkan Tabel 4.4 diketahui bahwa setelah dilakukan percobaan sebanyak 10 kali pada masing-masing data *training* untuk *range cost* $2^7 – 2^{11}$ dan *range* γ $2^{-15} – 2^{-9}$ terdapat nilai *cost*

optimal yaitu 2^7 dan nilai γ optimal yaitu 2^{-9} dengan nilai akurasi sebesar 91% dan nilai AUC sebesar 100% yang terdapat pada *fold* 8. Fungsi *hyperplane* yang terbentuk untuk klasifikasi SVM tanpa seleksi pada data kanker kolon berdasarkan nilai parameter *cost* dan γ optimal yang disajkan pada tabel 4.4 adalah,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana fungsi kernel yang digunakan adalah *Radial Basis Function* (RBF) dengan parameter γ yang diperoleh adalah 2^{-9} atau 0.001953 dengan rumus,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right)$$

$$= \exp\left(-0.001953 \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.001953 \left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2\right) + b$$

dengan $0 < \alpha_i \le C$ atau $0 < \alpha_i \le 2^7$ dan $i = 1, 2, ..., n$

Performa klasifikasi diperoleh dengan menghitung nilai akurasi, sensitifitas, spesifisitas, G-means dan AUC dari model SVM menggunakan parameter *cost* dan γ optimal yaitu 2^7 dan 2^{-9}. Berikut adalah performa klasifikasi yang dihitung dari rata-rata performa klasifikasi 10 *fold* data testing tanpa seleksi,

Tabel 4.5 Performa Klasfikasi SVM Kanker Kolon Tanpa Seleksi

Ukuran Performa Klasifikasi	**Nilai (%)**
Akurasi	83.57
Sensitifitas	83.33
Spesifisitas	89.50
G-*Means*	85.31
AUC	86.42

Performa klasifikasi seperti pada Tabel 4.5 menunjukkan bahwa dengan menggunakan metode klasifikasi SVM tanpa seleksi dapat mengklasifikasikan dengan benar untuk kelas normal ataupun tumor sebesar 83.57%. selain itu, dengan menggunakan parameter *cost* dan γ optimal yaitu 2^{7} dan 2^{-9} diperoleh nilai sensitifitas sebesar 83.33%, spesifisitas sebesar 89.50%, G-*means* sebesar 85.31% dan nilai AUC sebesar 86.42%. Berdasarkan pembagian ketegori AUC oleh Gorunescu 2011, untuk klasifikasi SVM tanpa seleksi tergolong baik karena nilai AUC berada pada *range* 0.80 – 0.90.

4.3.2 Klasifikasi SVM Seleksi Variabel FCBF

Variabel yang akan dianalisis menggunakan SVM dengan seleksi FCBF adalah data yang dihasilkan dari seleksi variabel dengan banyak variabel *independent* sebanyak 15. Data yang digunakan dalam analisis ini menggunakan data pembagian *k-fold* yang sama dengan klasifikasi data SVM tanpa seleksi. *Range cost* dan *range* γ menggunakan kombinasi *range* parameter *cost* yaitu $2^{-5} – 2^{-1}$, $2^{-1} – 2^{3}$, $2^{3} – 2^{7}$, $2^{7} – 2^{11}$, dan $2^{11} – 2^{15}$ serta kombinasi *range* nilai parameter γ adalah $2^{-15} – 2^{-9}$, $2^{-9} – 2^{-3}$, dan $2^{-3} – 2^{3}$. Untuk 1 *range cost* akan di kombinasikan dengan masing-masing *range* γ. 1 *range cost* dan 1 *range* γ akan dianalisis sebanyak sebanyak data *fold* yang digunakan, yaitu 10 *fold* dan akan menghasilkan nilai rata-rata akurasi ataupun nilai rata-rata AUC. Sebelum memperoleh performa klasifikasi SVM dengan seleksi FCBF, maka terlebih dahulu mencari nilai *cost* dan γ optimum menggunakan data *training*. Berikut hasil rata-rata akurasi untuk *range cost* dan *range* γ data *training* dengan seleksi FCBF,

Tabel 4.6 Hasil kombinasi *range* nilai *cost* dan γ data Kanker Kolon Seleksi FCBF

Range parameter		**Rata-rata akurasi (%)**	**Rata-rata AUC (%)**
C	***γ***		
$2^{-5} – 2^{-1}$	$2^{-15} – 2^{-9}$	65.19	66.16
	$2^{-9} – 2^{-3}$	65.10	70.23
	$2^{-3} – 2^{3}$	87.20	67.93
$2^{-1} – 2^{3}$	$2^{-15} – 2^{-9}$	64.53	64.52

Tabel 4.6 Hasil kombinasi *range* nilai *cost* dan γ data Kanker Kolon Seleksi FCBF (lanjutan)

Range parameter		**Rata-rata akurasi (%)**	**Rata-rata AUC (%)**
C	γ		
$2^{-1} – 2^{3}$	$2^{-9} – 2^{-3}$	88.13	98.38
	$2^{-3} – 2^{3}$	87.93	93.57
$2^{3} – 2^{7}$	$2^{-15} – 2^{-9}$	85.33	93.73
	$2^{-9} – 2^{-3}$	88.37	98.02
	$2^{-3} – 2^{3}$	88.37	100.00
$2^{7} – 2^{11}$	$2^{-15} – 2^{-9}$	87.87	98.39
	$2^{-9} – 2^{-3}$	88.63	99.46
	$2^{-3} – 2^{3}$	86.27	100.00
$2^{11} – 2^{15}$	$2^{-15} – 2^{-9}$	90.17	98.38
	$2^{-9} – 2^{-3}$	**88.90**	**100.00**
	$2^{-3} – 2^{3}$	85.63	100.00

Pada penelitian ini, untuk mendapatkan nilai parameter yang optimal menggunakan nilai AUC dan akurasi namun untuk prioritas utama dengan melihat nilai AUC. Hal ini dikarenakan data kanker kolon merupakan data *imbalance* sehingga nilai AUC lebih baik dibandingkan nilai akurasi dalam memilih parameter yang optimal. Tabel 4.6 menunjukkan bahwa terdapat 4 *range cost* dan γ yang memiliki rata-rata nilai AUC sebesar 100%. Hal ini menunjukkan bahwa terdapat 4 kemungkinan *range cost* dan γ optimal. Selain dilihat dari nilai AUC, penentuan *range cost* dan γ optimal juga dapat melalui nilai akurasi dimana pada *range cost* $2^{11} – 2^{15}$ dan *range* γ $2^{-9} – 2^{-3}$ memiliki nilai akurasi tertinggi dari 4 *range* parameter yang memiliki nilai AUC 100%.

Setelah mendapatkan *range cost* dan *range* γ optimal, maka selanjutnya yaitu mencari nilai *cost* dan nilai γ optimal yang terletak pada *range* parameter optimal tersebut. Masing-masing data *training* untuk setiap *fold* akan dianalisis menggunakan SVM dengan seleksi FCBF untuk mendapatkan nilai *cost* dan γ optimal dengan melihat nilai AUC dan juga akurasinya. Berikut nilai *cost* dan γ optimal dari masing-masing *fold* dengan menggunakan data *training*,

Tabel 4.7 Nilai *cost* dan γ optimal pada masing-masing *fold* Seleksi FCBF

Fold **ke-**	**Parameter Optimal**		**Akurasi (%)**	**AUC (%)**
	C	γ		
1	2^{13}	2^{-7}	89.33	100
2	2^{11}	2^{-9}	86.67	100
3	2^{12}	2^{-7}	91.67	100
4	2^{15}	2^{-9}	87.67	100
5	2^{14}	2^{-9}	89.33	100
6	$\mathbf{2^{11}}$	$\mathbf{2^{-5}}$	**92.67**	**100**
7	2^{15}	2^{-9}	91.33	100
8	2^{11}	2^{-9}	85.67	100
9	2^{11}	2^{-9}	85.33	100
10	2^{11}	2^{-4}	89.33	100

Nilai *cost* dan γ optimal adalah 2^{11} dan 2^{-5} karena memiliki nilai AUC dan akurasi lebih tinggi dibandingkan parameter *cost* dan γ lain yang diperoleh pada setiap *fold.* Nilai parameter *cost* dan γ yang diperoleh pada data *training* kemudian dihitung performa klasifikasi SVM seleksi FCBF menggunakan data *testing.* Namun sebelum menghitung performa klasifikasi menggunakan *cost* dan γ optimal, berikut fungsi *hyper-plane* untuk klasifikasi data kanker kolon menggunakan SVM se-leksi FCBF dengan γ yang diperoleh sebesar 2^{-5} atau 0.03125,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dengan,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.03125 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.03125 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dimana $0 < \alpha_i \leq C$ atau $0 < \alpha_i \leq 2^{11}$ dan $i = 1, 2, ..., n$

Dengan menerapkan model SVM optimal pada *cost* dan γ optimal menggunakan data *testing* maka berikut performa klasifikasi yang diperoleh,

Tabel 4.8 Performa Klasfikasi SVM Kanker Kolon Seleksi FCBF

Ukuran Performa Klasifikasi	Nilai (%)
Akurasi	91.90
Sensitifitas	88.33
Spesifisitas	95.50
G-*Means*	91.92
AUC	91.40

Tabel 4.8 menunjukkan bahwa akurasi yang diperoleh sebesar 91.90% yang berarti bahwa dengan menggunakan nilai *cost* 2^{11} dan γ sebesar 2^{-5} model SVM dapat mengklasifikasikan 91.90% pengamatan dengan benar. Selain itu, model juga dapat mengklasifikasikan kelas positif atau normal dengan benar sebesar 88.33% yang dapat diketahui dari nilai sensitifitas dan sebesar 95.50% model dapat mengklasifikasikan kelas negatif atau tumor dengan yang dapat diketahui dari nilai spesifisitas. Nilai AUC yang diperoleh pada klasifikasi SVM dengan seleksi FCBF tergolong sangat baik karena terletak pada *range* 90-100%. Artinya, model SVM seleksi FCBF dengan nilai *cost* 2^{11} dan γ sebesar 2^{-5} sudah sangat baik dalam mengklasifikasikan data kanker kolon.

4.4 Klasifikasi Kanker Kolon SVM dengan Optimasi GA

Setelah mendapatkan hasil klasifikasi dari parameter yang optimal menggunakan SVM *grid search* maka selanjutnya yaitu mengoptimasi parameter yang diperoleh dengan menggunakan *genetic algorithm* (GA). Berdasarkan algorithma genetika, nilai peluang *crossover* (P_c) yang digunakan untuk optimasi parameter adalah 0.8 dan nilai peluang *mutation* (P_m) adalah 0.01. Pada klasifikasi SVM optimasi GA, data *training* digunakan untuk mencari parameter optimal dan data *testing* digunakan untuk mencari performa klasifikasi berdasarkan parameter optimal yang diperoleh. Penentuan parameter optimal dilakukan menggunakan

nilai akurasi dan AUC dimana parameter yang memiliki nilai akurasi dan AUC tertinggi adalah parameter optimal yang terpilih untuk dilakukan pemodelan untuk data *testing*. Optimasi GA dilakukan pada data kanker kolon tanpa seleksi dan kanker kolon dengan seleksi FCBF.

4.4.1 Klasifikasi SVM Tanpa Seleksi Variabel dengan Optimasi GA

Pada analisis SVM menggunakan optimasi parameter GA, parameter yang akan dianalisis yaitu parameter optimal yang diperoleh pada SVM *grid searh* tanpa seleksi. Metode SVM *grid search* tanpa seleksi menghasilkan range *cost* dan *range* γ optimal adalah $2^{7} – 2^{11}$ dan *range* γ $2^{-15} – 2^{-9}$. Setiap *fold* akan dilakukan optimasi parameter GA dengan *range cost* dan *range* γ yang sama sehingga akan menghasilkan nilai *cost* maupun nilai γ yang berbeda untuk setiap *fold* dengan nilai *fitnessvalue* yang berbeda pula. Data yang digunakan untuk memperoleh parameter optimal yaitu data *training* masing-masing *fold.* Parameter optimal dengan GA dipilih dari nilai *fitnessvalue* tertinggi serta nilai *cost* dan γ yang diperoleh akan digunakan untuk menghitung performa klasifikasi untuk data *testing.* Berikut nilai *cost* dan γ optimal menggunakan optimasi GA untuk masing-masing *fold*,

Tabel 4.9 Nilai *cost* dan γ Optimasi GA pada masing-masing *fold* Tanpa Seleksi

***Fold* ke-**	**Parameter Optimal**		***Fitnessvalue* (%)**
	C	**γ**	
1	918.37	0.0009045	89.09091
2	1394.51	0.0012332	89.09091
3	1194.73	0.0010692	89.28571
4	1003.60	0.0009019	89.28571
5	1105.89	0.0005214	89.28571
6	1066.02	0.0008236	92.85714
7	1109.25	0.0010337	89.28571
8	**1272.99**	**0.0011941**	**94.64286**
9	1011.91	0.0011557	92.85714
10	1269.96	0.0007557	91.07143
Rata-rata			90.67532

Rata-rata nilai *fitnessvalue* seperti Tabel 4.9 yang diperoleh pada optimasi GA menggunakan *range cost* dan *range* γ optimal sebesar $2^{7} – 2^{11}$ dan *range* γ $2^{-15} – 2^{-9}$ adalah 90.67%. Nilai *fitnessvalue* yang sama belum tentu menghasilkan nilai *cost* dan γ optimal pada optimasi GA. Pada SVM optimasi GA kanker kolon tanpa seleksi diperoleh parameter *cost* optimal sebesar 1272.99 dan parameter γ optimal sebesar 0.0011941 dengan nilai *fitnessvalue* tertinggi yaitu 94.64%. Berdasarkan nilai *cost* dan γ optimal yang telah diperoleh, maka fungsi *hyperplane* untuk klasifikasi SVM optimasi GA tanpa seleksi adalah,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.0011941 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i \in S} \alpha_i y_i \exp\left(-0.0011941 \left\| \mathbf{x}_i - \mathbf{x} \right\|^2 \right) + b$$

dengan $0 < \alpha_i \leq C$ atau $0 < \alpha_i \leq 1272.99$

Performa klasifikasi pada data *testing* dihitung dari nilai *cost* dan γ optimal yang diperoleh dari data *training.* Berikut performa klasifikasi pada klasifikasi SVM optimasi GA tanpa seleksi,

Tabel 4.10 Performa Klasfikasi SVM Optimasi GA Tanpa Seleksi

Ukuran Performa Klasifikasi	**Nilai (%)**
Akurasi	83.57
Sensitifitas	83.33
Spesifisitas	89.50
G-*Means*	85.31
AUC	86.42

Hasil dari performa klasifikasi SVM optimasi GA tanpa seleksi menunjukkan bahwa akurasi, sensitifitas, spesifisitas, *g-means*, dan AUC secara berturut-turut mempunyai nilai sebesar 83.57%, 83.3%, 89.50%, 85.31%, dan 86.42%. Nilai sensitifitas menunjukkan kefektifan model dalam mengklasifikan kelas positif atau normal dengan baik yaitu 83.57%, sebesar 89.50% model SVM optimasi GA tanpa seleksi dapat mengklafikasikan kelas negatif atau tumor dan keseimbangan antara kinerja kelas minoritas dan kelas mayoritas dalam melakukan klasfikasi model SVM optiasi GA tanpa seleksi sebesar 85.31%. Selain itu, model juga tergolong baik dalam melakukan klasifikasi menggunakan metode SVM optimasi GA tanpa seleksi jika dilihat dari nilai AUC yang berada pada *range* 80 sampai 90%.

4.4.2 Klasifikasi SVM Seleksi Variabel dengan Optimasi GA

Setelah mendapatkan nilai *cost* dan γ optimal yang diperoleh dari klasifikasi *grid search* seleksi FCBF, maka selanjutnya akan dilakukan optimasi parameter tersebut menggunakan optimasi GA. Optimasi GA dilakukan dengan menggunakan *range* parameter terpilih pada metode SVM *grid search* yaitu *range cost* $2^{11} – 2^{15}$ dan *range* γ $2^{-9} – 2^{-3}$. Sama seperti SVM *grid search* seleksi FCBF, setelah mendapatkan *rage* optimal maka langkah selanjutnya yaitu mencari nilai parameter *cost* dan γ optimal pada setiap *fold* dengan data *training.* Nilai *cost* dan γ optimal akan dipilih berdasakan nilai *fitnessvalue* tertinggi diantara semua *fold.* Dari *cost* dan γ optimal akan dicari performa klasifikasi untuk SVM optimasi GA seleksi FCBF dengan menggunakan data *testing.* Berikut adalah hasil masing-masing nilai *cost* dan γ optimal untuk setiap *fold* beserta *fitnessvalue* yang diperoleh,

Tabel 4.11 Nilai *cost* dan γ Optimasi GA dengan Seleksi FCBF

***Fold* ke-**	**Parameter Optimal**		***Fitnessvalue* (%)**
	C	γ	
1	18666.06	0.058681	96.36364
2	13232.03	0.056044	94.54545
3	19730.88	0.076845	94.64286

Tabel 4.11 Nilai *cost* dan γ Optimasi GA dengan Seleksi FCBF (lanjutan)

Fold **ke-**	**Parameter Optimal**		***Fitnessvalue*** **(%)**
	C	γ	
4	**18652.00**	**0.061011**	**96.42857**
5	18311.12	0.065603	96.42857
6	15799.54	0.055124	96.42857
7	16006.2	0.056602	94.64286
8	22437.63	0.056966	92.85714
9	18634.11	0.068058	94.64286
10	19314.93	0.054379	94.64286
Rata-rata			95.16234

Nilai *fitnessvalue* parameter *cost* dan γ seperti Tabel 4.11 menunjukkan bahwa dari 10 *fold* yang dianalisis menggunakan SVM optimasi GA menggunakan data *training* mempunyai rata-rata nilai *fitnessvalue* sebesar 95.16%. Terdapat beberapa *fold* yang memiliki nilai *fitnessvalue* yang sama sehingga parameter *cost* dan γ optimal dapat dipilih salah satu diantara nilai *fitnessvalue* tertinggi. Nilai *fitnessvalue* tertinggi sebesar 96.428% yang berada pada *fold* 4,5, dan 6 dengan nilai parameter yang berbeda. Namun pada penelitian ini akan dipilih salah satu *fold* saja diantara ketiga *fold* yang memiliki nilai *fitnessvalue* tertinggi untuk mencari performa klasifikasi SVM optimasi GA dengan menggunakan data *testing*. *Fold* yang dipilih adalah *fold* 5, artinya parameter *cost* dan γ optimal pada klasifikasi SVM optimasi GA data *training* adalag 18652.00 dan 0.0161011.

Fungsi *hyperplane* yang terbentuk berdasarkan nilai *cost* dan γ optimal yang telah diperoleh pada klasifikasi SVM optimasi GA seleksi FCBF maka adalah sebagai berikut,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana ,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-0.0161011 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.0161011 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dengan $0 < \alpha_i \leq C$ atau $0 < \alpha_i \leq 18652$ dan $i = 1, 2, ..., n$

Performa klasifikasi dari nilai *cost* dan γ optimal akan dihitung menggunakan data *testing*. Berikut performa klasifikasi yang meliputi nilai akurasi, sensitifitas, spesifisitas, *gmeans*, dan AUC pada klasifikasi SVM optimasi GA tanpa seleksi pada data kanker kolon,

Tabel 4.12 Performa Klasfikasi SVM Optimasi GA Seleksi FCBF

Ukuran Performa Klasifikasi	Nilai (%)
Akurasi	95.00
Sensitifitas	91.67
Spesifisitas	97.50
G-*Means*	94.29
AUC	94.58

Tabel 4.12 menunjukkan bahwa sebesar 95% model dapat mengklasfikasikan kelas tumor dan normal dengan benar yang dapat diketahui dari nilai akurasi. Semua nilai performa klasifikasi yang meliputi nilai sensitifitas, spesifisitas, *g-means*, dan AUC mempunyai nilai diatas 90%. Nilai sensitifitas menunjukkan kemampuan model dalam mengklasifikasikan kelas positif atau normal dengan benar dan untuk nilai spesifisitas menunjukkan kemampuan model dalam mengklasifikasikan kelas negatif atau tumor dengan benar.

4.5 Klafikikasi Kanker Kolon dengan *Fuzzy* SVM

Selain metode klasifikasi menggunakan *support vector machines* juga akan dilakukan metode klasifikasi dengan *fuzzy support vector machines* atau FSVM. Kinerja dari FSVM yaitu dengan memberikan bobot pada kelas minoritas dengan bobot 1 dan kelas mayoritas dengan bobot 0.1. Menurut Hsu Lin 2003, nilai

bobot (s_i) bersisar antara 0 sampai 1. Hal ini dimaksudkan agar dapat melakukan klasfikasi pada data *imbalance*. Sama seperti klasfikasi SVM, dilakukan analisis FSVM untuk 10 data *training* untuk mendapatkan nilai *cost* dan γ optimal dimana data *training* yang digunakan untuk klasifikasi FSVM sama seperti data *traning* yang digunakan untuk klasifikasi SVM. Hsu Lin menyebutkan bahwa nilai parameter *C* yang menghasilkan nilai akurasi yang bagus pada klasfikasi yaitu 2^{-5}, 2^{-4}, 2^{-3},..,2^{15} dan nilai parameter γ yaitu 2^{-15}, 2^{-14}, 2^{-13},..,2^{3}. Dalam mencari nilai *cost* dan γ optimal di FSVM tidak dapat dilakukan menggunakan *range* namun nilai rata-rata akurasi dihitung untuk setiap kombinasi *cost* dan γ dari 10 *fold*. Sama dengan klasifikasi SVM, pada analisis FSVM juga akan dilakukan klasifikasi tanpa seleksi dan dengan menggunakan seleksi FCBF.

4.5.1 Klasifikasi FSVM Tanpa Seleksi Variabel

Selain melakukan klasifikasi menggunakan SVM, juga dilakukan klasifikasi dengan FSVM karena data kanker kolon tergolong data *imbalance*. Namun sebelum melakukan klasifikasi FSVM terlebih dahulu menentukan parameter *cost* dan γ optimal yang diperoleh dari data *training* seperti yang telah dilakukan pada analisis SVM. Untuk menentukan parameter optimal dilihat melalui nilai AUC dan akurasi tertinggi. Berikut nilai rata-rata akurasi dan rata-rata AUC untuk masing-masing kombinasi *cost* dan γ ,

Tabel 4.13 Nilai rata-rata akurasi *cost* dan γ FSVM tanpa seleksi

Parameter (2^x)		**Rata-rata Akurasi (%)**	**Rata-rata AUC (%)**
Cost	γ		
-5	-15	64.5195	50.0000
	-14	64.7013	50.2632
	:	:	:
	2	65.4286	6.4401
	3	65.4286	6.4401

Tabel 4.13 Nilai rata-rata akurasi *cost* dan γ FSVM tanpa seleksi (lanjutan)

Parameter (2^x)		**Rata-rata Akurasi (%)**	**Rata-rata AUC (%)**
Cost	γ		
-4	-15	64.6981	50.3743
	-14	64.6981	50.3743
	:	:	:
	2	65.4286	51.4401
	3	65.4286	51.4401
.	:	:	:
.	:	:	:
14	-15	0.0000	0
	-14	69.1071	68.72368
	:	:	:
	2	66.8799	53.6564
	3	66.8799	53.6564
15	-15	40	40
	-14	**98.7435**	**98.2237**
	:	:	:
	2	66.8799	53.65643
	3	66.8799	53.65643

Nilai rata-rata akurasi dan AUC seperti Tabel 4.13 menunjukkan bahwa setiap kombinani nilai *cost* dan γ mempunyai rata-rata nilai akurasi dan AUC yang berbeda-beda. Rata-rata tertinggi untuk nilai AUC dan nilai akurasi pada data *training* adalah pada *cost* 2^{15} atau 32768 dan γ 2^{-14} atau 0.000061035 dengan nilai akurasi yang diperoleh sebesar 98.74% dan nilai AUC sebesar 98.22%. Setelah mendapatkan nilai parameter yang optimal pada data *training* maka selanjutnya mencari performa klasifikasi metode melalui data *testing* dengan menggunakan parameter yang optimal di data *training*. Namun, sebelum mencari

nilai performa klasifikasi, fungsi *hyperplane* yang diperoleh menggunakan klasifikasi FSVM tanpa seleksi adalah sebagai berikut,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.000061035 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.000061035 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dengan $0 < \alpha_i \le s_i C$ atau $0 < \alpha_i \le 32768$ dan $i = 1, 2, ..., n$

Berikut performa klasifikasi yang diperoleh menggunakan nilai parameter yang optimal pada data *training* yaitu *cost* 2^{15} dan γ 2^{-14},

Tabel 4.14 Performa Klasfikasi FSVM Tanpa Seleksi

Ukuran Performa Klasifikasi	Nilai (%)
Akurasi	75.71
Sensitifitas	40
Spesifisitas	95
G-*Means*	47.13
AUC	67.50

Performa klasifikasi FSVM tanpa seleksi menunjukkan bahwa sebesar model FSVM tanpa seleksi mempunyai nilai akurasi sebesar 75.71% yang berarti bahwa model dapat meng-klasfikasikan kelas dengan benar sebesar 75.71%. Kemampuan model dalam mengklasifikasikan kelas positif atau normal sebesar 40%, namun untuk kemampuan model dalam mengklasifikasikan kelas negatif atau tumor sebesar 95%. Untuk data *imbalance* performa klasifikasi dapat dilihat melui nilai AUC dan pada penelitian ini, nilai AUC yang diperoleh adalah 67.50. Berdasarkan

kategori AUC oleh Gorunescu 2011, nilai AUC yang berada pada *range* 60-70 menunjukkan klasifikasi yang buruk. Hal ini juga didukung dengan nilai *gmeans* sebesar 47.13% yang berarti bahwa keseimbangan antara kinerja klasifikasi kelas minoritas dan kelas mayoritas sebesar 47.13%.

4.5.2 Klasifikasi FSVM dengan Seleksi Variabel FCBF

Setelah mendapatkan performa klasifikasi untuk FSVM tanpa seleksi, maka selanjutnya mencari performa klasifikasi untuk FSVM seleksi FCBF. Pertama yang harus dilakukan yaitu mencari nilai *cost* dan γ optimal untuk FSVM seleksi FCBF. Nilai *cost* dan γ optimal dari FSVM tanpa seleksi tidak bisa digunakan dalam FSVM seleksi FBCF. Hal ini dikarenakan data yang akan dianalisis berbeda, pada FSVM tanpa seleksi, banyak variabel *independent* adalah 2000 variabel, sedangkan banyak variabel *independent* pada FSVM seleksi FCBF adalah 15 variabel. Nilai *cost* dan γ optimal diperoleh berdasarkan kombinasi *cost* dan γ yang memiliki nilai rata-rata AUC dan akurasi tertinggi. Nilai *cost* dan γ untuk setiap kombinasi adalah sebagai berikut,

Tabel 4.15 Nilai rata-rata akurasi dan AUC *cost* dan γ FSVM seleksi FCBF

Parameter (2^*)		**Rata-rata Akurasi (%)**	**Rata-rata AUC (%)**
Cost	**Gamma**		
-5	-15	64.5195	50
	-14	64.5195	50
	:	:	:
	2	64.5195	50
	3	64.5195	50
-4	-15	64.5195	50
	-14	64.5195	50
	:	:	:
	2	0	0
	3	19.519481	15

Tabel 4.15 Nilai rata-rata akurasi dan AUC *cost* dan γ FSVM seleksi (lanjutan)

Parameter (2^*)		**Rata-rata Akurasi (%)**	**Rata-rata AUC (%)**
Cost	**Gamma**		
:	:	:	:
	-15	77.2403	67.9211
	-14	77.0617	67.6711
	:	:	:
	-4	29.6396	29.4868
11	**-3**	**98.3896**	**97.8480**
	-2	36.3896	34.8421
	:	:	:
	2	77.5974	68.4211
	3	81.7175	74.2237
	:	:	:
	:	:	:
	-15	0	0
	-14	8.727273	8.2822
14	:	:	:
	2	77.5974	68.4211
	3	81.7175	74.2237
	-15	8.7273	8.2822
	-14	0	0
15	:	:	:
	2	77.5974	68.4211
	3	81.7175	74.2237

Hasil rata-rata nilai akurasi dan AUC FSVM seleksi FCBF menunjukkan bahwa nilai *cost* dan γ optimal adalah 2^{11} atau 2048 dan 2^{-3} atau 0.125 dengan rata-rata nilai Akurasi 98.39% dan rata-rata nilai AUC sebesar 97.89%. Rata-rata nilai akurasi dan nilai

AUC diperoleh dari nilai akurasi dan AUC masing-masing *fold* untuk setiap parameter. Parameter yang optimal akan digunakan untuk mencari performa klasifikasi data *testing*. Berdasarkan nilai parameter *cost* dan γ optimal diperoleh fungsi *hyperplane* yang terbentuk untuk klasifikasi kanker kolon menggunakan metode FSVM seleksi FCBF adalah,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana ,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.125 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.125 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dengan $0 < \alpha_i \le s_i C$ atau $0 < \alpha_i \le 2048$ dan $i = 1, 2, ..., n$

Performa klasifikasi pada data *testing* akan dihitung dengan menggunakan nilai parameter yang optimal pada data *training* yaitu *cost* 2^{15} dan γ 2^{-14} denganhasil sebagai berikut,

Tabel 4.16 Performa Klasfikasi FSVM Seleksi FCBF

Ukuran Performa Klasifikasi	**Nilai (%)**
Akurasi	90.24
Sensitifitas	81.67
Spesifisitas	95
G-*Means*	87.09
AUC	88.33

Performa klasifikasi FSVM seleksi FCBF menunjukkan bahwa dari 10 *fold* yang di analisis menggunakan *cost* 2^{15} dan γ 2^{-14} diperoleh nilai rata-rata akurasi sebesar 90.24% yang berarti bahwa model dapat mengklasifika-sikan dengan benar sebesar 90.24%. Selain itu, model juga dapat mengklasifikasikan kelas

positif atau normal dan kelas negatif atau tumor dengan benar sebesar 81.67% dan 95%. Nilai AUC yang diperoleh adalah 88.33% yang berarti bahwa model sudah baik dalam melakukan klasifikasi.

4.6 Klasifikasi *Fuzzy* SVM pada Kanker Kolon dengan Optimasi GA

Parameter-parameter optimal yang diperoleh dari FSVM akan di optimasi menggunakan optimasi GA. Parameter optimal pada FSVM tanpa seleksi akan dioptimasi dengan FSVM optimasi GA tanpa seleksi juga, begitu juga untuk parameter optimal pada FSVM seleksi FCBF akan dioptimasi dengan FSVM optimasi GA seleksi FCBF.

4.6.1 Klasifikasi FSVM Tanpa Seleksi Variabel dengan Optimasi GA

Nilai *cost* dan γ optimal yang diperoleh dari FSVM tanpa seleksi adalah 2^{15} dan 2^{-14}. Untuk optimasi FSVM GA melalui *range* parameter dimana dalam *range* tersebut terdapat nilai *cost* dan γ optimal. Pemilihan *range* dilakukan dengan *trial and error* yang menghasilkan nilai *fitnessvalue* terbesar. Optimasi akan dilakukan pada masing-masing *fold* dengan *range cost* 2^{15}- 2^{16} dan *range* γ adalah 2^{-14} – 2^{-13}. Rata-rata nilai *fitnessvalue* yang dihasilkan untuk *range cost* 2^{15}- 2^{16} dan *range* γ adalah 2^{-14} – 2^{-13} pada data *training* adalah 98.22%. Sedangkan nilai *fitnessvalue* untuk masing-masing *fold* dengan nilai *cost* dan γ yang dihasilkan adalah sebagai berikut,

Tabel 4.17 Nilai *cost* dan γ Optimasi GA pada masing-masing *fold* FSVM Tanpa Seleksi

Fold	Nilai Parameter		Fitnessvalue (%)
	Cost	Gamma	
1	47494.4	0.000095710	50
2	43011.2	0.000088722	100
3	49171.2	0.000088021	100
4	46546.3	0.000095263	50
5	51019.3	0.000095278	100

Tabel 4.17 Nilai *cost* dan γ Optimasi GA pada masing-masing *fold* FSVM Tanpa Seleksi (lanjutan)

Fold	Nilai Parameter		Fitnessvalue (%)
	Cost	Gamma	
6	52126.7	0.000086322	100
7	50326.1	0.000091523	62.5
8	48698.3	0.000093967	100
9	47159.2	0.000090703	100
10	47249.8	0.000091091	100
Rata-rata			86.25

Nilai rata-rata nilai *fitnessvalue* seperti pada tabel 4.17 adalah 86.25%. Terdapat 7 *fold* yang memiliki nilai *fitnessvalue* mencapai 100% sehingga parameter *cost* dan γ optimal hasil optimasi GA dapat diambil salah satu dari ketujuh nilai *cost* dan γ yang memiliki nilai *fitnessvalue* sebesar 100%. Pada penelitian ini, diambil *cost* dan γ hasil optimasi GA pada *fold* ke 2 dengan nilai *cost* adalah 43011.2 dan nilai γ adalah 0.000088722, sehingga fungsi *hyperplane* yang diperoleh dari klasifikasi FSVM optimasi GA tanpa seleksi adalah,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

dimana ,

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.000088722 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.000088722 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dengan $0 < \alpha_i \le s_i C$ atau $0 < \alpha_i \le 43011.2$ dan $i = 1, 2, ..., n$

4.6.2 Klasifikasi FSVM Seleksi Variabel dengan Optimasi GA

Parameter *cost* dan γ optimal yang diperoleh dari FSVM seleksi FCBF akan dioptimasi menggunakan optimasi GA. *Range* yang digunakan pada FSVM optimasi GA seleksi FCBF adalah 2^{11}- 2^{12} untuk parameter *cost* dan 2^{-3}- 2^{-2} untuk parameter γ . Dengan menggunakan *range* parameter ini, nilai *fitnessvalue* yang dihasilkan pada data *training* mencapai 98.35% sehingga nilai *range* 2^{11}- 2^{12} untuk parameter *cost* dan 2^{-3}- 2^{-2} untuk parameter γ dapat digunakan untuk mencari parameter optimal dengan menggunakan data *testing.* Berikut adalah hasil optimasi GA untuk masing-masing *fold* dengan menggunakan FSVM optimasi GA seleksi FCBF.

Tabel 4.18 Nilai *cost* dan γ Optimasi GA FSVM Seleksi FCBF

Fold	Nilai Parameter		Fitnessvalue (%)
	Cost	Gamma	
1	3146.3	0.19038	100
2	3046.2	0.18109	100
3	3219.6	0.18538	100
4	2968.8	0.15275	100
5	3105.4	0.19883	100
6	3201.3	0.18404	100
7	3003.4	0.19812	100
8	3025.9	0.15704	100
9	3036.3	0.18279	100
10	3299.2	0.16769	75
Rata-rata			97.5

Rata-rata nilai *fitnessvalue* yang dihasilkan dengan menggunakan *range cost* 2^{11}- 2^{12} dan *range* γ 2^{-3}- 2^{-2} adalah 97.5%. Nilai *fitnessvalue* yang diperoleh untuk *fold* 1 sampai *fold* 9 adalah 100% sehingga untuk *cost* dan γ optimal bisa diambil dari salah satu *fold* yang memiliki nilai *fitnessvalue* 100%. Pada penelitian ini, nilai parameter optimal diambil dari *fold* 4 yang memiliki nilai *fitnessvalue* 100% dan nilai *cost* optimal sebesar 2968.8 dan nilai γ optimal adalah 0.15275. Fungsi *hyperplan*

berdasarkan parameter optimal hail optimasi GA pada klasifikasi FSVM optimasi GA seleksi FCBF adalah sebegai berikut,

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i K(\mathbf{x}_i, \mathbf{x}_j) + b$$

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

$$= \exp\left(-0.15275 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right)$$

Sehingga fungsi *hyperplane* yang diperoleh menjadi

$$f(x) = \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i y_i \exp\left(-0.15275 \left\| \mathbf{x}_i - \mathbf{x}_j \right\|^2 \right) + b$$

dengan $0 < \alpha_i \le s_i C$ atau $0 < \alpha_i \le 2968.8$ dan $i = 1, 2, ..., n$.

4.7 Perbandingan Hasil Klasifikasi

Setelah diperoleh performa klasifikasi dari masing-masing metode klasifikasi yang digunakan mak selanjutnya yaitu melakukan perbandingan metode berdasarkan nilai AUC dan akurasi. Perbandingan klasifikasi dilakukan untuk mengetahui metode terbaik untuk mengklasfikasikan data kanker kolon. Berikut perbandingan hasil klasifikasi yang telah dilakukan,

Tabel 4.19 Perbandingan Hasil Klasifikasi

Metode	Tanpa seleksi		Seleksi FCBF	
	Akurasi (%)	AUC (%)	Akurasi (%)	AUC (%)
SVM Grid Search	83.57	86.42	91.90	91.92
SVM optimasi GA	83.57	86.42	95.00	94.58
FSVM	75.71	67.50	90.24	88.33
FSVM optimasi GA	87.38	86.25	90.03	97.50

Metode SVM *grid search* dan SVM optimasi GA tanpa seleksi memiliki nilai akurasi dan AUC yang sama seperti pada Tabel 4.19. Hal ini dapat terjadi karena variabel belum terseleksi. Pada SVM *grid seacrh* seleksi FCBF memiliki nilai performa klasifikasi baik dari nilai akurasi ataupun nilai AUC. Jika

dibandingkan SVM *grid search* dan SVM optimasi GA, dapat diketahui bahwa dengan dilakukan optimasi GA dapat meningkatkan nilai akurasi dan nilai AUC. SVM optimasi GA memiliki nilai AUC sebesar 94.58% dan nilai akurasi sebesar 91.90%. Jika dilihat dari nilai AUC makaklasifikasi SVM optimasi GA sudah sangat baik.

Selain itu, metode klasifikasi performa klasifikasi FSVM optimasi GA lebih tinggi dibandingkan dengan FSVM yaitu nilai AUC FSVM optimasi GA sebesar 86.25%. Sama halnya degan SVM, FSVM ataupun FSVM GA memiliki nilai performa klasifikasi lebih tinggi setelah seleksi FCBF dibandingkan tanpa seleksi FCBF. Secara umum, metode terbaik yang dapat digunakan untuk mengklasifikasikan kelas pada kanker kolon adalah FSVM seleksi variabel FCBF dengan optimasi GA karena memiliki nilai AUC yang paling tinggi dibandingkan metode lainnya yaitu 97.50%.

(*Halaman ini sengaja dikosongkan*)

BAB V
KESIMPULAN DAN SARAN

5.1Kesimpulan

Berdasarkan hasil analisis yang telah dilakukan menggunakan analisis klasifikasi metode *supprot vector machine* dan *fuzzy support vector machine* pada data kanker kolon, maka diperoleh kesimpulan sebagai berikut.

1. Klasifikasi menggunakan metode SVM *grid seacrh* menghasilkan nilai AUC sebesar 86.417% dan nilai akurasi sebesar 83.571% untuk semua variabel. Sedangkan untuk klasifikasi SVM *grid search* seleksi FCBF menghasilkan nilai AUC sebesar 91.917% dan akurasi sebesar 91.905% dengan parameter optimal *cost* 2^{7} dan γ 2^{-9}.
2. Optimasi SVM *grid search* menghasilkan nilai AUC sebesar 94.583% untuk seleksi FCBF dan 86.417% dengan tanpa seleksi FCBF dimana parameter *cost* dan γ optimal untuk SVM *grid seacrh* seleksi FCBF adalah 18652 dan 0.061011.
3. Nilai AUC dan Akurasi FSVM lebih rendah dibandingkan dengan SVM baik untuk yang tanpa seleksi ataupun dengan seleksi FCBF. Nilai AUC dan akurasi FSVM tanpa seleksi adalah 67.50% dan 75.714% sedangkan FSVM seleksi FCBF adalah 88.33% dan 90.238%.
4. Metode klasifikasi FSVM optimasi GA menghasilkan nilai *fitnessvalue* tertinggi dibandingkan metode klasifikasi lainnya dalam penelitian ini. Nilai *fitnessvalue* untuk seleksi FCBF sebesar 97.50% dan yang tanpa seleksi adalah 86.25%.

5.2 Saran

Saran yang dapat diberikan dari penelitian ini yaitu agar menambahkan *range cost* dan *range* γ berdasarkan *literature* sehingga nilai akurasi yang diperoleh lebih baik. Selain itu, dalam analisis juga dapat dilakukan dengan melakukan perbandingan metode seleksi variabel ataupun optimasi parameter lain selain FCBF dan *genetic algorithm* sehingga dapat diketahui klasifikasi terbaik yang dapat digunakan pada data kanker kolon.

(*Halaman ini sengaja dikosongkan*)

DAFTAR PUSTAKA

Abe, Shingo dan; Takuya Inoue. (2002). *Fuzzy Support Vector Machine for Multiclass Problems.* European Symposium on Artificial Neural Networks, Bruges, Belgia

Aisyah, S.N. (2016). *Klasifikasi Berita Online Menggunakan Metode Support Vector Machine dan K-Nearest Neighbor*. Surabaya : Departemen Statistika FMKSD ITS.

Akbani, R., Kwek, S., & Japkowicz, N. (2004). Applying Support Vector Machines to Imbalanced Datasets . European Conference on Machine Learning, Springer, 39-50.

Alon, U., Barkai, N., Notterman, D. A., Gish, K., Ybarra, S., Mack, D., & Levine, A.J. (1999). *Broad Patterns Of Gene Expression Revealed By Clustering Analysis Of Tumor And Normal Colon Tissues Probed By Oligonucleotide Arrays.* Proc Natl Acad Sci U S A. 1999 Jun 8;96(12):6745-6750

Babu, M. M. (2013). Introduction To Micoarray Data Analysis. U.K : Horizon Press.

Bekkar, M., Djemma, H.K., & Alitouch, T.A. (2013). Evaluation Measures for Mmodels Assessment over imbalanced Data Sets. *Journal of Information Engineering and Applications, Vol.3, No.10,* 27-28.

Cariadhi, E, M dkk. (2014). *Implementasi Fuzzy Support Vector Machine Untuk Pengklasifikasian Genre Musik Berdasarkan Variabel Audio.* Malang : Ilmu Komputer Universitas Brawijaya.

Chen, Y.-N., Lu, C.-A., & Huang, C.-Y. (2009). *Anti Spam Filter Based on Naive Bayes, SVM and KNN Model.* Sillicon Valley: Carnegie Mellon School.

Canedo, V. B., Marono, N. S., Betanzos, A. A., Benitez, J., & Herrera, F. (2014). *A Review of Microarray Datasets and Applied Feature Selection Methods. information Science,* 111-135

Furey, T.S., Cristianini, N., Duffy, N., Bednarski, D>W., Schummer, M., & Haussler, D. (2000). *Support Vector Machine Classification and Validation of Cancer Tissue Samples Using Microarray Expression Data.* Bioinformatics, Vol. 16, No. 6, 906-914.

Gibson. (2002). *Organisasi Perilaku Struktur Proses terjemahan edisi V*. Jakarta : Erlangga.

Gokgoz, E., & Subasi, A. (2015). *Comparison of Decision Tree Algorithms for EMG Signal Classification Using DWT*. Biomedical Signal Processing and Control, 18, 138–144. doi:10.1016/j.bspc.2014.12.005

Golub, T. R., Slonim, D. K., Tamayo, P., Huard, C., Gaasenbeek, M., P, M. J., dkk. (1999). Molecular Classification of Cancer: Class Discovery and Class Prediction by Gene Expression Monitoring. Science 286, 531-537.

Gunawan dkk. (2012). Evolutionary Neural Network for Othello Game. *Procedia-Social and Behavioral Sciences*, 419-425.

Gunn, S. (1998). *Support Vector Machine for Classification and Regression.* Southampton: University of Southampton.

Han, J., Kamber, M., & Pei, J. (2012). *Data Mining Concepts and Techniques 3rd Edition.* USA : Morgan Kaufmann

Hardle, W. K., & Simar, L. (2015*). Applied Multivariate Statistical Analysis*. Verlag Berlin Heidelberg: Springer

Hsu, C., Chang, C., & Lin, C. (2003). *A Practical Guide to Support Vector Classification.* Department of Computer Science and Information Engineering, National Taiwan University.

Indriani. (2018). *Fungsi Kolon dalam Pencernaan Tubuh Manusia.. http://www.sridianti.com/fungsi-kolon-dalam-pencernaan-tubuh-manusia.html.* Diakses pada Sabtu, 10 Maret 2018 pukul 19.30 WIB

Info Datin. (2015). *Info Datin, Pusat Data dan Informasi Kementerian Kesehatan Republik Indonesia.* Jakarta : Kementerian Kesehatan RI.

Joachims, T. (1998). *Text Categorization with Support Vector Machines*: Learning with Many Relevant Features

Kusumaningrum, A.P, (2018). *Optimasi Parameter Supprort Vector Machine Menggunakan Genetic Algorithm Untuk Klasifikasi Microarray Data*. Surabaya : Departemen Statistika FMKSD ITS.

Lessmann, S., Stahbolck, R., & Crone, S. F. (2005). *Optimizing Hyperparameters of Support Vector Machines by Genetic Algorithm.* Proceedings of the 2005 International Conference on Artificial Intelligence (ICAI 2005), 74-82

Lin, C., & Wang, S. (2002). Fuzzy Support Vector Machines. IEEE Trans. Neural Network , 464-471.

Mujiyarto, A. (2015). Kompasiana, Daftar Penyakit Mematikan di Dunia. https://www.kompasiana.com/ahmadmujiyarto/daftar-penyakit-mematikan-di-dunia_5500d2d5a33311d37251251b. Diakses pada Minggu, 4 Maret 2018 pukul 21.45 WIB

Nugroho, S.S., Witarto, A.B., & Handoko, D. (2013). *Support Vector Machines : Teori dan Aplikasinya dalam Bioinformatika.* Indonesia Scientific Meeting in Central Japan

Pratiwi, S.N.D. (2016). *Klasifikasi email spam dengan menggunakan Support Vector Machine dan k-nearest neighbor*. Surabaya : Departemen Statistika FMKSD ITS.

Rusydina, A.W. (2016). *Perbandingan Metode Feature Selection Pada High Dimensional Data dan Klasifikasi Menggunakan Support Vector Machines.* Surabaya : Departemen Statistika FMKSD ITS.

Santosa, B. (2007). *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis.* Yogyakarta: Graha Ilmu

Scholkopf, B., & Smola, A. (2002). *Learning with Kernel: Support Vector Machines, Regulerization, Optimization, and Beyond.* Cambridge: MA: MIT Press.

Selvaraj, S., & Natarajan, J. (2011). Microarray Data Analysis and Mining Tools. *Bioinformation*, 6(3), 9-99.

Supianto, A.A & Sutrisno. 2013. *Transformasi Model Warna Yuv Dan Fuzzy Support Vector Machine Untuk Klasifikasi Citra Satelit.* Malang : Informatika Universitas Brawijaya.

Wibowo, A. (2017). *10 Fold Cross Validation.* Jakarta : Universitas Binus

World, G. (2017). Ahlikanker, Urutan Jenis Penyakit Kanker yang Sangat Mematikan. http://www.ahlikanker.com/urutan-jenis-penyakit-kanker-yang-sangat-mematikan/. Diakses pada Minggu, 4 Maret 2018 pukul 22.05 WIB

Vapnik, V. N. (1999). *The Nature of Statictical Learning Theory 2nd Edition.* Springer-Verlag: New York Berlin Heidelberg

Yu, L., & Liu, H. (2003). Feature Selection for High Dimentional Data : A Fast Correlation-Based Filter Solution. *Proceedings of the Twentieth International Conference on Machine Learning (ICML)*. Washington DC

LAMPIRAN

Lampiran 1. Data *Colon cancer* dan Nama Variabel

Pengamatan ke	class	X_1	X_2	....	X_{2000}
1	tumor	8589.416	5468.241	....	28.701
2	normal	9164.254	6719.53	....	16.774
3	tumor	3825.705	6970.361	....	15.156
4	normal	6246.449	7823.534	....	16.085
5	tumor	3230.329	3694.45	....	31.813
6	normal	2510.325	1960.655	....	21.884
7	tumor	7126.599	3779.068	....	24.445
8	normal	4028.71	3156.159	....	52.29
9	tumor	9330.679	7017.23	....	26.848
10	normal	5271.518	4740.768	....	44.043
.	.	.	.	....	.
.	.	.	.	....	.
.	.	.	.	....	.
61	tumor	6234.623	4005.3	....	23.265
62	normal	7472.01	3653.934	....	39.631

Gen ke-	**Gen Assession Number**		**Gene Description**
1	Hsa.3004	H55933	H.sapiens mRNA for homologue to yeast ribosomal protein L41.
2	Hsa.13491	R39465	EUKARYOTIC INITIATION FACTOR 4A (Oryctolagus cuniculus)
3	Hsa.13491	R39465	EUKARYOTIC INITIATION FACTOR 4A (Oryctolagus cuniculus)

Lampiran 1. Data *Colon cancer* dan Nama Variabel (lanjutan)

Gen ke-	Gen Assession Number		Gene Description
4	Hsa.37254	R85482	SERUM RESPONSE FACTOR (Homo sapiens)
5	Hsa.541	U14973	"Human ribosomal protein S29 mRNA, complete cds. "
.	.	.	.
.	.	.	.
.	.	.	.
.	.	.	.
2000	Hsa.9683	T49647	MYRISTOYLATED ALANINE-RICH C-KINASE SUBSTRATE (Homo sapiens)

Lampiran 2. Variabel Terpilih dari Seleksi FCBF

No	Variabel	Gen Assession Number		Gene Description
1	X1671	Hsa.627	M26383	"Human monocyte-derived neutrophil-activating protein (MONAP) mRNA, complete cds.
2	X249	Hsa.8147	M63391	"Human desmin gene, complete cds.
3	X1772	Hsa.6814	H08393	COLLAGEN ALPHA 2(XI) CHAIN (Homo sapiens)
4	X625	Hsa.3306	X12671	Human gene for heterogeneous nuclear ribonucleoprotein (hnRNP) core protein A1.

Lampiran 2. Variabel Terpilih dari Seleksi FCBF (lanjutan)

No	Variabel	Gen Assession Number		Gene Description
5	X1042	Hsa.549	R36977	P03001 TRANSCRIPTION FACTOR IIIA ;.
6	X1227	Hsa.8040	T96873	HYPOTHETICAL PROTEIN IN TRPE 3'REGION (Spirochaeta aurantia)
7	X1153	Hsa.1047	R84411	SMALL NUCLEAR RIBONUCLEOPROTEIN ASSOCIATED PROTEINS B AND B' (HUMAN);.
8	X467	Hsa.2588	H40560	THIOREDOXIN (HUMAN);.
9	X377	Hsa.36689	Z50753	H.sapiens mRNA for GCAP-II/uroguanylin precursor.
10	X1328	Hsa.6317	R39209	HUMAN IMMUNODEFICIENCY VIRUS TYPE I ENHANCER-BINDING PROTEIN 2 (Homo sapiens)
11	X1473	Hsa.1410	R54097	TRANSLATIONAL INITIATION FACTOR 2 BETA SUBUNIT (HUMAN);.
12	X279	Hsa.41126	K03460	"Human alpha-tubulin isotype H2-alpha gene, last exon.
13	X576	Hsa.2487	D14812	"Human mRNA for ORF, complete cds.

Lampiran 2. Variabel Terpilih dari Seleksi FCBF (lanjutan)

No	Variabel	Gen Assession Number		Gene Description
14	X682	Hsa.10047	T51849	TYROSINE-PROTEIN KINASE RECEPTOR ELK PRECURSOR (Rattus norvegicus)
15	X1560	Hsa.21195	R06601	METALLOTHIONEIN-II (Homo sapiens)

Lampiran 3. *Syntax* K*fold*

```
library(MXM)
library(caret)
library(rminer)
library(car)
library(GA)
library(e1071)
set.seed(123)
datacolon=read.csv("E:/FCBF/colonseleksiFCBF.csv",
header=TRUE, sep=";")
CV=generatefolds(datacolon$kelas, nfolds=10,
stratified=TRUE, seed=12)
datatraining1=datacolon[-CV[[1]],]
datatesting1=datacolon[CV[[1]],]
write.csv(datatraining1,"E:/datatraining1.csv")
write.csv(datatesting1,"E:/datatesting1.csv")
```

Lampiran 4. *Syntax* SVM *Grid Search* Data *Training*

```
test=read.csv("E:/FCBF/datatesting10.csv", header=TRUE,
sep=";")
train=read.csv("E:/FCBF/datatraining10.csv",
header=TRUE, sep=";")
library(e1071)
ptm<-proc.time()
range_cost = 2^seq(11,15, by=1)
range_gamma = 2^seq(-3,3, by=1)
hasil = matrix(0,1,4)
auc=function(model)
{
 pred=predict(model,train[,2:16],type="kelas")
 Tabel=table(pred=pred, train[,1])
 r=matrix(0,2,1)
 bobot=matrix(0,2,1)
 for(i in 1:2)
 {
  sum=sum(Tabel[,i])
  r[i]=Tabel[i,i]/sum
  bobot[i]=sum(Tabel[,i])/sum(Tabel[,])
 }
 hasilauc=sum(r*bobot)*100
}
{
 tune_par=tune(svm,
          kelas~.,
          data=train,
          ranges=list(cost=range_cost,
gamma=range_gamma),
          scale=FALSE)
 model=svm(kelas~.,data=train,
cost=tune_par$best.parameters$cost,gamma=
tune_par$best.parameters$gamma)
 hasil=c(tune_par$best.parameters$cost,
```

Lampiran 4. *Syntax* SVM *Grid Search* Data *Training* (lanjutan)

```
tune_par$best.parameters$gamma, auc(model), 1-
tune_par$best.performance)
}
hasil
tune_par$best.parameters$cost
tune_par$best.parameters$gamma
a=auc(model)
a
1- tune_par$best.performance
tune_par$best.parameters
tune_par
tune_par$performance
proc.time()-ptm
```

Lampiran 5. *Syntax* SVM *Grid Search* Data *Testing*

```
library(e1071)
test=read.csv("E:/FCBF/datatesting10.csv", header=TRUE,
sep=";")
train=read.csv("E:/FCBF/datatraining10.csv",
header=TRUE, sep=";")
model<-svm(kelas~., data=train, cost=18652,
gamma=0.06101087, scale=FALSE, kernel='radial')
pred<-predict(model, test[,2:16])
tab=table(pred, test[,1])
sensitivitas=(tab[1,1])/(tab[1,1]+tab[1,2])
spesifisitas=(tab[2,2])/(tab[2,2]+tab[2,1])
akurasi=(tab[1,1]+tab[2,2])/(tab[1,1]+tab[1,2]+tab[2,1]+ta
b[2,2])
AUC<-(sensitivitas+spesifisitas)/2
Gmeans<-(sensitivitas*spesifisitas)^(0.5)
pred
tab
akurasi
sensitivitas
spesifisitas
AUC
Gmeans
```

Lampiran 6. *Syntax* SVM Optimasi GA

```
library(e1071)
library(GA)
datacolon<-read.csv("E:/FCBF/datatraining1.csv",
header=TRUE,sep=";")
ptm<-proc.time()
fitnessFunc<-function(x)
{
  par_cost<-x[1]
  par_gamma<-x[2]
  model<-svm(kelas~.,
             data=datacolon,
             cost=par_cost,
             gamma=par_gamma, cross=10, scale=FALSE)
  return(model$tot.accuracy)
}
theta_min<-c(p_cost=2^11, p_gamma=2^-9)
theta_max<-c(p_cost=2^15, p_gamma=2^-3)
gaControl("real-
valued"=list(selection="ga_rwSelection",crossover="gareal
_laCrossover",mutation="gareal_raMutation"))
fitnesvalue<-c()
solutions<-c()
results<-ga(type="real-valued",fitness=fitnessFunc,
            names=names(theta_min), min=theta_min,
            max=theta_max,          selection=gaControl("real-
valued")$selection,
            crossover=gaControl("real-valued")$crossover,
mutation=gaControl("real-valued")$mutation,
            popSize=100,        maxiter=1000,        run=100,
maxFitness=100, pcrossover=0.8, pmutation=0.01,
            monitor=plot)
summary(results)
solutions=c(fitnesvalue, summary(results)[11])
fitnesvalue=c(fitnesvalue, summary(results)[10])
```

Lampiran 6. *Syntax* SVM Optimasi GA (lanjutan)

```
solutions
fitnesvalue
proc.time()-ptm
```

Lampiran 7. *Syntax* FSVM

```
dataku=vector("list",10)
dataku[[1]]=c(3,9,8,37,35,23,44)
dataku[[2]]=c(14,6,5,60,36,49,39)
dataku[[3]]=c(13,22,29,32,47,56)
dataku[[4]]=c(12,19,24,54,34,27)
dataku[[5]]=c(16,21,31,28,48,57)
dataku[[6]]=c(11,17,52,43,53,38)
dataku[[7]]=c(1,2,41,50,33,25)
dataku[[8]]=c(4,20,55,62,30,42)
dataku[[9]]=c(10,18,51,46,26,45)
dataku[[10]]=c(7,15,61,58,59,40)
datacolonfcbf<-read.csv("E:/FCBF/colonfsvmfcbf.csv",
header=TRUE,sep=";")
y=as.matrix(datacolon[,1])
x=as.matrix(datacolon[,2:16])
set.seed(123)
quad<-function(x,y,cost, Sigma){
  library(kernlab)
  x = as.matrix(x)
  y = as.matrix(y)
  m=dim(x)[1]
  rbf <- rbfdot(sigma = Sigma)
  H <- kernelPol(rbf,x,,y)
  c <- matrix(rep(-1,m))
  A <- t(y)
  b <- 0
  l <- matrix(rep(0,m))
  u <- cost
```

Lampiran 7. *Syntax* FSVM (lanjutan)

```
r <- 0
  capture.output(sv <- ipop(c,H,A,b,l,u,r,verb=TRUE,
sigf=5, margin=1e-8))
  ipopsol<-primal(sv)
  alpha<-matrix(ipopsol, nrow=m)
  #-------------------------------------------------#
  # Calculation of the normal vector W and bias term b
  #-------------------------------------------------#
  w=t(alpha*y)%*%(x) #W
  ff=as.matrix(matrix(rep(alpha*y,m),m,m))%*%H
  fout=matrix(t(apply(ff,2,sum)))
  pos=which(alpha>1e-6)
  b = mean(y[pos]-fout[pos]) #b
  list(W = w, b = b)
}
GG=vector("list", 10)
C=2^-4
G=2^-3
pred<-function(x,lable){
  C=NULL
  n = length(x)
  for (i in 1:n){
    if(x[i]>0){C[i]=1}
    else {C[i]=-1}
  }
  TP = 0
  for (i in 1:n) {if (C[i]==1 & lable[i]==1) (TP=TP+1)}
  FN = 0
  for (i in 1:n) {if (C[i]!=1 & lable[i]==1) (FN=FN+1)}
  FP = 0
  for (i in 1:n) {if (C[i]==1 & lable[i]!=1) (FP=FP+1)}
  TN = 0
for (i in 1:n) {if (C[i]!=1 & lable[i]!=1) (TN=TN+1)}
  Sensi = TP/(TP+FN)
```

Lampiran 7. *Syntax* FSVM (lanjutan)

```
    Spesi = TN/(TN+FP)
  Gmean = sqrt(Sensi*Spesi)
  AUC = (1+(TP/(TP+FN))-(FP/(FP+TN)))/2
  accuracy=mean(C==lable)
  result=list(C,accuracy)
  conmat=table(C,lable) #confusion matrix
  list(accuracy=accuracy, conmat=conmat,
       Sensi = Sensi, Spesi = Spesi,
       Gmean = Gmean, AUC = AUC)
}
for (j in 1:10) {
  xj = x[-dataku[[j]],]
  yj = y[-dataku[[j]],]
  n = length(yj)
  s = NULL
  for (i in 1:n){
    if (yj[i] == 1) {
      s[i] = 1
    } else {
      s[i] = 0.1}
  }
  hasil =(quad(xj,yj,C*s,G))
  Xj = x[-dataku[[j]],]
  Yj = y[-dataku[[j]],]
  fx=t(hasil$W %*% t(as.matrix(Xj))) + hasil$b
  Q=pred(fx,Yj)
  akurasi=Q$accuracy
  sensi=Q$Sensi
  spesi=Q$Spesi
  Gmeans=Q$Gmean
  AUC=Q$AUC
  GG[[j]]=cbind(fx,Yj,akurasi,sensi,spesi,Gmeans,AUC)
}
GG
```

Lampiran 8. *Syntax* FSVM Optimasi GA

```
dataku=vector("list",10)
dataku[[1]]=c(3,9,8,37,35,23,44)
dataku[[2]]=c(14,6,5,60,36,49,39)
dataku[[3]]=c(13,22,29,32,47,56)
dataku[[4]]=c(12,19,24,54,34,27)
dataku[[5]]=c(16,21,31,28,48,57)
dataku[[6]]=c(11,17,52,43,53,38)
dataku[[7]]=c(1,2,41,50,33,25)
dataku[[8]]=c(4,20,55,62,30,42)
dataku[[9]]=c(10,18,51,46,26,45)
dataku[[10]]=c(7,15,61,58,59,40)
datacolon<-read.csv("E:/colonfsvmfcbf.csv",
header=TRUE,sep=";")
y=as.matrix(datacolon[,1])
x=as.matrix(datacolon[,2:16])
set.seed(123)


quad<-function(x,y,cost, Sigma){
  library(kernlab)
  x = as.matrix(x)
  y = as.matrix(y)
  m=dim(x)[1]
  rbf <- rbfdot(sigma = Sigma)
  ## create H matrix etc.
  H <- kernelPol(rbf,x,,y)
  c <- matrix(rep(-1,m))
  A <- t(y)
  b <- 0
  l <- matrix(rep(0,m))
  u <- cost
  r <- 0
  capture.output(sv <- ipop(c,H,A,b,l,u,r,maxiter=500))
  ipopsol<-primal(sv)
```

Lampiran 8. *Syntax* FSVM Optimasi GA (lanjutan)

```
alpha<-matrix(ipopsol, nrow=m)
  #-------------------------------------------------#
  # Calculation of the normal vector W and bias term b
  #-------------------------------------------------#
  w=t(alpha*y)%*%(x) #W
  ff=as.matrix(matrix(rep(alpha*y,m),m,m))%*%H
  fout=matrix(t(apply(ff,2,sum)))
  pos=which(alpha>1e-6)
  b = mean(y[pos]-fout[pos]) #b
  list(W = w, b = b)
}

FSVM=function(x,y,c,G,datak,k){
  pred<-function(x,lable){
    C=NULL
    n = length(x)
    for (i in 1:n){
      if(x[i]>0){C[i]=1}
      else {C[i]=-1}
    }
    TP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]==1) (TP=TP+1)}
    FN = 0
    for (i in 1:n) {if (C[i]!=1 & lable[i]==1) (FN=FN+1)}
    FP = 0
    for (i in 1:n) {if (C[i]==1 & lable[i]!=1) (FP=FP+1)}
    TN = 0
    for (i in 1:n) {if (C[i]!=1 & lable[i]!=1) (TN=TN+1)}
    Sensi = TP/(TP+FN)
    Spesi = TN/(TN+FP)
    Gmean = sqrt(Sensi*Spesi)
    AUC = (1+(TP/(TP+FN))-(FP/(FP+TN)))/2
    accuracy=mean(C==lable)
```

Lampiran 8. *Syntax* FSVM Optimasi GA (lanjutan)

```
    result=list(C,accuracy)
    conmat=table(C,lable) #confusion matrix
    list(accuracy=accuracy, conmat=conmat,
         Sensi = Sensi, Spesi = Spesi,
         Gmean = Gmean, AUC = AUC)
  }

  xj = x[dataku[[k]],]
  yj = y[dataku[[k]],]
  n = length(yj)
  s = NULL
  for (i in 1:n){
    if (yj[i] == 1) {
      s[i] = 1
    } else {
      s[i] = 0.1}
  }
  hasil =quad(xj,yj,c*s,G)
  fx=t(hasil$W %*% t(as.matrix(xj))) + hasil$b
  Q=pred(fx,yj)
  akurasi=Q$accuracy
  sensi=Q$Sensi
  spesi=Q$Spesi
  Gmeans=Q$Gmean
  AUC=Q$AUC
  return(AUC)
}

ptm<-proc.time()
fitnessFunc<-function(x)
{
  par_cost<-x[1]
  par_gamma<-x[2]
```

Lampiran 8. *Syntax* FSVM Optimasi GA (lanjutan)

```
model<-
FSVM(x=x1,y=y1,c=par_cost,G=par_gamma,dataku,9)
  return(model)
}
theta_min<-c(p_cost=2^11, p_gamma=2^-3)
theta_max<-c(p_cost=2^12, p_gamma=2^-2)
gaControl("real-
valued"=list(selection="ga_rwSelection",crossover="gareal
_laCrossover",mutation="gareal_raMutation"))
fitnesvalue<-c()
solutions<-c()
results<-ga(type="real-valued",fitness=fitnessFunc,
        names=names(theta_min),        lower=theta_min,
upper=theta_max,
        selection=gaControl("real-valued")$selection,
        crossover=gaControl("real-valued")$crossover,
mutation=gaControl("real-valued")$mutation,
        popSize=100,        maxiter=100,        run=100,
maxFitness=100, pcrossover=0.8, pmutation=0.01,
        monitor=plot)
summary(results)
solution=c(fitnesvalue, summary(results)[11])
fitnesvalue=c(fitnesvalue, summary(results)[10])
solutions
fitnesvalue
proc.time()-ptm
```

(*halaman ini sengaja dikosongkan*)

BIODATA PENULIS

Penulis yang biasa dikenal dengan panggilan Elok memiliki nama lengkap yaitu Elok Faiqoh. Penulis dilahirkan di Lamongan, 14 Januari 1996 sebagai anak kedua dari tiga bersaudara. Penulis bertempat tinggal di Babat Kabupaten Lamongan dengan ayah bernama Drs. Sucipto, M.Pd dan ibu bernama Antidjah, S.Pd. Penulis menempuh pendidikan formal dimulai dari TK Aisyah Bustanul Athfal V Babat, SDN Banaran 1 Babat, SMPN 1 Babat, dan MAN Babat. Setelah lulus dari SMA/MA, penulis menamatkan studinya sebagai mahasiswi Diploma III Jurusan Statistika FMIPA ITS Surabaya tahun 2016 dengan NRP 1313030067 serta menjadi bagian dari keluarga besar sigma 24 (*legendary*). Tahun pertama masa perkuliahan Diploma, penulis menjadi *volunteer* pengajar ITS Mengajar di pulau Mandangin Madura. Tahun kedua, penulis menjadi staff DAGRI HIMADATA-ITS sekaligus dan lolos PKM didanai DIKTI. Tahun kedua, penulis menjadi ketua biro advokasi departemen kesejahteraan mahasiswa HIMADATA-ITS periode 2015/2016. Saat ini penulis sedang menempuh pendidikan lintas jalur Statistika ITS Surabaya dengan NRP 06211645000005. Selama masa perkuliahan, penulis mengisi waktu luang dengan melakukan *survey* dan juga *input* data di beberapa Perusahaan. Apabila ada kritik dan saran yang ingin didiskusikan dengan penulis melalui email *elokfaiqoh212@gmail.com.*

(*Halaman ini sengaja dikosongkan*)