TUGAS AKHIR – SS141501

KLASIFIKASI *MICROARRAY* KANKER PROSTAT MENGGUNAKAN METODE *HYBRID SUPPORT VECTOR MACHINE – GENETIC ALGORITHM* (SVM-GA) DAN *NAÏVE BAYES*

VIOLITA PERTIWI
NRP 062116 4500 0020

Dosen Pembimbing
Irhamah, M.Si, Ph.D.
Pratnya Paramitha Oktaviana, S.Si., M.Si

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

TUGAS AKHIR – SS141501

KLASIFIKASI *MICROARRAY* KANKER PROSTAT MENGGUNAKAN METODE *HYBRID SUPPORT VECTOR MACHINE – GENETIC ALGORITHM* (SVM-GA) DAN *NAÏVE BAYES*

VIOLITA PERTIWI
NRP 062116 4500 0020

Dosen Pembimbing
Irhamah, M.Si, Ph.D.
Pratnya Paramitha Oktaviana, S.Si., M.Si

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

FINAL PROJECT – SS141501

MICROARRAY CLASSIFICATION OF PROSTATE CANCER USING *HYBRID SUPPORT VECTOR MACHINE – GENETIC ALGORITHM* (SVM-GA) AND *NAÏVE BAYES*

VIOLITA PERTIWI
SN 062116 4500 0020

Supervisors
Irhamah, M.Si, Ph.D.
Pratnya Paramitha Oktaviana, S.Si., M.Si

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

LEMBAR PENGESAHAN

KLASIFIKASI *MICROARRAY* KANKER PROSTAT MENGGUNAKAN METODE *HYBRID SUPPORT VECTOR MACHINE - GENETIC ALGORITHM* (SVM-GA) DAN *NAIVE BAYES*

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 :
Violita Pertiwi
NRP. 062116 4500 0020

Disetujui oleh Pembimbing:

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

Pratnya Paramitha Oktaviana, S.Si., M.Si ( )
NIP. 1300 201405 001

Mengetahui,
Kepala Departemen

Dr. Suhartono
NIP. 19710929 199512 1 001

SURABAYA, JULI 2018

KLASIFIKASI *MICROARRAY* KANKER PROSTAT MENGGUNAKAN METODE *HYBRID SUPPORT VECTOR MACHINE - GENETIC ALGORITHM* (SVM-GA) DAN *NAÏVE BAYES*

Nama Mahasiswa : Violita Pertiwi
NRP : 062116 4500 0020
Departemen : Statistika FMKSD ITS
Dosen Pembimbing : Irhamah, M.Si., Ph.D
Pratnya Paramitha O., S.Si., M.Si

Abstrak

Kanker merupakan masalah kesehatan yang cukup besar di seluruh dunia, salah satu jenis kanker yang banyak menyebabkan kematian adalah kanker prostat yang menduduki peringkat ke lima. Secara khusus, teknologi microarray telah diterapkan untuk prediksi dan diagnosis kanker, sehingga diharapkan dapat mendeteksi tumor atau kanker secara lebih dini dan tepat. Untuk mengklasifikasikan individu yang terjangkit tumor atau kanker secara tepat, pemilihan variabel yang berkaitan dengan kanker harus tepat karena ekspresi gen pada data microarray menghasilkan banyak gen. Pada data microarray jumlah variabel lebih besar dibandingkan jumlah observasi, sehingga perlu dilakukan klasifikasi dengan metode machine learning salah satunya metode Support Vector Machine (SVM) dan naïve bayes. Pada penelitian ini parameter SVM diatur menggunakan metode Grid Search, selain itu juga menggunakan Genetic Algorithm (GA) untuk mengoptimasi parameter SVM. Hasil analisis menunjukkan bahwa metode SVM-GA dengan seleksi variabel menggunakan FCBF menghasilkan performa klasifikasi yang lebih baik dibandingkan metode SVM dan naïve bayes untuk data microarray kanker prostat. Selain itu, optimasi parameter menggunakan metode GA dapat meningkatkan nilai akurasi pada klasifikasi data.

Kata Kunci : Genetic Algorithm, Microarray, Naïve Bayes,* Optimasi Parameter, *Support Vector Machine.

(Halaman ini sengaja dikosongkan)

MICROARRAY CLASSIFICATION OF PROSTATE CANCER USING *HYBRID SUPPORT VECTOR MACHINE – GENETIC ALGORITHM* (SVM-GA) AND *NAÏVE BAYES*

Student Name : Violita Pertiwi
Student Number : 062116 4500 0020
Department : Statistics
Supervisors : Irhamah, M.Si., Ph.D
Pratnya Paramitha O., S.Si., M.Si

Abstract

Cancer is one of health problems in world. Prostate cancer is ranked fifth on causing death of human. In particular, microarray technology has been applied to prediction and diagnosis of cancer, so it is expected to detect tumors or cancers more early and precisely. In order to correctly classify individuals who are infected with a tumor or cancer, selection of variables related to cancer should be appropriate. Usually in microarray data, the number of variables is greater than the number of observations, so it is needed to be classified by machine learning method. One of suitable methods is Support Vector Machine (SVM) and naïve bayes. SVM is a machine learning that has been successfully used to solve classification problems in various fields. The problem in SVM is the difficulty of determining the optimal SVM parameter. In this research, SVM parameter is set using Grid Search method and Genetic Algorithm (GA) is used to optimize the parameter of SVM. GA is a population-based search that can seek a global optimum solution. The results of analysis show that the GA-SVM method gives better classification performance than SVM and naïve bayes for prostate data. In addition, parameter optimization using GA can improve the accuracy value in the data classification.

Keywords : Genetic Algorithm, Microarray, Naïve Bayes, Parameter Optimization, Support Vector Machine.

(This page intentionally left blank)

KATA PENGANTAR

Puji syukur kehadirat Allah SWT atas segala limpahan rahmat serta hidayah-Nya, sehingga penulis dapat menyelesaikan Tugas Akhir yang berjudul **"Klasifikasi *Microarray* Kanker Prostat Menggunakan Metode *Hybrid Support Vector Machine – Genetic Algorithm* (SVM-GA) dan *Naïve Bayes*"**. 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 dan Pratnya Paramitha Oktaviana, S.Si, M.Si selaku dosen pembimbing yang sangat banyak membantu penulis dalam mengatasi masalah-masalah dalam pengerjaan Tugas Akhir, selalu memberikan semangat serta motivasi dan murah hati kepada penulis selama pengerjaan Tugas Akhir ini.
2. Bapak Prof. Nur Iriawan, MIkom., Ph.D. dan Dra. Kartika Fithriasari, M.Si selaku dosen penguji yang telah memberikan banyak saran, kritik dan motivasi demi kesempurnaan Tugas Akhir ini.
3. Ibu Ni Luh Putu Satyaning, M.Sc yang selama ini selalu membantu penulis dalam memberikan wawasan mengenai metode yang digunakan dalam Tugas Akhir ini.
4. Bapak Dr. Suhartono, M.Sc. selaku Ketua Jurasan Statistika ITS yang telah memberikan banyak fasilitas yang menunjang kelancaran penyelesaian Tugas Akhir.
5. Bapak Dr. Sutikno, S.Si, M.Si selaku Ketua Program Studi Sarjana departemen Statistika ITS atas bantuan dan semua informasi yang diberikan.
6. Bapak Dr. Bambang Widjanarko O., S.Si., M.Si. selaku dosen wali atas dukungan dan motivasi yang diberikan sewaktu perwalian.
7. Seluruh dosen dan karyawan di departemen Statistika ITS yang telah memberikan banyak ilmu, pengalaman dan

bantuan kepada penulis selama menempuh proses perkuliahan.

8. Mama, papa dan kak virga tercinta, beserta keluarga besar yang tak henti-hentinya memberikan dukungan, semangat, motivasi dan doa kepada penulis agar dapat menyelesaikan Tugas Akhir ini.
9. Elok faiqoh, cicilia ajeng pratiwi, beti kartikasari, fausania hibatullah dan cici yang telah menjadi sahabat terbaik dari pertama ketemu sampai saat ini, yang terus memberikan semangat dan motivasi kepada penulis.
10. Teman-teman dekat lainnya yang telah banyak memberikan dukungan, semangat, serta keceriaan sehingga penulis dapat menikmati segala proses di Departemen Statistika ITS dengan lancar dan sukses.
11. Teman-teman Lintas Jalur angkatan 2016 yang telah memberikan pengalaman dan kenangan selama menempuh proses perkuliahan.

Dengan berakhirnya pengerjaan laporan Tugas Akhir ini, penulis berharap laporan ini dapat memberikan manfaat kepada 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 TABEL

Halaman

DAFTAR GAMBAR

Halaman

(Halaman ini sengaja dikosongkan)

DAFTAR LAMPIRAN

Halaman

(Halaman ini sengaja dikosongkan)

BAB I
PENDAHULUAN

1.1 Latar Belakang

Cancer Helps (2014), mendefinisikan penyakit kanker secara umum sebagai suatu penyakit yang disebabkan oleh pertumbuhan sel-sel jaringan tubuh yang tidak normal. Sel-sel kanker akan berkembang dengan cepat, tidak terkendali dan terus membelah diri. Selanjutnya menyusup ke jaringan sekitarnya dan terus menyebar melalu jaringan ikat, darah, serta menyerang organ-organ penting dan saraf tulang belakang. Menurut data *International Agency for Research on Cancer* [IARC] (2012), terdapat 14,1 juta kasus baru kanker dengan sekitar 8,2 juta penderita meninggal. Sedangkan pada tahun 2030 diprediksikan angka kejadian kanker meningkat menjati 21,7 juta penderita. Menurut Kemenkes Republik Indonesia (2015), kanker serviks merupakan kanker dengan prevalensi tertinggi di Indonesia, sedangkan kanker terbesar selanjutnya adalah kanker payudara dan kanker prostat.

Kanker prostat adalah salah satu penyakit ganas yang sering terjadi di negara barat dengan insiden sekitar 57 sampai 185 per 100.000 kasus dan kenaikan insiden antara 25% sampai 114% dalam 10 tahun. Menurut Akaza dkk (2002), di negara asia meskipun penyakit ini belum sebanyak apabila dibandingkan di negara barat, namun dalam 10 tahun terakhir pada tahun 1985 – 1995 menunjukkan peningkatan kejadian yang besar di beberapa negara Asia seperti di Jepang dan Korea Selatan. Menurut Umbas (2008), di Indonesia berdasarkan laporan hasil pemeriksaan histopatologi dari 13 pusat pendidikan kedokteran di seluruh Indonesia, penyakit ini termasuk dalam sepuluh penyakit keganasan tersering pada laki-laki. Di rumah sakit "Dr. Cipto Mangunkusumo (RSCM) dan rumah sakit kanker "Dharmais" (RSKD) jumlah penderita kanker prostat menunjukkan peningkatan sekitar 2,5 kali pada periode 1995-2004. Terdapat beberapa cara yang berperan untuk menurunkan resiko terjadinya

kanker prostat, salah satunya yaitu modalitas diagnostik yang lebih baik dan peningkatan usia harapan hidup yang diimbangi oleh pelayanan masyarakat lanjut usia.

Menurut Pusat Data dan Informasi Kemenkes (2015), kejadian kanker prostat akan meningkat pada kelompok usia lebih dari 65 tahun dan sangat jarang terjadi pada usia dibawah 50 tahun. Salah satu faktor yang menyebabkan kejadian baru dari kanker prostat cukup tinggi dapat dikarenakan jumlah lansia di Indonesia yang juga semakin meningkat. Meskipun begitu angka kematian dari kanker prostat tidak terlalu tinggi karena penyebaran dari kanker prostat menyebar secara perlahan. Di Indonesia terdapat peningkatan jumlah spesialis urologi, dengan demikian dapat diperkirakan bahwa peningkatan insiden kanker prostat yang cukup besar akan terjadi pada beberapa tahun ke depan. Sehingga perlu adanya metode klasifikasi yang tepat untuk menekan insiden kanker prostat dengan dilakukan deteksi dini. Deteksi dini kanker bisa menggunakan *microarray* sebagai medianya. Informasi yang terkandung di dalam rangkaian DNA makhluk hidup dapat diketahui melalui teknologi *microarray*. *Microarray* adalah teknologi yang mampu menyimpan ribuan ekspresi gen yang diambil dari beberapa jaringan manusia sekaligus, di dalamnya memiliki potensi yang sangat besar untuk pengetahuan baru yang mendasari kemajuan dalam fungsional genomik dan biologi molekuler.

Pada pengolahan data *microarray* didapatkan informasi penting dari data yang berukuran besar, yang artinya memiliki banyak sekali variabel dengan jumlah bisa ratusan bahkan ribuan. Analisis secara konvensional sangat sulit dilakukan pada data *microarray*, sehingga untuk mengatasi masalah tersebut menggunakan metode *macine learning*. *Macine learning* adalah salah satu disiplin dari *artificial intelligence* yang berfokus untuk membuat komputer mampu belajar secara otomatis tanpa harus diprogram secara eksplisit dan berulang kali. Pada penelitian ini menggunakan seleksi variabel *fast correlation based function* (FCBF) untuk mengurangi variabel data asli menjadi lebih sedikit

namun tetap dapat digunakan untuk klasifikasi. Untuk klasifikasi digunakan metode *support vector macine* (SVM) dan *naïve bayes* sebagai pembandingnya.

Didukung dari penelitian sebelumnya yang berhubungan dengan *gene selection* telah dilakukan oleh Wirasna dkk (2017), mendapatkan kesimpulan metode *backpropagation* termodifikasi dan *principle component analysis* (PCA) memberikan hasil waktu pelatihan yang bagus dalam klasifikasi deteksi kanker, melihat perbandingan dari semua sistem. Penelitian oleh Mubarok dkk (2017), mengenai implementasi *mutual information* dan *bayesian network* untuk klasifikasi data *microarray* didapatkan kesimpulan bahwa metode *Mutual Information* dan *Bayesian Network* dapat mengklasifikasi data *Microarray* dengan hasil rata-rata ketepatan klasifikasi 91.06%, hasil ini didapatkan dari jumlah variabel dan nilai k yang berbeda untuk setiap data. Selain penelitian mengenai *gene selection*, juga terdapat penelitian sebelumnya mengenai metode SVM dan naïve bayes. Firdausanti (2017) dengan klasifikasi kelas risiko pasien pneumonia menggunakan metode *Hybrid* analisis diskriminan linier - *Particle Swarm Optimization* (ADL-PSO) dan *Naïve Bayes Classification*, mempunyai kesimpulan bahwa menggunakan metode analisis diskriminan dengan seleksi variabel PSO mempunyai nilai akurasi yang lebih baik dibandingkan analisis diskriminan dengan seleksi variabel *(backward, forward, stepwise*) dan naïve bayes. Nuansa (2017) juga melakukan penelitian mengenai analisis sentimen pengguna *twitter* terhadap pemilihan gubenur DKI Jakarta dengan metode *Naïve Bayesian Classification* (NBC) *dan Support vector Macine* (SVM) dengan kesimpulan perbandingan antara kedua metode NBC dan SVM didapatkan hasil SVM kernel *Radian Basis Function* lebih baik dibandingkan dengan NBC.

Berdasarkan uraian yang telah dijelaskan sebelumnya, pada tugas akhir ini akan dilakukan penelitian mengenai klasifikasi kanker prostat menggunakan metode *hybrid Support vector Macine – Genetic Algorithm* (SVM-GA) dan *naïve bayes*. Pemilihan metode naïve bayes adalah dikarenakan memiliki

kelebihan salah satu algoritma klasifikasi yang sederhana namun memiliki nilai akurasi yang tinggi. Sedangkan pemilihan metode SVM karena kemampuan generalisasi dalam mengklasifikasikan suatu pola. SVM juga bekerja sangat baik pada data dengan berbagai banyak dimensi dan menghindari kesulitan dari permasalahan dimensionalitas.

1.2 Perumusan Masalah

Seperti yang telah dijelaskan sebelumnya pada latar belakang, bahwa kanker merupakan salah satu penyebab utama kematian seseorang di seluruh dunia. Oleh karena itu, perlu dilakukan pendeteksian secara dini salah satunya dapat melalui ekspresi gen. Ekspresi gen merupakan metode ekstrasi gen menjadi data yang bernama data *microarray*. Dalam kasus *microarray* terdapat masalah dengan banyaknya variabel yang dihasilkan. Sehingga pada penelitian ini digunakan metode *Hybrid Support vector Macine – Genetic Algorithm* (SVM-GA) dan *Naïve Bayes* untuk membandingkan hasil pengklasifikasian kelas seseorang terkena kanker prostat atau tidak melalui nilai akurasi.

1.3 Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penelitian ini adalah sebagai berikut:

1. Mengetahui klasifikasi seseorang terkena kanker prostat dari nilai akurasi menggunakan semua variabel dengan metode *Support vector Macine* (SVM), *Support vector Macine – Genetic Algorithm* (SVM-GA) dan *naïve bayes*.
2. Mengetahui klasifikasi seseorang terkena kanker prostat dari nilai akurasi menggunakan variabel yang telah terseleksi dari metode FCBF dengan metode *Support vector Macine* (SVM), *Support vector Macine – Genetic Algorithm* (SVM-GA) dan *naïve bayes*.
3. Membandingkan ketepatan klasifikasi seseorang terkena kanker prostat menggunakan metode *Support vector Macine*

(SVM), *Hybrid Support Vector Macine – Genetic Algorithm* (SVM-GA) dan *naïve bayes*.

1.4 Batasan Masalah

Batasan masalah yang digunakan pada penelitian ini yaitu data *microarray* menggunakan kanker prostat. Selain itu juga untuk fungsi kernel metode SVM menggunakan kernel RBF.

1.5 Manfaat Penelitian

Hasil penelitian ini diharapkan dapat memberikan manfaat dalam pengembangan ilmu pengetahuan khusunya *Genetic Algorithm* dalam bidang kesehatan. Selain itu juga diharapkan dapat memberikan informasi apabila di Indonesia telah ada data mengenai gen, maka dapat membantu mempercepat proses klasifikasinya.

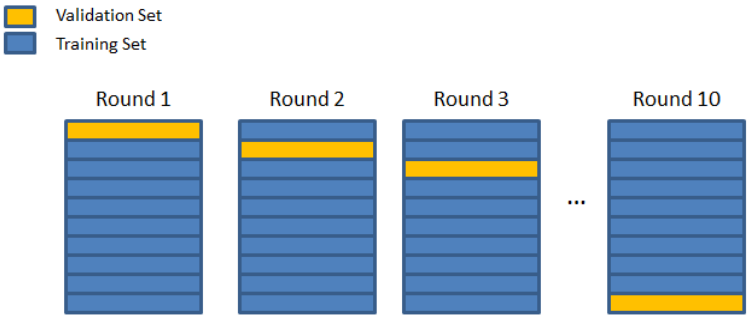
(Halaman ini sengaja dikosongkan)

BAB II
TINJAUAN PUSTAKA

Tinjauan statistik yang digunakan dalam penelitian ini adalah *k-fold cross validation*, *fast correlation based filter*, *genetic algorithm*, *support vector macine*, *naïve bayes* dan pemilihan model terbaik.

2.1 *K-Fold Cross Validation*

Menurut Fu (1994) *K-Fold Cross Validation* adalah sebuah metode yang membagi himpunan contoh secara acak menjadi k himpunan bagian (*subset*). Pada proses ini dilakukan pengulangan sebanyak k kali untuk data pelatihan dan pengujian. Pada setiap pengulangan, satu *subset* digunakan untuk pengujian sedangkan *subset* lainnya digunakan untuk pelatihan. Data awal dibagi menjadi k *subset* secara acak dengan ukuran *subset* yang hampir sama dengan mempertahankan perbandingan antar kelas. Pada iterasi pertama, *subset* satu menjadi data pengujian sedangkan *subset* lainnya menjadi data pelatihan. Pada iterasi kedua, *subset* kedua digunakan sebagai data pengujian dan *subset* lainnya sebagai data pelatihan, dan seterusnya hingga seluruh *subset* digunakan sebagai data pengujian. Menurut Han dkk tahun 2012, pada umumnya banyaknya fold yang digunakan untuk mengestimasi klasifikasi adallaha 10 fold. Ilustrasi dengan menggunakan 10 fold telah digambarkan pada Gambar 2.1.



Gambar 2.1 Ilustrasi 10 *Fold Cross Validation*

2.2 *Fast Correlation Based Filter*

Seleksi variabel bertujuan untuk mengurangi dimensi data, menghilangkan variabel yang tidak sesuaidan memberikan performa yang baik untuk klasifikasi. Salah satu metode seleksi variabel adalah dengan menggunakan algoritma *Fast Correlation Based Filter* (FCBF) yang dikembangkan oleh Yu dan Liu (2003). Algoritma ini didasarkan pada pemikiran bahwa suatu variabel yang baik adalah variabel yang relevan terhadap kelas tapi tidak *redundant* terhadap variabel relevan yang lain. Oleh karena itu, Lei Yu dan Huan Liu melakukan dua pendekatan dengan mengukur korelasi antara dua variabel acak yaitu berdasar pada *linear correlation coefficient* dan berdasar pada teori informasi. Pendekatan *linear correlation coefficient* untuk setiap variabel (X, Y) dirumuskan pada persamaan 2.1.

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

Jika X dan Y memiliki korelasi maka nilai r adalah 1 atau -1, jika tidak berkorelasi maka nilai r adalah 0. Terdapat beberapa keuntungan menggunakan pendekatan ini yaitu mudah untuk menghilangkan variabel yang tidak relevan dengan memilih variabel yang mempunyai nilai korelasi 0 dan membantu mengurangi redundant pada variabel yang sudah dipilih. Namun pendekatan ini juga memiliki keterbatasan yaitu membutuhkan variabel yang memiliki nilai-nilai numerik.

Untuk mengatasi keterbatasan dari pendekatan dengan mengukur korelasi, maka dapat dilakukan pendekatan yang kedua yaitu pendekatan berdasar pada *information-theorical concept of entropy* (mengukur ketidakpastian pada random variabel). Berikut ini adalah langkah-langkah untuk seleksi variabel menggunakan metode FCBF.

1. Menghitung entropi dari setiap variabel prediktor yang didefinisikan pada persamaan 2.2.

$$H(x) = -\sum_{i=1}^{n} P(x_i) \log_2 (P(x_i)) \quad (2.2)$$

2. Menghitung nilai entropi setiap variabel prediktor jika diketahui variabel respon yang didefinisikan pada persamaan 2.3.

$$H(X \mid Y) = -\sum_{j=1}^{n} P(y_i) \sum_{i=1}^{n} P(x_i \mid y_j) \mid \log_2 (P(x_i \mid y_j)) \quad (2.3)$$

dengan,

$P(y_i)$: *Prior probability* untuk semua nilai Y

$P(x_i \mid y_j)$: *Posterior probability* dari X jika diketahui nilai Y

3. Dari nilai entropi yang sudah dihitung, diperoleh *Information Gain* pada persamaan 2.4.

$$IG(X \mid Y) = H(X) - H(X \mid Y) \quad (2.4)$$

4. Setelah didapatkan nilai *information gain*, untuk mengukur korelasi antar variabel maka digunakan *symmetrical uncertainty* (SU). *Symmetrical uncertainty* dirumuskan pada persamaan 2.5.

$$SU(X,Y) = 2\left[\frac{IG(X \mid Y)}{H(X) + H(Y)}\right] \quad (2.5)$$

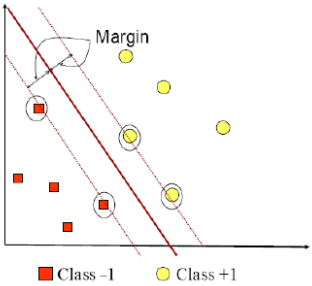
Nilai *symmetrical uncertainty* berkisar pada rentang 0 sampai dengan 1. Nilai 1 yang menunjukkan bahwa variabel X dan Y dependen, serta nilai 0 yang menunjukkan bahwa X dan Y adalah independen.

5. Setelah mendapatkan nilai SU untuk setiap variabel, langkah selanjutnya adalah mengurutkan nilai SU dan kemudian membandingkannya dengan nilai *threshold* untuk memilih variabel. Nilai threshold yang digunakan pada penelitian ini adalah 0,1 (berdasarkan penelitian Ladaya pada tahun 2017 menggunakan nilai *threshold* 0,1dengan data microarray, didapatkan hasil nilai akurasi yang lebih tinggi setelah dilakukan seleksi variabel). Apabila nilai SU variabel prediktor lebih dari nilai threshold maka variabel tersebut

dianggap relevan dan berkorelasi tinggi dengan variabel respon.

2.3 *Support vector Macine (SVM)*

Support vector macine (SVM) adalah metode pembelajaran supervised yang diperkenalkan pertama kali oleh Vapnik pada tahun 1995 dan sangat berhasil dalam melakukan prediksi, baik dalam kasus regresi maupun klasifikasi. Menurut Tan, Steinbach dan Kumar (2006), ide dasar dari SVM adalah menemukan fungsi pemisah (*hyperplane*) yang mampu memisahkan antara dua kelas dengan optimal. Optimal disini artinya adalah fungsi *hyperplane* mampu memisahkan kedua kelas dengan *margin* yang maksimal. *Margin* adalah jarak antara garis *hyperplane* dengan data terdekat dari kedua kelas. Bidang pembatas pertama membatasi kelas pertama dan bidang pembatas kedua membatasi kelas kedua. Sedangkan data yang berada pada bidang pembatas merupakan vektor-vektor yang terdekat dengan *hyperplane* terbaik disebut *support vector*. Ilustrasi dari *hyperplane* dengan *margin* yang maksimal dapat dilihat pada Gambar 2.2.



Gambar 2.2 *Hyperplane* dengan *Margin* Maksimal (Sitohang; 2012)

2.3.1 SVM Linier

Misalkan $\{x_1,\ldots,x_n\}$ adalah dataset dan $y_i \in \{+1,-1\}$ adalah label kelas dari data x_i dengan i = 1,2,…,n. Sehingga menurut Santosa (2007), *hyperplane* dinotasikan pada persamaan (2.6)

$$f(x) = \mathbf{w}^T\mathbf{x} + b \qquad (2.6)$$

Data x_i termasuk dalam kelas +1 bila $\mathbf{w}^T\mathbf{x}_i + b > 0$ dan data x_i termasuk dalam kelas -1 bila $\mathbf{w}^T\mathbf{x}_i + b < 0$. Karena data tersebut dipisahkan secara linier maka persamaan *hyperplane* dapat ditulis seperti persamaan (2.2)

$$\mathbf{w}^T\mathbf{x}_i + b = 0$$

dimana semua observasi harus memenuhi persamaan (2.7)

$$(\mathbf{w}^T\mathbf{x}_i) + b \geq 1 \text{ untuk } \mathrm{y_i} = +1$$
$$(\mathbf{w}^T\mathbf{x}_i) + b \leq -1 \text{ untuk } \mathrm{y_i} = -1 \qquad (2.7)$$

Yang dapat dikombinasikan menjadi satu pada persamaan (2.8)

$$y_i(\mathbf{w}^T\mathbf{x}_i) + b - 1 \geq 0 \text{ untuk i} = 1, 2, \ldots, n \qquad (2.8)$$

Menurut Santosa (2007), untuk mendapatkan *hyperplane* terbaik diperlukan nilai *margin* yang optimal. Nilai margin merupakan nilai jarak terdekat *hyperplane* dengan data yang paling dekat dengan *hyperplane* tiap kelas. Nilai *Margin* dapat dihitung dengan persamaan (2.9).

$$\frac{2}{\|\mathbf{w}\|} \qquad (2.9)$$

Nilai margin akan optimal apabila nilai ||**w**|| minimal. Oleh karena itu, pencarian *hyperplane* terbaik dengan nilai margin optimal dapat dirumuskan menjadi masalah optimasi pada persamaan (2.10).

$$\min \frac{1}{2} \| \mathbf{w} \|^2 \qquad (2.10)$$

dengan syarat $y_i(\mathbf{w}^T\mathbf{x}_i) + b \geq 1$, $i = 1, 2, \ldots, n$

Secara umum, persoalan optimasi konstrain pada persamaan (2.10) akan lebih mudah diselesaikan jika diubah ke dalam fungsi *Lagrange* pada persamaan (2.11).

$$L_p(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|^2 - \sum_{i=1}^{n} \alpha_i \{y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1\} \qquad (2.11)$$

Dimana $\alpha_i \geq 0$ yang merupakan nilai dari koefisien *lagrange*.

Nilai minimum w dan b dari fungsi Lag*range* tersebut ada pada persamaan (2.12).

$$\frac{\partial L_p(\mathbf{w},b,\alpha)}{\partial b} = 0 \Rightarrow \sum_{i=1}^{n} \alpha_i y_i = 0$$

$$\frac{\partial L_p(\mathbf{w},b,\alpha)}{\partial w} = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^{n} \alpha_i \mathbf{x}_i y_i \qquad (2.12)$$

Untuk mendapatkan dual problem, maka perlu menjabarkan dari persamaan (2.11) menjadi persamaan (2.13).

$$L_p(\mathbf{w},b,\alpha) = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \sum_{i=1}^{n} \alpha_i y_i (\mathbf{w}^T . \mathbf{x}_i) - b\sum_{i=1}^{n} \alpha_i y_i + \sum_{i=1}^{n} \alpha_i \qquad (2.13)$$

Dengan menggunakan nilai-nilai dari *w*, maka didapatkan

$$\mathbf{w}^T\mathbf{w} = \sum_{i=1}^{n} \alpha_i y_i \mathbf{w}^T \mathbf{x}_i = \sum_{i=1}^{n}\sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \qquad (2.14)$$

Maka persamaan (2.13) menjadi persamaan (2.15)

$$L_D(\mathbf{w},b,\alpha) = \sum_{i=1}^{n} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathbf{x}_i^T \mathbf{x}_j \qquad (2.15)$$

dengan batasan $\alpha_i \geq 0$ dimana $i = 1,2,\ldots,n$ dan $\sum_{i=1}^{n} \alpha_i y_i = 0$

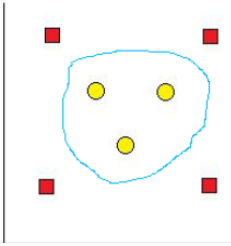
Data training dengan $\alpha_i > 0$ terletak pada *hyperplane* disebut *support vector*. Sedangkan data training yang tidak terletak pada *hyperplane* mempunyai $\alpha_i = 0$. Setelah ditemukan nilai α_i, maka kelas dari data yang akan diprediksi atau data testing dapat ditentukan berdasarkan nilai pada fungsi (2.16).

$$f(x) = \sum_{i,j=1}^{ns} \alpha_i y_i \mathbf{x}_i . \mathbf{x}_j + b \qquad (2.16)$$

2.3.2 SVM *Non-Linear*

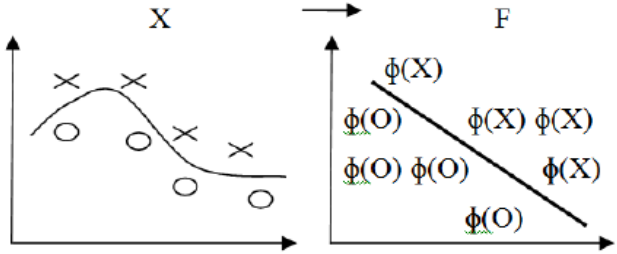
SVM memiliki prinsip dasar *linier classifier* yaitu kasus klasifikasi yang secara linier dapat dipisahkan, namun dalam penerapannya di lapangan seringkali ditemukan kasus-kasus yang tidak dapat dipecahkan oleh SVM *linear*. Sehingga model SVM

non-linier dikembangkan untuk mengatasi permasalahan ini. Pada Gambar 2.3 dapat dilihat bahwa SVM dapat menghasilkan *hyperplane* dengan persamaan non-linier.



Gambar 2.3 Ilustrasi Penyebaran Data dan *Hyperplane* Non-*linear* (Sitohang; 2012)

Ide dasar dari model SVM non-*linear* ini adalah memetakan data dari suatu bidang dengan dimensi tertentu, ke dalam bidang dengan dimensi yang lebih tinggi. Bidang tersebut kemudian dikenal dengan nama bidang *input* dan bidang *fitur* (F). Ilustrasi untuk memperlihatkan adanya permasalahan klasifikasi yang tidak dapat diselesaikan secara linier pada sampel data X dapat melihat Gambar 2.4



Gambar 2.4 Suatu Fungsi Kernel Mengubah Masalah yang Tidak Linier Menjadi Linier Dalam Ruang Baru (Sitohang; 2012)

SVM telah dikembangkan agar dapat menyelesaikan kasus klasifikasi dengan masalah non-linier dengan pendekatan kernel pada variabel data awal himpunan data. Fungsi kernel yang digunakan untuk memetakan dimensi awal (dimensi yang lebih

rendah) himpunan data ke dimensi baru (dimensi yang relatif lebih tinggi) yaitu $\Phi(\boldsymbol{x}_i).\Phi(\boldsymbol{x}_j)$.

Karena umumnya transformasi Φ ini tidak diketahui dan sangat sulit untuk dipahami secara mudah, maka perhitungan *dot product* dapat digantikan dengan fungsi kernel $K(\boldsymbol{x}_i, \boldsymbol{x}_j)$ yang mendefinisikan secara implisit transformasi Φ. Hal ini disebut dengan kernel trick yang dirumuskan sebagai berikut.

$$K(\boldsymbol{x}_i, \boldsymbol{x}_j) = \Phi(\boldsymbol{x}_i).\Phi(\boldsymbol{x}_j)$$

$$f(\Phi(\boldsymbol{x})) = \boldsymbol{w}.\Phi(\boldsymbol{x}) + b$$

$$= \sum_{i=1, \boldsymbol{x}_i \in SV}^{n} \alpha_i y_i K(\boldsymbol{x}, \boldsymbol{x}_i) + b$$

$$= \sum_{i=1, \boldsymbol{x}_i \in SV}^{n} \alpha_i y_i \Phi(\boldsymbol{x}).\Phi(\boldsymbol{x}_i) + b$$

Menurut Prasetyo (2012), fungsi kernel yang biasanya digunakan dalam literature SVM yaitu:

1. *Linear* = $K(\boldsymbol{x}_i, \boldsymbol{x})$
2. *Polinomial* Kernel = $K(\boldsymbol{x}_i, \boldsymbol{x}) = (\boldsymbol{x}, \boldsymbol{x}^T + d)^d$
3. *Gaussian Radian Basis Function* (RBF)
   $K(\boldsymbol{x}_i, \boldsymbol{x}) = exp(-\gamma(\|\boldsymbol{x} - \boldsymbol{x}_i\|^2))$
4. *Sigmoid* Kernel = $K(\boldsymbol{x}_i, \boldsymbol{x}) = tanh(\beta_0 \boldsymbol{x}^T \boldsymbol{x}_i + \beta_1)$

2.4 *Genetic Algorithm* (GA)

Menurut Sivandam (2008), genetic algorithm (GA) pertama kali diperkenalkan oleh John Holland dari Universitas Michigan pada awal 1970-an di New York, Amerika Serikat. John Holland bersama murid-murid serta rekan kerjanya lalu menghasilkan buku yang berjudul *"Adaption in Natural and Artificial Systems"* pada tahun 1975, yang cara kerjanya berdasarkan pada seleksi dan genetika alam. Konsep yang dipergunakan dalam algoritma genetika adalah mengikuti apa yang dilakukan oleh alam.

Genetic algorithm dimulai dari himpunan solusi yang dihasilkan secara acak. Himpunan ini disebut populasi, sedangkan setiap individu dalam populasi disebut kromosom. Setiap kromosom terdiri dari beberapa gen yang membentuk satu kromosom. Nilai dari gen tersebut dinamakan *allele* yang didapatkan berdasarkan inisialisasi awal. Kromosom - kromosom tertsebut berevolusi dalam suatu proses iterasi yang berkelanjutan yang disebut generasi. Pada setiap generasi, kromosom dievaluasi berdasarkan suatu fungsi evaluasi yang disebut dengan fungsi *fitness*. *Genetic algorithm* mempunyai tujuan untuk mencari nilai *fitness* yang optimal. Nilai *fitness* dari suatu kromosom akan menunjukkan kualitas dari kromosom dalam suatu populasi.

Pada GA untuk optimasi parameter, jumlah gen pada suatu kromosom adalah sebanyak parameter pada model yang akan dioptimasi. Pada klasifikasi menggunakan SVM dengan fungsi kernel RBF parameter yang digunakan ada 2 yaitu C dan γ. Pada metode GA, terdapat 5 komponen untuk membangun metode ini dengan rincian sebagai berikut.

1. Pengkodean (Inisialisasi)

Tahap pertama adalah inisialisasi kromosom. Sebelum melakukan inisialisasi terlebih dahulu adalah menentukan bentuk representasi kromosom. Dalam *genetic algorithm* terdapat berbagai macam represetasi kromosom mulai dari representasi biner, representasi integer, representasi *real code*, dan representasi permutasi. Setelah penentuan kromosom telah ditentukan langkah pertama dalam *genetic algorithm* adalah inisialisasi, yaitu proses pembangkitan nilai kromosom. Nilai yang dibangkitan secara acak sesuai dengan batasan untuk tiap gen. Selain itu juga akan ditentukan sejumlah *n* individu atau *popSize*.

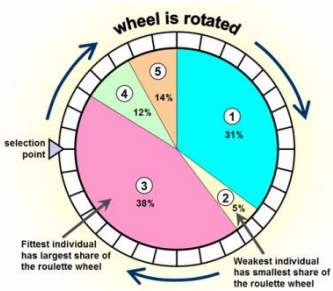
2. Nilai Fitness

Nilai *fitness* adalah ukuran performansi dari satu individu yang akan bertahan hidup. Di dalam evolusi alam, individu yang memiliki nilai *fitness* tinggi akan tetap diikutkan dalam iterasi berikutnya dan sebaliknya individu yang memiliki nilai *fitness*

rendah tidak dapat diikutkan dalam iterasi berikutnya. Fungsi *fitness* yang digunakan dalam penelitian ini adalah nilai ketepatan klasifikasi yaitu akurasi, sehingga individu yang akan bertahan ke generasi selanjutnya adalah individu yang memiliki nilai *fitness* tertinggi.

3. Proses Seleksi

Dalam penelitian ini proses seleksi menggunakan *roulette wheel selection* (RWS). Prinsip dari metode seleksi ini adalah setiap segmen *roulette* ditempati oleh masing-masing kromosom yang nantinya akan digunakan sebagai orang tua (*parent*) jika terpilih.



Gambar 2.5. Proses Seleksi dengan Metode RWS

Gambar 2.5 menggambarakan bahwa untuk besaran bagian *roulette* sesuai dengan rasio nilai *fitness* tiap individu dengan total nilai *fitness*. Semakin besar segmen *roulette* tentunya peluang individu tersebut terpilih juga semakin besar. Selain itu, pada proses seleksi ini juga digunakan juga *elitilism* untuk mempertahankan nilai *fitness* terbaik suatu generasi agar tidak turun di generasi selanjutnya. Dalam implementasinya, *elitilism* tersebut dilakukan dengan menyalin kromosom dengan *fitness* terbaik tersebut sebanyak yang diinginkan. Penentuan kromosom dengan *fitness* terbaik dengan melakukan pengurutan nilai *fitness*. Adapun proses seleksi dengan RWS adalah sebagai berikut.

a. Menghitung nilai fitness $f(h)$ untuk masing-masing kromosom
b. Menghitung total nilai fitness untuk kromosom

$$F = \sum_{h=1}^{n_{pop}} f(h)$$

c. Menghitung proporsi masing-masing kromosom

$$P_h = \frac{f(h)}{F}$$

d. Menghitung nilai kumulatif proporsi untuk masing-masing kromosom

$$S_h = \sum_{q=1}^{h} P_q$$

e. Jika $r \le S_1$, maka pilih kromosom v_1, lainnya pilih v_h, sehingga $S_{h-1} < r \le S_h$

f. Mengulangi tahapan v hingga semua kromosom yang berjumlah N terpilih semuanya.

4. Proses Pindah Silang (*Crossover*)

Proses crossover ini bertujuan untuk menghasilkan kromosom baru dari dua individu terpilih yang nantinya dijadikan sebagai orang tua (*parent*). Banyaknya jumlah orang tua tersebut ditentukan oleh parameter P_c (peluang *crossover*). Semakin besar nilai dari parameter P_c tersebut artinya akan semakin banyak pula orang tua yang terpilih. Berdasarkan hasil penelitian GA yang sudah pernah dilakukan sebaiknya nilai probabilitas pindah silang tinggi, yaitu antara 0,8 sampai dengan 0,9 agar memberikan hasil yang baik.

5. Proses Mutasi

Mutasi digunakan untuk mencegah algoritma yang terjebak pada solusi optimum dan melakukan tugasnya untuk mengembalikan atau membenahi material genetika yang hilang karena informasi acak genetika yang mengganggu. Proses mutasi cukup sederhana, jika bilangan random yang dibangkitkan kurang dari peluang mutasi yang ditentukan maka gen tersebut akan diubah menjadi kebalikannya. Nilai probabilitas mutasi menyatakan seberapa sering gen dalam kromosom akan mengalami mutasi. Proses mutasi ini bersifat acak sehingga tidak

menjamin akan diperoleh kromosom dengan *fitness* yang lebih baik setelah terjadinya mutasi tersebut. Sama seperti proses *crossover*, banyaknya gen yang mengalami mutasi dibatasi oleh suatu parameter tertentu. Dalam hal ini, parameter tersebut adalah P_m. Nilai P_m berpengaruh terhadap *fitness* yang dihasilkan. Nilai P_m yang besar dapat menyebabkan penurunan *fitness* suatu individu.

2.5 Naïve Bayes

Menurut Gorunescu (2011), teorema bayes adalah teorema yang digunakan dalam statistika untuk menghitung peluang suatu hipotesis, menghitung peluang dari suatu kelas dari masing-masing kelompok variabel yang ada, dan menentukan kelas mana yang paling optimal. Formula naïve bayes untuk klasifikasi dapat dilihat pada persamaan (2.17).

$$P(Y \mid X) = \frac{P(X \mid Y)P(Y)}{P(X)}, \tag{2.17}$$

dengan,

$P(Y \mid X)$:Probabilitas Y berdasarkan kondisi X (*posterior probability*)

$P(Y)$: Probabilitas hipotesis Y (*prior probability*)

$P(X \mid Y)$:Probabilitas X akan terjadi pada saat Y sudah diketahui

$P(X)$: Total probabilitas X

Klasifikasi *Naïve Bayes* merupakan klasifikasi berdasarkan teorema Bayes dan digunakan untuk menghitung probabilitas tiap kelas dengan asumsi tidak ada hubungan antar kelas (independen). Diberikan $(X_1, X_2, \ldots, X_p)$ merupakan variabel yang digunakan untuk menentukan kelas. Perhitungan *prosterior probability* untuk setiap kelas Y_j menggunakan teorema Bayes ada pada persamaan (2.18).

$$P(Y_j \mid X_1, X_2, \ldots, X_p) = \frac{P(X_1, X_2, \ldots, X_p \mid Y_j).\, P(Y_j)}{P(X_1, X_2, \ldots, X_p)} \tag{2.18}$$

Kelas yang dipilih adalah kelas yang memaksimalkan nilai $P(Y_j \mid X_1, X_2, \ldots, X_p)$ atau memaksimalkan $P(X_1, X_2, \ldots, X_p \mid Y_j).P(Y_j)$. Oleh karena itu untuk menghitung nilai $P(X_1, X_2, \ldots, X_p \mid Y_j)$ dapat dilihat pada persamaan (2.19), dimana setiap variabel diasumsikan saling bebas untuk kelas Y.

$$P(X_1, X_2, \ldots, X_p \mid Y) = P(X_1 \mid y) \,.\, P(X_2 \mid y) \,.\, \ldots P(X_p \mid y) \quad (2.19)$$

Umumnya, Bayes mudah dihitung untuk variabel bertipe kategorik seperti variabel jenis kelamin dengan kategori (pria dan wanita). Namun jika terdapat variabel numerik dapat dihitung menggunakan pendekatan distribusi normal yang ada pada persamaan (2.20). Estimasi $P(X_i \mid Y_j)$ dapat dihitung untuk setiap variabel X_i dan kelas Y_j sehingga data baru akan dapat diklasifikasikan ke dalam kelas Y_j dengan melihat nilai peluang yang didapatkan lebih besar.

$$P(X_i \mid Y_j) = \frac{1}{\sqrt{2\pi}\sigma_{ij}} \exp\left(-\frac{(X_i - \mu_{ij})^2}{2\sigma_{ij}^2} \right) \quad (2.20)$$

Pada Naïve Bayes digunakan *Hypothesis Maximum A Posterior* (HMAP) untuk memaksimalkan nilai probabilitas masing-masing kelas dengan menggunakan persamaan (2.21).

$$H_{MAP} = \arg\max \frac{P(X_1, X_2, \ldots, X_p \mid Y_j).P(Y_j)}{P(X_1, X_2, \ldots, X_p)} \quad (2.21)$$

2.6 Pemilihan Model Terbaik

Menurut Prasetyo (2012), akurasi dapat dihitung dari *confusion matrix* apabila data yang digunakan *balanced.* Tabel 2.1 merupakan *confusion matrix* berisi informasi tentang kelas data asli yang direpresentasikan pada baris matriks dan kelas data hasil prediksi suatu algoritma direpresentasikan pada kolom klasifikasi. Berkaitan dengan evaluasi performansi klasifikasi biner dapat dilihat dari akurasi.

Tabel 2.1 *Confusion Matrix*

Classifier	*Actual*	
	Actual Positive	*Actual Negative*
Classified Positive	*True Positive* (TP)	*False Positive* (FP)
Classified Negative	*False Negative* (FN)	*True Negative* (TN)

Ketepatan klasifikasi dapat dilihat dari akurasi klasifikasi pada persamaan 2.21.

$$\text{Akurasi Klasifikasi (\%)} = \frac{\text{TP} + \text{TN}}{\text{TP} + \text{FN} + \text{FP} + \text{TN}} \quad (2.21)$$

2.7 Microarray Data

Microarray merupakan salah satu teknologi yang memungkinkan peneliti untuk mengukur tingkat ekspresi dari ribuan gen secara bersamaan dalam satu pengamatan dan muncul sebagai perangkat penting dalam penelitian biomedis. Hasil pengukuran dari *microarray* tersebut biasanya dirangkum dalam daftar gen yang dinyatakan dalam dua kondisi atau diklasifikasikan berdasarkan fenotipnya. Menurut Yu dan Liu pada tahun 2011, *microarray data* merupakan jenis dari 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. Menurut Selvaraj dan Natarajan pada tahun 2011, dua metode umum yang dilakukan untuk menganalisis *microarray data* adalah *clustering* dan klasifikasi. Berdasarkan informasi yang dimiliki, *microarray* memiliki peranan penting dalam penelitian biomedis sebagai alat untuk identifikasi dan klasifikasi penyakit, khususnya kanker.

Data *microarray* diperoleh melalui suatu penelitian yang disebut *microarray experiment*. Langkah pertama yaitu dengan

mendapatkan mRNA dari sel yang akan diamati. Misalkan pada kasus tumor, sampel sel diamati dari sel yang terkena tumor dan sel normal. Selanjutnya mRNA yang telah diperoleh akan dikonversikan menjadi cDNA menggunakan enzim *reverse tranciptase*. Dengan menggunakan *fluorescent*, cDNA dari sel tumor ditandai dengan warna merah dan cDNA dari sel normal ditandai dengan warna hijau. Sampel kemudian mengalami hibridisasi, yaitu cDNA saling mengikat terhadap DNA. Setelah mengalami hibridisasi, sampel dipindai untuk mengukur ekspresi setiap gen melalui *fluorescence* yang terkandung. Intensitas *fluorescence* berhubungan dengan jumlah cDNA dalam sampel untuk gen tersebut. Titik yang bersinar merah terang adalah gen yang sangat diekspresikan dalam sel tumor, sedangkan titik yang bersinar hijau terang merupakan gen yang sangat diekspresikan ke dalam sel normal. 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 diinterpretasikan. Titik-titik warna harus dirubah menjadi sebuah nilai tertentu untuk selanjutnya dapat dianalisis.

2.8 Kanker Prostat

Kanker prostat berkembang di prostat seorang pria, kelenjar kenari berukuran tepat di bawah kandung kemih yang menghasilkan beberapa cairan dalam air mani. Ini adalah kanker paling umum pada pria setelah kanker kulit. Kanker prostat sering tumbuh sangat lambat dan tidak dapat menyebabkan kerusakan signifikan. Tetapi beberapa jenis lebih agresif dan dapat menyebar dengan cepat tanpa pengobatan. Pada tahap awal, pria mungkin tidak memiliki gejala. Kemudian, selanjutnya terdapat gejala-gejala sebagai berikut :

1. Sering buang air kecil, terutama pada malam hari
2. Kesulitan memulai atau menghentikan buang air kecil
3. Aliran kencing lemah atau terputus
4. Nyeri atau sensasi terbakar pada saat kencing atau ejakulasi
5. Darah dalam urin atau air mani

Kanker stadium lanjut dapat menyebabkan rasa sakit yang dalam di punggung bawah, pinggul, atau paha atas. Sehingga untuk melihat seberapa jauh kanker prostat menyebar sudah terdapat 4 tingkatan sebagai berikut.

1. Stadium I: Kanker kecil dan masih dalam prostat.
2. Stadium II: kanker lebih maju, tetapi masih terbatas pada prostat.
3. Stadium III: kanker telah menyebar ke bagian luar prostat dan vesikula seminalis dekatnya.
4. Stadium IV: kanker telah menyebar ke kelenjar getah bening, organ terdekat atau jaringan seperti kandung kemih atau rektum, atau organ jauh seperti tulang atau paru-paru.

BAB III
METODOLOGI PENELITIAN

3.1 Sumber Data

Data yang digunakan dalam penelitian ini adalah data sekunder, yang merupakan data *microarray gene expression* dari Jurnal Singh, dkk (2002) yang berjudul "*Gene Expression Correlates Of Clinical Prostate Cancer Behavior*" untuk mengklasifikasikan seseorang ke dalam golongan kanker dan bukan kanker. Data yang sudah terkumpul dibagi menjadi dua yaitu data testing dan data training dengan *k-fold cross validation*. Data training digunakan untuk mencari parameter terbaik, sedangkan untuk mencari metode terbaik dapat membandingkan nilai akurasi dari data testing.

3.2 Struktur Data

Pada penelitian ini variabel yang digunakan untuk mengklasifikasikan seseorang terkena kanker prostat atau tidak, sebanyak 6.033 variabel. Sedangkan untuk variabel responnya adalah data *binary* dimana jika seseorang terkena kanker maka akan dikategorikan 1, akan tetapi apabila tidak terkena kanker maka dikategorikan 0. Jumlah observasi yang digunakan pada data prostat ada sebanyak 102 orang. Struktur data yang digunakan untuk variabel penelitian klasifikasi *microarray* kanker prostat menggunakan metode *Hybrid Support vector Macine – Genetic Algorithm* (SVM-GA) dan *naïve bayes* dapat dilihat pada Tabel 3.1.

Tabel 3.1 Struktur Data Kanker Prostat

Y_i	X_1	X_2	…	$X_{6.033}$
Y_1	$X_{1(1)}$	$X_{2(1)}$	…	$X_{6.033(1)}$
Y_2	$X_{1(2)}$	$X_{2(2)}$	…	$X_{6.033(2)}$
⋮	⋮	⋮	…	⋮
Y_n	$X_{1(102)}$	$X_{2(102)}$	…	$X_{6.033(102)}$

3.3 Langkah Analisis

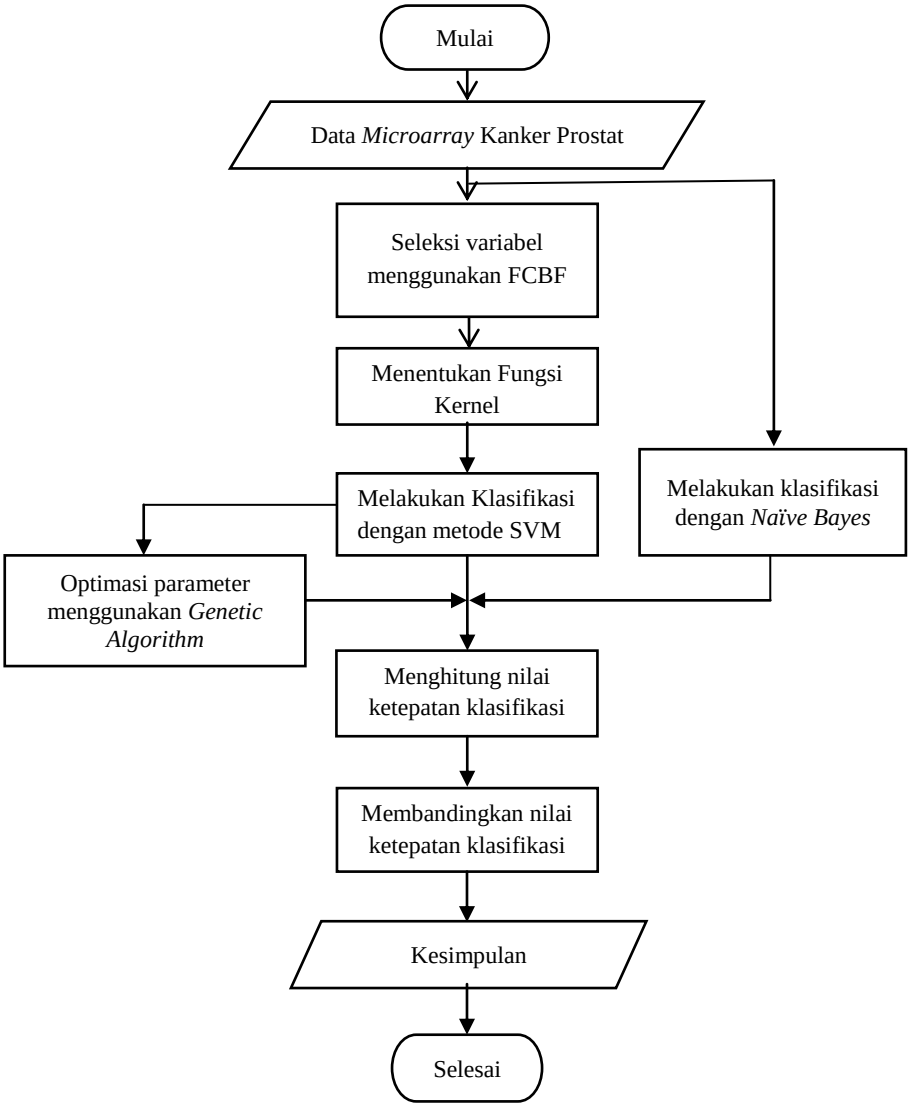
Metode analisis yang digunakan untuk menjawab tujuan dari penelitian ini adalah metode *support vector macine* (SVM) dengan *genetic algorithm* (GA) dan *naïve bayes*. SVM merupakan konsep klasifikasi yang dilakukan dengan cara mencari *hyperplane* (batas keputusan) terbaik sebagai fungsi pemisah dua buah kelas data. Sedangkan *naïve bayes* merupakan konsep klasifikasi yang dilakukan dengan mencari nilai probabilitas yang paling tinggi. Tahapan yang dilakukan dalam analisis ini adalah sebagai berikut:

1. Melakukan seleksi variabel dengan metode FCBF
2. Membagi data menggunakan *k-fold cross validation* dengan k = 10
3. Menganalisis klasifikasi menggunakan metode *Support vector Macine*.
4. Melakukan optimasi dengan metode SVM-GA pada data training dengan langkah-langkah sebagai berikut.
    a. Menyusun kromosom dengan membangkitkan 100 kromosom. Kromosom yang dibangkitkan terdiri dari dua gen yang merupakan parameter dari SVM. Nilai inisial kromosom diperoleh dari nilai parameter yang optimal pada metode SVM.
    b. Mengevaluasi masing-masing kromosom berdasarkan nilai *fitness* yaitu nilai ketepatan klasifikasi.
    c. Melakukan proses seleksi sebanyak 100 kromosom dari sejumlah 100 *parent* yang berasal dari populasi dengan seleksi *roulette wheel selection* (RWS).
    d. Melakukan proses pindah silang *(crossover)* jika nilai bilangan *random r* antara [0,1] yang dibangkitkan kurang dari probabilitas pindah silang (P_c).
    e. Melakukan proses mutasi jika nilai bilangan *random r* antara [0,1] yang dibangkitkan kurang dari nilai probabilitas proses mutasi (P_m).

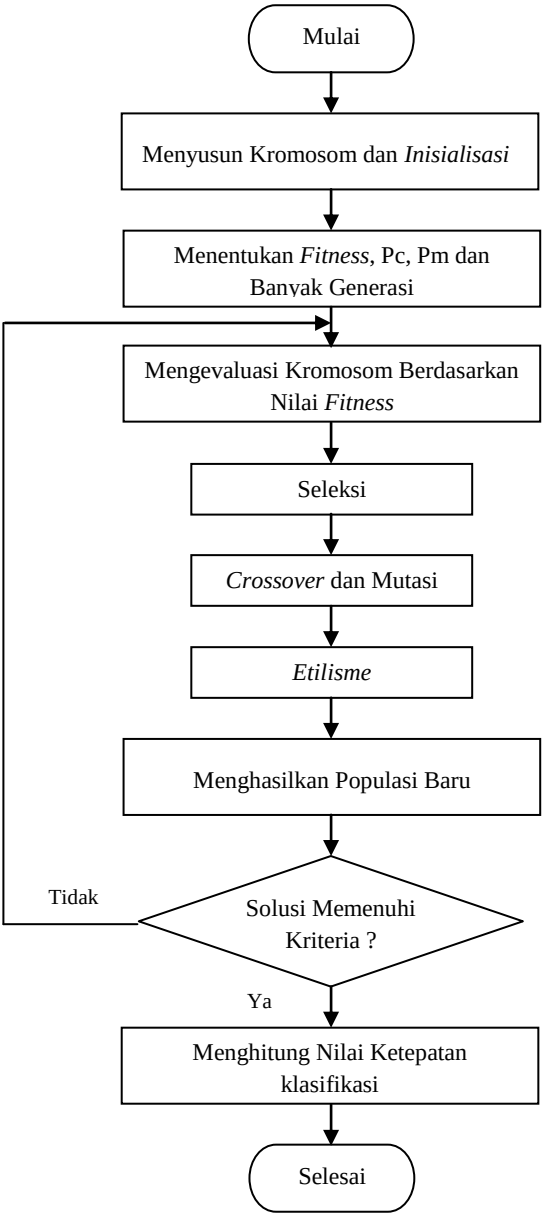
f. Melakukan proses elitisme dimana tiga kromosom dengan nilai *fitness* terbaik akan bertahan ke generasi selanjutnya.
g. Melakukan pergantian populasi lama dengan generasi baru dengan cara memilih kromosom terbaik berdasarkan nilai *fitness*-nya dari tahapan proses seleksi sampai dengan proses elitisme.
h. Melakukan pengecekan untuk solusi yang telah didapatkan. Solusi dikatakan telah memenuhi kriteria apabila nilai *fitness* terbaik telah konvergen, apabila belum terpenuhi maka perlu mengulangi proses dari tahap d hingga mendapatkan nilai *fitness* yang konvergen.

5. Melakukan klasifikasi dengan menggunakan metode *naïve bayes*, baik dengan semua variabel independen maupun dengan variabel hasil dari seleksi variabel menggunakan *genetika algorithm* (GA). Berikut ini adalah langkah-langkah klasifikasi menggunakan naïve bayes.
   a. Menghitung nilai probabilitas untuk setiap variabel ke-i, dimana i = 1,2,…,k. Apabila data yang digunakan adalah data numerik (kontinyu), maka akan dihitung nilai rata-rata, varian dan standar deviasi dari masing-masing variabel di setiap kategori . Selanjutnya adalah menentukan nilai probabilitas mengunakan pendekatan distribusi normal.
   b. Menentukan probabilitas akhir dari semua variabel independen untuk setiap kategori.
   c. Menentukan kelas berdasarkan nilai probabilitas yang tertinggi.
   d. Menghitung ketepatan klasifikasi.
6. Membandingkan nilai ketepatan klasifikasi dari metode *Support vector Macine* (SVM), *Support vector Macine-Genetic Algorithm* (SVM-GA), *Naïve Bayes Classifier* dan *Naïve Bayes Genetic Algorithm*.
7. Menarik kesimpulan dan memberi saran

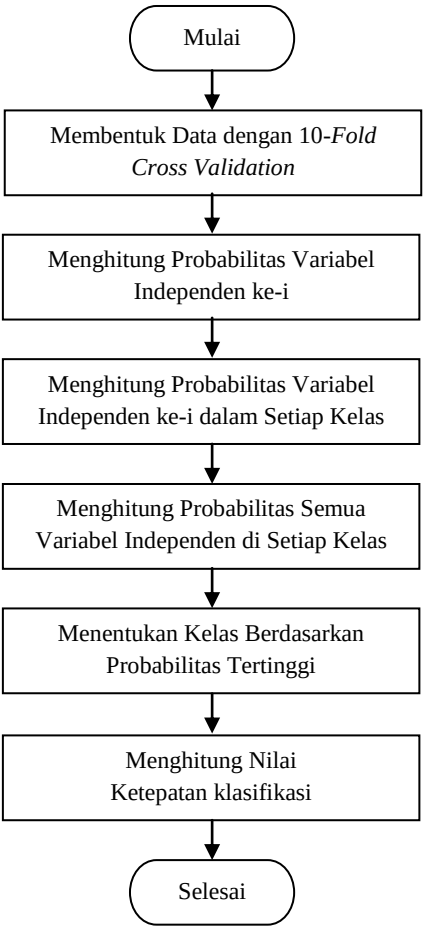
Langkah-langkah analisis tersebut digambarkan dalam diagram alir yang dapat dilihat pada Gambar 3.1.



Gambar 3.1 Diagram Alir Penelitian



Gambar 3.3 Diagram Alir *Genetic Algorithm* (GA)



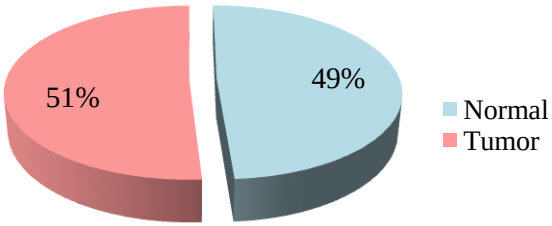
Gambar 3.4 Diagram Alir *Naïve Bayes*

BAB IV
ANALISIS DAN PEMBAHASAN

Pada bab ini akan dibahas dan dijelaskan mengenai langkah-langkah dalam klasifikasi seseorang terkena kanker prostat. Pembahasan pertama akan dilakukan identifikasi data untuk mengetahui karakteristik data yang dijelaskan menggunakan statistika deskriptif. Metode yang digunakan yaitu *support vector macine* (SVM), SVM dengan *Genetic Algorithm* (SVM-GA) dan *naïve bayes*. Hasil klasifikasi pada ketiga model tersebut, akan dibandingkan untuk memperoleh hasil klasifikasi terbaik.

4.1 Karakteristik Gen Prostat

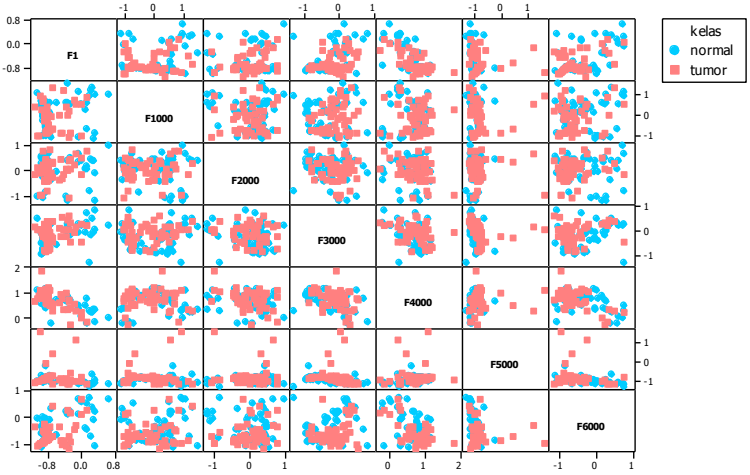
Data yang digunakan terdiri dari dua kelas yaitu kelas positif dan kelas negatif, dimana kelas negatif merupakan kelas dengan status pasien yang terkena tumor dan kelas positif merupakan kelas dengan status pasien yang normal. Secara visual data kanker prostat termasuk data yang *balanced*, dikarenakan hasil dari nilai proporsi antar kelas tidak jauh berbeda. Hal tersebut telah digambarkan pada Gambar 4.1.



Gambar 4.1 Proporsi Jumlah Pasien Berdasarkan Status Penyakit

Selain itu, karakteristik data klasifikasi juga dapat melihat pola persebaran data dari setiap variabel dan kelas. Apabila dilihat dari jumlah variabel (gen) sejumlah 6033 yang digunakan sebagai penilaian untuk menentukan seseorang masuk dalam kelas negatif atau positif, maka dapat dipastikan bahwa pola persebaran data menjadi sangat kompleks. Ilustrasi gambaran

persebaran data dari beberapa variabel telah digambarkan pada Gambar 4.2.



Gambar 4.2 Persebaran Data dari Beberapa Variabel

Gambar 4.2 menjelaskan bahwa persebaran data untuk beberapa variabel dari data kanker prostat terlihat sangat kompleks. Hal ini dikarenakan pada Gambar 4.2 dapat dilihat dari masing-masing kelas untuk setiap variabelnya hampir menyatu, sehingga untuk memisahkan pasien berdasarkan kelasnya tidak dapat dipisahkan secara linier namun dilakukan secara non linier dengan menggunakan bantuan kernel pada metode *support vector macine* (SVM).

4.2 Seleksi Variabel dengan *Fast Correlation Based Filter* (FCBF)

Pada penelitian ini dilakukan seleksi variabel dengan menggunakan metode *fast correlation based filter* (FCBF). Proses seleksi variabel dilakukan karena data yang digunakan merupakan *high dimensional* data yaitu data dengan jumlah variabel sangat besar namun ukuran sampel kecil. Adanya seleksi variabel diharapkan mampu meningkatkan nilai ketepatan klasifikasi serta

mempercepat proses pemodelan. Pemilihan variabel dilakukan berdasarkan nilai threshold yaitu nilai batas korelasi yang merupakan nilai minimum dari nilai *symetrical uncertainty* (SU). Nilai threshold berada pada *range* 0 sampai dengan 1, dimana nilai threshold yang digunakan pada penelitian ini adalah 0,1. Variabel yang terseleksi telah ditampilkan pada Tabel 4.1.

Tabel 4.1 Variabel yang Telah Terseleksi

No	Variabel	No	Variabel	No	Variabel
1	X_{1881}	11	X_{2695}	21	X_{4849}
2	X_{1897}	12	X_{2852}	22	X_{5016}
3	X_{1903}	13	X_{3037}	23	X_{5177}
4	X_{1998}	14	X_{3118}	24	X_{5230}
5	X_{2025}	15	X_{3429}	25	X_{5278}
6	X_{2215}	16	X_{3518}	26	X_{5485}
7	X_{2238}	17	X_{4212}	27	X_{5639}
8	X_{2619}	18	X_{4266}	28	X_{5663}
9	X_{2634}	19	X_{4335}	29	X_{6029}
10	X_{2694}	20	X_{4391}		

Tabel 4.1 menunjukkan bahwa hasil dari seleksi variabel dengan menggunakan metode FCBF didapatkan 29 variabel. Hal ini menunjukkan bahwa tidak semua variabel sesuai dengan kelasnya. Variabel yang terseleksi adalah variabel yang memiliki nilai SU diatas nilai threshold. Selanjutnya variabel yang telah terseleksi dapat digunakan pada tahapan klasifikasi.

Sebelum melakukan klasifikasi, dilakukan pembagian data dengan 10-fold cross validation untuk memvalidasi nilai akurasi. Hal ini dikarenakan jumlah observasi cukup banyak dan didukung pada penelitian sebelumnya yang didapatkan kesimpulan dengan 10-fold terbukti mendapatkan hasil klasifikasi yang lebih stabil. Data testing yang digunakan setelah membagi data menjadi 10 fold telah ditampilkan pada Tabel 4.2.

Tabel 4.2 Pembagian Data Testing Dengan 10-*Fold Cross Validation*

Fold	Observasi ke-										
	1	2	3	4	5	6	7	8	9	10	11
1	6	42	34	37	18	**54**	**93**	**102**	**71**	**99**	**92**
2	31	41	33	26	20	**66**	**52**	**78**	**76**	**72**	**67**
3	30	11	5	23	14	**86**	**63**	**89**	**81**	**90**	
4	48	35	2	45	46	**75**	**51**	**96**	**61**	**83**	
5	40	38	32	3	27	**58**	**60**	**53**	**82**	**80**	
6	29	47	21	12	49	**74**	**77**	**65**	**79**	**85**	
7	1	43	13	16	7	**73**	**62**	**88**	**57**	**59**	
8	10	9	39	4	36	**84**	**68**	**95**	**55**	**97**	
9	28	50	19	15	25	**98**	**91**	**94**	**70**	**56**	
10	22	8	44	17	24	**87**	**69**	**101**	**100**	**64**	

Ket : Huruf yang bercetak tebal menandakan observasi yang termasuk dalam kategori tumor

Tabel 4.2 menunjukkan bahwa jumlah data untuk setiap fold tidak sama, dikarenakan pada data *microarray* kanker prostat mempunyai 102 pasien. Data testing pada Tabel 4.2 ini yang akan digunakan untuk membandingkan metode mana yang menghasilkan akurasi tertinggi untuk klasifikasi data microarray kanker prostat, sedangkan data training digunakan untuk mencari *range* optimum dengan melihat rata-rata nilai akurasi yang paling kecil.

4.3 Klasifikasi dengan *Support vector Macine*

Algoritma *support vector macine* pada penelitian ini menggunakan kernel *Radial Basis Function* (RBF) dalam menentukan *hyperplane* terbaik untuk klasifikasi data. Parameter yang dibutuhkan dalam RBF adalah parameter *cost* (C) dan gamma (γ). Penentuan parameter terbaik dilakukan dengan metode *Grid Search*. Menurut Yao (2014), algoritma *Grid Search* membagi *range* pencarian parameter yang akan dioptimalkan ke dalam *grid* dan melintasi semua titik *grid* untuk mendapatkan

nilai optimal. Data yang digunakan untuk mencari *range* yang optimal adalah data training. Nilai parameter yang optimal pada metode klasifikasi SVM *Grid Search* yaitu dilihat pada *range* yang optimum.

Pada metode ini juga akan dilakukan dua jenis analisis yaitu analisis menggunakan semua variabel dan analisis yang hanya menggunakan variabel yang telah terseleksi menggunakan FCBF. Akan tetapi dalam GA yang digunakan untuk optimasi diperlukan *range* terbaik dari model, sehingga dilakukan klasifikasi metode *Grid Search* svm untuk mendapatkan *range* dari nilai parameter yang paling optimum. Menurut Hsu, Chang dan Lin (2016), klasifikasi SVM dengan metode *Grid Search* menggunakan kernel RBF didapatkan *range* yang terbaik yaitu untuk parameter *cost* dari *range* $-5,-3,-1,\ldots,13,15$ dan untuk parameter γ dari *range* $-15,-13,-11,\ldots,1,3$.

4.3.1 SVM Tanpa Seleksi Variabel

Percobaan parameter *cost* dan γ menghasilkan nilai akurasi yang berbeda karena terdapat 15 kombinasi untuk setiap kali percobaan, sehingga akan dilihat dari setiap fold nilai akurasi yang terbaik. Pemilihan *range* parameter yang optimum berdasarkan nilai rata-rata akurasi dari 10 fold paling maksimum dapat dilihat pada Tabel 4.2.

Tabel 4.3 Kombinasi *Range* Parameter Untuk *Grid Search* Tanpa Seleksi

Parameter		Fold					Akurasi
Cost	Gamma	1	2	⋯	9	10	
$2^{-5}-2^{-1}$	$2^{-15}-2^{-9}$	0.8589	0.8456	⋯	0.8589	0.8500	0.8467
	$2^{-9}-2^{-3}$	0.5511	0.5144	⋯	0.6078	0.5767	0.5920
	$2^{-3}-2^{3}$	0.4500	0.5067	⋯	0.3911	0.3911	0.4028
2^{-1} - 2^{3}	**2^{-15} - 2^{-9}**	0.9444	0.9222	⋯	0.9344	0.9344	**0.9327**
	$2^{-9}-2^{-3}$	0.8456	0.8778	⋯	0.8900	0.8600	0.8661
	$2^{-3}-2^{3}$	0.4633	0.4622	⋯	0.3911	0.3911	0.4076
$2^{3}-2^{7}$	$2^{-15}-2^{-9}$	0.9456	0.9111	⋯	0.9133	0.9367	0.9277

	$2^{-9}-2^{-3}$	0.8022	0.8233	⋯	0.8600	0.8922	0.8468
	$2^{-3}-2^{3}$	0.5067	0.4511	⋯	0.3600	0.4622	0.4196

Ket : Huruf yang bercetak tebal menandakan *range* yang paling optimum

Tabel 4.3 Kombinasi *Range* Parameter Untuk *Grid Search* Tanpa Seleksi (Lanjutan)

Parameter		Fold					Akurasi
Cost	Gamma	1	2	⋯	9	10	
$2^{7}-2^{11}$	$2^{-15}-2^{-9}$	0.9133	0.9033	⋯	0.9022	0.9256	0.9191
	$2^{-9}-2^{-3}$	0.8344	0.8467	⋯	0.8600	0.8933	0.8561
	$2^{-3}-2^{3}$	0.4844	0.3411	⋯	0.3600	0.3811	0.3959
$2^{11}-2^{15}$	$2^{-15}-2^{-9}$	0.9111	0.9033	⋯	0.9022	0.8911	0.9150
	$2^{-9}-2^{-3}$	0.8467	0.8467	⋯	0.8600	0.8478	0.8546
	$2^{-3}-2^{3}$	0.5389	0.3411	⋯	0.3600	0.3600	0.3890

Tabel 4.3 menunjukkan bahwa menggunakan kernel RBF dari 15 kombinasi, didapatkan *range* yang optimum yaitu *range cost* -1 sampai dengan 3 dan *range* γ -15 sampai dengan -9 untuk mencari parameter terbaik. Hal ini dikarenakan *range* tersebut memiliki rata-rata akurasi terbesar dibandingkan kombinasi lainnya sebesar 0,9327.

Setelah didapatkan *range* yang optimum dari 15 kombinasi, maka langkah selanjutnya adalah mencari parameter yang optimum berdasarkan *range* yang terpilih dengan melihat nilai akurasi setiap fold. Hasil perhitungan akurasi pada data training untuk setiap fold dapat dilihat pada Tabel 4.4.

Tabel 4.4 Hasil Percobaan *Grid Search* SVM Tanpa Seleksi Variabel

Fold	Parameter Opt		Akurasi
	Cost	Gamma	
1	2	-13	0.9444
2	3	-14	0.9222

3	3	-15	0.9267
4	3	-14	0.9244
5	3	-14	0.9344
6	2	-13	0.9256
7	3	-14	0.9222
8	**3**	**-14**	0.9578

Ket : Huruf yang bercetak tebal menandakan parameter yang paling optimum

Tabel 4.4 Hasil Percobaan *Grid Search* SVM Tanpa Seleksi Variabel (Lanjutan)

Fold	Parameter Opt		Akurasi
	Cost	Gamma	
9	1	-12	0.9344
10	3	-15	0.9344

Tabel 4.4 menunjukkan bahwa dengan *range* optimum yang terpilih, didapatkan hasil yang berbeda untuk parameter yang optimum di setiap fold. Nilai parameter yang dipilih dilihat berdasarkan nilai akurasi terbesar dari fold 1 sampai dengan fold 10. Nilai akurasi tertinggi ada pada fold ke-8 yaitu sebesar 0,9578 dengan nilai parameter untuk *cost* yang optimum sebesar 3 dan nilai parameter γ yang optimum sebesar -14. Setelah parameter *cost* dan γ yang paling optimum telah didapatkan, kemudian didapatkan fungsi *hyperplane* yang terbentuk untuk klasifikasi pada data kanker prostat menggunakan metode *Grid Search* SVM adalah

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \exp(-0,000061 \| x_i - x \|^2) + b$$

dengan $0 \le \alpha_i \le 8$, untuk i=1,2,…,n.

Berdasarkan fungsi *hyperplane* yang sudah didapatkan, kemudian dilakukan klasifikasi pada data testing untuk semua fold dengan nilai parameter yang telah optimum unuk dilakukan evaluasi. Hal ini dilakukan untuk mengevaluasi efektivitas metode dan model yang telah dibuat. Ukuran yang digunakan untuk evaluasi yaitu akurasi klasifikasi, yang dapat dihitung pada Tabel 4.5

Tabel 4.5 Konfusi *Matrix Grid Search* SVM Fold ke-1 Tanpa Seleksi

Aktual	Prediksi	
	Normal	Tumor
Normal	5	1
Tumor	0	5

Tabel 4.5 menjelaskan bahwa pada fold ke-1 dengan menggunakan nilai *cost* 8 dan γ 0,000061 maka seseorang diklasifikasikan benar terkena tumor yaitu sebanyak 5 orang, selain itu juga untuk klasifikasi seseorang benar tidak terkena tumor (normal) yaitu sebanyak 5 orang. Sedangkan untuk kesalahan klasifikasi pada fold ke-1 hanya terdapat 1 kesalahan pada saat seharusnya seseorang diklasifikasikan normal, akan tetapi masuk dalam kategori tumor. Untuk mengetahui nilai akurasi pada data testing untuk semua fold, maka dapat dilihat pada Tabel 4.6.

Tabel 4.6 Nilai Akurasi *Grid Search* SVM Tanpa Seleksi (Testing)

Fold	Akurasi
1	0.91
2	1
3	0.9
4	0.9
5	0.9
6	1
7	1
8	0.7
9	1
10	0.9

Tabel 4.6 menunjukkan hasil akurasi dari semua fold menggunakan parameter yang optimum. Nilai akurai yang didapatkan sudah cukup tinggi, apabila dirata-rata mendapatkan hasil nilai akurasi sebesar 0,92%.

4.3.2 SVM dengan Seleksi Variabel

Selain dilakukan klasifikasi menggunakan seluruh variabel, juga dilakukan klasifikasi dengan variabel yang telah terseleksi dengan FCBF. Hasil klasifikasi menggunakan SVM *Grid Search* dengan seleksi variabel mempunyai kesimpulan yang berbeda dibandingkan SVM tanpa seleksi variabel. Hasil klasifikasi dengan metode *Grid Search* menggunakan 29 variabel telah ditabelkan pada Tabel 4.7 .

Tabel 4.7 Kombinasi *Range* Parameter *Grid Search* Dengan Seleksi

Parameter		Fold					Akurasi
Cost	Gamma	1	2	…	9	10	
	$2^{-15} - 2^{-9}$	0.8467	0.7667	…	0.9011	0.9156	0.8461
$2^{-5} - 2^{-1}$	$2^{-9} - 2^{-3}$	0.9667	0.9456	…	0.9456	0.9556	0.9552
	$2^{-3} - 2^{3}$	0.9578	0.9456	…	0.9444	0.9667	0.9529
	2^{-15} - 2^{-9}	0.9556	0.9444	…	0.9456	0.9567	0.9518
2^{-1} - 2^{3}	**2^{-9} - 2^{-3}**	0.9667	0.9567	…	0.9567	0.9667	**0.9598**
	$2^{-3} - 2^{3}$	0.9556	0.9333	…	0.9356	0.9456	0.9489
	$2^{-15} - 2^{-9}$	0.9667	0.9456	…	0.9567	0.9678	0.9596
$2^{3} - 2^{7}$	$2^{-9} - 2^{-3}$	0.9667	0.9456	…	0.9556	0.9667	0.9572
	$2^{-3} - 2^{3}$	0.9556	0.9233	…	0.9333	0.9456	0.9399
	$2^{-15} - 2^{-9}$	0.9556	0.9567	…	0.9444	0.9667	0.9569
$2^{7} - 2^{11}$	$2^{-9} - 2^{-3}$	0.9556	0.9333	…	0.9456	0.9444	0.9441
	$2^{-3} - 2^{3}$	0.9556	0.9233	…	0.9344	0.9467	0.9446
	$2^{-15} - 2^{-9}$	0.9578	0.9456	…	0.9456	0.9578	0.9502
$2^{11} - 2^{15}$	$2^{-9} - 2^{-3}$	0.9456	0.9344	…	0.9333	0.9567	0.9413
	$2^{-3} - 2^{3}$	0.9556	0.9122	…	0.9444	0.9444	0.9377

Ket : Huruf yang bercetak tebal menandakan *range* yang paling optimum

Tabel 4.7 menunjukkan bahwa dengan menggunakan 29 variabel *range* yang mempunyai rata-rata paling tinggi ada pada *range* -9 sampai -3 untuk γ , sedangkan untuk *range cost* sama seperti sebelumnya yaitu ada di *range* -1 sampai 3. Setelah ditentukan *range* yang optimum, maka dapat dilihat fold mana yang menghasilkan nilai akurasi tertinggi dari data testing.

Tabel 4.8 Hasil Percobaan *Grid Search* SVM dengan Seleksi Variabel

Fold	Parameter Opt	Akurasi

	Cost	Gamma	
1	1	-8	0.9667
2	1	-5	0.9567
3	0	-9	0.9567
4	3	-9	0.9556
5	1	-9	0.9567
6	2	-9	0.9567

Tabel 4.8 Hasil Percobaan *Grid Search* SVM dengan Seleksi Variabel (Lanjutan)

Fold	Parameter Opt		Akurasi
	Cost	Gamma	
7	3	-8	0.9578
8	**1**	**-9**	0.9678
9	3	-8	0.9567
10	3	-8	0.9667

Ket : Huruf yang bercetak tebal menandakan parameter yang paling optimum

Fold yang mempunyai nilai akurasi tertinggi dengan *range* yang optimum berada pada fold ke-8 yang dapat dilihat pada Tabel 4.8. Parameter yang optimal untuk *cost* sebesar 2 dan parameter γ sebesar 0,00195. Sehingga fungsi *hyperplane* yang terbentuk untuk klasifikasi pada data kanker prostat menggunakan metode *grid search* SVM dengan 29 variabel adalah

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \exp(-0,00195 \| x_i - x \|^2) + b$$

dengan x adalah *vector* observasi dari variabel independen dimana n adalah jumlah observasi. α_i merupakan *vector* dengan nilai $0 \le \alpha_i \le 2$, sedangkan y_i adalah suatu tandadan b adalah suatu konstanta yang disebut bias. Nilai α_i dan b pada fungsi hyperplane yang terbentuk dapat dilihat pada Lampiran 5. Setelah didapatkan parameter optimal maka parameter tersebut yang digunakan untuk klasifikasi data testing setiap *fold.* Hasil akurasi

klasiifikasi data testing dengan menggunakan parameter yang optimal telah ditampilkan pada Tabel 4.9.

Tabel 4.9 Nilai Akurasi *Grid Search* SVM dengan Seleksi (Testing)

Fold	Akurasi
1	0.9091
2	1
3	1

Tabel 4.9 Nilai Akurasi *Grid Search* SVM dengan Seleksi (Testing) (Lanjutan)

Fold	Akurasi
4	1
5	0.9
6	1
7	1
8	0.8
9	1
10	0.9

Klasifikasi menggunakan 29 variabel mempunyai hasil nilai akurasi yang lebih tinggi sebesar 3% dibandingkan dengan menggunakan seluruh variabel. Rata-rata nilai akurasi dari klasifikasi data testing dengan parameter yang optimal sebesar 0,95% yang dihitung dari Tabel 4.9. Selain itu juga dari Tabel 4.9 menunjukkan bahwa nilai akurasi untuk setiap foldnya sudah baik untuk klasifikasi dikarenakan nilai akurasi terendah yaitu sebesar 0,8 pada fold ke-8. Apabila dihitung nilai AUC pada fold tersebut didapatkan nilai sebesar 0,8571, artinya model klasifikasi yang digunakan sudah baik.

4.4 Optimasi Parameter *Support Vector Macine* (SVM) Menggunakan Genetic Algorithm (GA)

Pada klasifikasi SVM telah ditentukan nilai parameter yang optimal, oleh karena itu langkah selanjutnya adalah mendapatkan nilai parameter yang optimal menggunakan metode Genetic

Algorithm (GA). Adanya penambahan algoritma GA bertujuan untuk mendapatkan parameter SVM yang mempunyai nilai akurasi lebih tinggi. SVM-GA ini akan menggunakan *range* dari nilai parameter yang optimal pada SVM.

Langkah pertama untuk melakukan optimasi parameter GA adalah melakukan inisialisasi kromosom sebanyak 100 yang memiliki nilai estimasi parameter. Parameter yang digunakan yaitu C dan γ, sehingga kromosom yang telah dibangkitkan sejumlah 100 akan memiliki dua gen. Misalkan nilai parameter C berada pada *range* 0,5 samapai dengan 8 dan parameter γ berada pada *range* 0,125 sampai dengan 8, maka ilustrasi kromosom awal yang terbentuk dapat dilihat pada Gambar 4.3.

Parameter	C	γ
Kromosom	8	0,125

Gambar 4.3 Ilustrasi Kromosom Awal Optimasi GA

Gambar 4.3 merupakan salah satu nilai inisialisasi pada kromosom pertama yang terdiri dari dua gen. Kromosom yang telah terbentuk akan mengikuti proses GA, meliputi seleksi, pindah silang, mutasi dan elitisme. Sebagai acuan dalam proses GA dapat melihat nilai *fitness*. *Fitness* yang digunakan adalah nilai akurasi, dengan memaksimumkan ketepatan klasifikasi. Setelah membentuk 100 kromosom, kemudian melakukan evaluasi terhadap masing-masing kromosom yang telah terbentuk menggunakan nilai *fitness*. Nilai *fitness* untuk setiap kromosom dapat dilihat pada Tabel 4.10.

Tabel 4.10 Ilustrasi Nilai Fitness Setiap Kromosom

Kromosom	Gen		*Fitness*
	C	γ	
1	6,2180	6,8250	0,9111
2	1,2931	1,5625	0,9033
3	5,6523	3,0676	0,9222
⋮	⋮	⋮	⋮
98	0,7724	2,6092	0,9556

99	2,3330	2,5599	0,9022
100	4,4564	0,4613	0,8911

Setelah mendapatkan nilai fitness untuk setiap kromosom, langkah selanjutnya adalah proses seleksi dengan melakukan pembentukan kromosom orang tua menggunakan metode *roulette wheel selection* (RWS). Metode ini menirukan permainan *roulette wheel* dimana masing-masing kromosom menempati lingkaran pada roda *roulette* secara proporsional sesuai dengan nilai *fitness*nya. Semakin besar *fitness* suatu kromosom, maka semakin besar pula peluang kromosom tersebut terpilih. Tahapan proses seleksi *roulette wheel* dapat dilihat pada Tabel 4.11.

Tabel 4.11 Ilustrasi Proses RWS pada Optimasi GA

Kromosom	*Fitness*	Proporsi Nilai *Fitness*	Kumulatif Proporsi Nilai *Fitness*	Bilangan *Random*
1	0,9111	0,0108	0,0108	0,0233
2	0,9033	0,0107	0,0216	0,9137
3	0,9222	0,0110	0,0325	0,4705
⋮	⋮	⋮	⋮	⋮
98	0,9556	0,0114	0,9787	0,4173
99	0,9022	0,0107	0,9894	0,0343
100	0,8911	0,0106	1	0,8937

Tabel 4.11 menunjukkan bahwa langkah awal pada proses seleksi *roulette wheel* adalah menghitung proporsi dari nilai fitness dengan cara membagi setiap nilai fitness kromosom dengan total nilai fitness. Kemudian menghitung nilai kumulatif dari proporsi nilai fitness tersebut dan membangkitkan bilangan random dengan distribusi *uniform* dengan batas 0 sampai dengan 1 sebanyak 100. Tahapan tersebut berhenti apabila telah diperoleh 100 kromosom calon orang tua berdasarkan bilangan *random* yang telah dibangkitkan. Bilangan random pertama pada Tabel 4.11 mempunyai nilai 0,0233, maka kromosom calon orang tua pertama didapatkan dari kumulatif nilai *fitness* yang labih besar dari 0,0233. Nilai kumulatif yang lebih besar dari 0,0233 yaitu pada kromosom ke-3, sehingga kromosom tersebut menjadi

kromosom pertama pada populasi baru. Setelah mendapatkan 100 kromosom orang tua, dilakukan proses pindah silang (*crossover*).

Proses pindah silang dilakukan apabila bilangan random yang dibangkitkan kurang dari peluang pindah silang (P_c). Nilai P_c yang digunakan sebesar 0,8. Kromosom yang mengalami pindah silang disebut dengan kromosom orang tua terpilih, karena proses pindah silang menghasilkan kromosom baru (anak) dari hasil perpaduan 2 kromosom orang tua . Proses pindah silang menggunakan metode *local arithmetic crossover*. Perhitungan *local arithmetic crossover* adalah sebagai berikut (Dumitrescu, 2000).

$$C_1 = \alpha P_1 + (1-\alpha) P_2$$
$$C_2 = \alpha P_2 + (1-\alpha) P_1$$

dengan C_1 adalah kromosom anak hasil pindah silang pada kromosom ke-1 dan C_2 adalah kromosom anak hasil pindah silang pada kromosom ke-2. P_1 adalah kromosom orang tua ke-1, sedangkan untuk P_2 adalah kromosom orang tua ke-2 dan untuk nilai α adalah nilai bangkitan dari bilangan random berdistribusi uniform dengan *range* 0 sampai dengan 1. Ilustrasi proses pindah silang telah digambarkan pada Gambar 4.4.

Sebelum Pindah Silang			Bil. *Random*
Orang Tua 1	6,2073	5,1530	0,5420
Orang Tua 2	7,8576	2,2260	0,1588
Sesudah Pindah Silang			
Anak 1	6,4527	4,7178	
Anak 2	7,6122	2,6612	

Gambar 4.4 Ilustrasi Proses Pindah Silang Pada Optimasi GA

Gambar 4.4 menjelaskan mengenai ilustrasi proses pindah silang kromosom orang tua 1 dan orang tua 2 yang menghasilkan anak 1 dan anak 2. Misalkan, nilai α yang didapat sebesar 0,8513, sehingga kromosom anak hasil pindah silang dapat diketahui nilainya dengan perhitungan sebagai berikut.

$C_1 = 0,8513 \text{ x } 6,2073 + (1 - 0,8513) \text{ x } 7,8576$

$= 5,2843 + 1,1684$

$= 6,4527$

$C_2 = 0,8513 \text{ x } 7,8576 + (1 - 0,8513) \text{ x } 6,2073$

$= 6,6892 + 0,9230$

$= 7,6122$

Setelah dilakukan proses pindah silang, selanjutnya adalah melakukan mutasi gen menggunakan mutasi *uniform* yaitu dengan memberikan kesempatan yang sama pada setiap gen untuk dilakukan proses mutasi. Tahapan awal proses mutasi adalah dengan membangkitkan dua bilangan random (sebanyak gen dalam kromosom). Bilangan random tersebut kemudian dibandingkan dengan peluang mutasi sebesar 0,01 pada penelitian ini. Apabila nilai bilangan random pada salah satu gen kurang dari peluang mutasi, maka proses mutasi akan dilakukan pada gen tersebut. Proses mutasi dilakukan dengan mengganti nilai estimasi parameter dengan bilangan random yang masih berada dalam *range* pada gen yang mengalami proses mutasi. Ilustrasi pada proses mutasi telah digambarkan pada Gambar 4.5.

Kromosom	C	γ
Bilangan Random	0,0012	0,6392
1	6,4527	4,7178
Anak	Mutasi	Tidak di Mutasi
1	0.6823	4,7178

Gambar 4.5 Ilustrasi Proses Mutasi Pada Optimasi GA

Gambar 4.5 menjelaskan terjadinya proses mutasi pada kromosom pertama. Gen yang mengalami mutasi adalah gen yang memiliki bilangan random kurang dari peluang mutasi (P_m) sebesar 0,01 yaitu pada parameter *cost*. Nilai parameter *cost* pada kromosom pertama yang telah dimutasi menjadi 0,6823. Proses selanjutnya adalah proses *elitisme* untuk mempertahankan kromosom terbaik dalam populasi. Kromosom yang

dipertahankan sejumlah 3 kromosom terbaik dari total kromosom dalam populasi.

4.4.1 Optimasi Parameter Tanpa Seleksi Variabel

Setelah melakukan klasifikasi dengan SVM dan mendapatkan nilai parameter yang optimal, selanjutnya adalah melakukan klasifikasi dengan menambahkan *genetic algorithm* (GA) pada metode SVM. Dalam menentukan nilai parameter optimal pada optimasi ini diperoleh dari konsep GA yang telah dijelaskan pada sub bab 4.4. Sama seperti sebelumnya, dengan optimasi GA data terlebih dahulu dibagi menjadi 10-*fold cross validation*. Untuk mencari nilai parameter yang optimal maka menggunakan *range* dari parameter optimal pada SVM. *Range* untuk parameter *cost* adalah -1 sampai dengan 3, sedangkan *range* untuk parameter γ adalah -15 sampai dengan -9. Hasil dari SVM-GA tanpa seleksi variabel telah ditabelkan pada Tabel 4.12.

Tabel 4.12 Hasil Percobaan SVM-GA Tanpa Seleksi Variabel

Fold	Parameter Opt		Akurasi
	Cost	Gamma	
1	4,68477	0,00031	0,9451
2	4,46083	0,00050	0,9341
3	5,15708	0,00013	0,9457
4	2,90884	0,00027	0,9239
5	4,04258	0,00037	0,9457
6	2,39528	0,00055	0,9239
7	2,91868	0,00019	0,9348
8	**5,34815**	**0,00016**	0,9565
9	3,36852	0,00032	0,9344
10	3,36852	0,00032	0,9457

Ket : Huruf yang bercetak tebal menandakan parameter yang paling optimum

Nilai fitness yang digunakan dalam optimasi parameter yaitu ketepatan klasifikasi dari nilai akurasi. Tabel 4.12 menunjukkan bahwa dari 10 percobaan yang telah dilakukan, nilai parameter yang optimal ada pada fold ke-8. Nilai akurasi fold ke-8 lebih tinggi jika dibandingkan dengan fold lainnya, dimana nilai C dan

γ yang optimal adalah 5,34815 dan 0,00016. Berdasarkan nilai parameter C dan γ yang optimal, fungsi *hyperplane* yang terbentuk untuk klasifikasi pada data kanker prostat menggunakan SVM-GA adalah

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \exp(-0,00016 \| x_i - x \|^2) + b$$

dengan *x* adalah *vector* observasi dari variabel independen dimana n adalah jumlah observasi. α_i merupakan *vector* dengan nilai $0 \le \alpha_i \le 5,34815$, sedangkan y_i adalah suatu tanda dan b adalah suatu konstanta yang disebut bias. Setelah didapatkan nilai parameter C dan γ yang optimal, maka akan dilakukan klasifikasi pada data testing untuk setiap fold dengan nilai parameter yang optimal. Nilai akurasi data testing dari klasifikasi SVM-GA ditampilkan pada Tabel 4.13.

Tabel 4.13 Nilai Akurasi SVM-GA Tanpa Seleksi (Testing)

Fold	Akurasi
1	0.91
2	0.91
3	0.9
4	0.9
5	0.9
6	1
7	1
8	0.7
9	1
10	0.9

Hasil klasifikasi dengan menggunakan SVM-GA untuk semua variabel mempunyai nilai rata-rata akurasi sebesar 0,91%. Nilai akurasi untuk setiap fold dengan SVM-GA yang terlihat pada Tabel 4.13 sama dengan klasifikasi SVM. Akan tetapi, dengan optimasi GA pada fold ke-2 nilai akurasinya lebih rendah 0,09% dibandingkan dengan SVM. Sehingga dengan data microarray kanker prostat untuk semua variabel apabila

menggunakan optimasi GA didapatkan hasil yaitu nilai akurasinya menjadi turun 0,0091%. Oleh karena itu dilakukan klasifikasi menggunakan variabel yang telah terseleksi dengan metode FCBF sejumlah 29 variabel.

4.4.2 Optimasi Parameter dengan Seleksi Variabel

Klasifikasi dengan membandingkan jumlah variabel yang digunakan bertujuan untuk meningkatkan nilai akurasi yang ada. Hal ini dikarenakan apabila terdapat banyak variabel yang kurang sesuai terhadap observasi, maka dapat memberikan performa yang kurang baik untuk klasifikasi. Hal ini didukung dari penelitian yang telah dilakukan oleh Hajiloo, Rabiee dan Anooshahpour pada tahun 2015 mengenai data microarray, dimana hasilnya menunjukkan bahwa dengan dilakukan seleksi variabel maka dapat meningkatkan nilai akurasi.

Pada sub bab 4.3.2 telah dilakukan klasifikasi SVM menggunakan 29 variabel. Hasil dari klasifikasi tersebut didapatkan *range* untuk menentukan parameter yang optimal. *Range* dari klasifikasi SVM dengan 29 variabel yang akan digunakan untuk optimasi GA. Tabel 4.14 telah menunjukkan bahwa dengan *range* yang terbaik pada SVM, mempunya nilai parameter optimal yang berbeda di setiap fold.

Tabel 4.14 Hasil Percobaan SVM-GA dengan Seleksi Variabel

Fold	Parameter Opt		Akurasi
	Cost	Gamma	
1	6.0223	0.0167	0.9890
2	4.7071	0.0156	0.9890
3	5.9045	0.0078	0.9891
4	5.3951	0.0096	0.9891
5	**4.6847**	**0.0163**	1
6	4.8137	0.0362	0.9783
7	4.4107	0.0227	0.9891
8	4.5228	0.0776	0.9674
9	4.0008	0.0207	0.9891
10	6.8705	0.0354	0.9891

Ket : Huruf yang bercetak tebal menandakan parameter yang paling optimum

Tabel 4.14 menunjukkan bahwa SVM-GA dengan 29 variabel dapat meningkatkan nilai akurasi dari klasifikasi. Hal ini dikarenakan rata-rata nilai akurasi yang didapatkan sebesar 0,98. Pertambahan nilai akurasi 7% ini membuktikan bahwa dengan selekssi variabel dapat menghasilkan ketepatan klasifikasi yang lebih baik. Selain itu, Tabel 4.14 juga memberikan informasi bahwa fold yang memiliki nilai akurasi tertinggi ada pada fold ke-5. Hasil klasifikasi menggunakan data training dengan parameter yang optimal pada fold ke-5 adalah sebesar 100%. Sehingga fungsi *hyperplane* yang tebentuk untuk klasifikasi adalah

$$f(x) = \sum_{i=1}^{n} \alpha_i y_i \exp(-0,0163 \| x_i - x \|^2) + b$$

Selanjutnya, untuk klasifikasi data testing digunakan nilai parameter dari fold ke-5 untuk semua fold. Hasil klasifikasi SVM-GA data testing menggunakan parameter yang optimal dengan 29 variabel telah ditampilkan pada Tabel 4.15.

Tabel 4.15 Nilai Akurasi SVM-GA dengan Seleksi (Testing)

Fold	Akurasi
1	0.91
2	1
3	1
4	1
5	0.9
6	1
7	1
8	0.8
9	1
10	0.9

Rata-rata akurasi yang didapatkan dari 10 *fold* dengan SVM-GA adalah sebesar 0,95%. Perhitungan tersebut dihitung berdasarkan Tabel 4.15 Nilai akurasi yang didapatkan untuk setiap *fold* juga lebih baik dibandingkan SVM-GA tanpa seleksi variabel.

4.5 Klasifikasi Menggunakan Naïve Bayes

Pada saat melakukan klasifikasi *naïve bayes* langkah yang dilakukan sama seperti pada saat klasifikadi SVM yaitu membagi data menjadi data training dan testing menggunakan 10-fold cross validation. Kemudian dilakukan klasifikasi *naïve bayes* untuk semua variabel maupun variabel yang telah terseleksi dengan *genetic algorithm* (GA). Langkah pertama yang dilakukan adalah menghitung nilai probabilitas prior dengan cara membagi jumlah observasi setiap kelas dengan keseluruhan observasi pada data. Data training yang digunakan untuk setiap *fold* berbeda, sehingga probabilitas priornya juga berbeda. Nilai probabilitas prior untuk setiap fold dapat dilihat pada Tabel 4.16.

Tabel 4.16 Nilai Probabilitas Prior Data Training

Fold	Prob. Prior	
	Normal	Normal
1	0.4945	0.5055
2	0.4945	0.5055
3	0.4891	0.5109
4	0.4891	0.5109
5	0.4891	0.5109
6	0.4891	0.5109
7	0.4891	0.5109
8	0.4891	0.5109
9	0.4891	0.5109
10	0.4891	0.5109

Sama seperti sebelumnya, pada metode *naïve bayes* juga dilakukan perbandingan nilai akurasi antara semua variabel dan variabel yang telah terseleksi sebanyak 29 variabel.

4.5.1 Naïve Bayes Tanpa Seleksi Variabel

Langkah selanjutnya adalah mencari peluang bersyarat pada setiap kelas untuk setiap variabel independen. Variabel yang digunakan bersifat kontinu, oleh karena itu untuk mendapatkan nilai peluang bersyarat dilakukan pendekatan distribusi normal. Sehingga diperlukan nilai rata-rata dan standar deviasi dari setiap

kelas variabel independen pada data training untuk menghitung semua peluang pada tiap kelas. Nilai rata-rata dan standar deviasi telah ditabelkan pada Tabel 4.17.

Tabel 4.17 Rata-rata dan Standar Deviasi Tiap Kelas Data *Training*

Variabel	Kelas			
	Normal		Normal	
	Rata-Rata	St. Deviasi	Rata-Rata	St. Deviasi
1	-0.5546	0.4893	-0.6889	0.3061
2	-0.5502	0.4228	-0.7770	0.2930
3	-0.5320	0.5811	-0.6393	0.4008
⋮	⋮	⋮	⋮	⋮
6031	-0.7901	0.2913	-0.8264	0.2256
6032	-0.5188	0.6904	-0.6029	0.5004
6033	-0.4510	0.5358	-0.4365	0.4482

Tabel 4.17 menunjukkan nilai rata-rata dan standar deviasi dari data *training* tiap kelas yang digunakan untuk menentukan peluang tiap kelas pada data *testing* menggunakan persamaan (2.20). Tabel 4.18 menunjukkan peluang parsial kedua kelas pada data testing untuk fold ke-1.

Tabel 4.18 Peluang Parsial pada Tiap Kategori Data *Testing* Pertama

Variabel	x_{1i}	$P(x_{1i} \mid y=0)$	$P(x_{1i} \mid y=1)$
1	-0.3195	0.7266	0.6294
2	-0.1063	0.5439	0.4025
3	-1.0857	0.4362	0.7242
⋮	⋮	⋮	⋮
6031	-0.6533	1.2271	0.1417
6032	1.1448	0.0317	0.6792

6033	-1.0857	0.3693	0.8605

Setelah menemukan peluang parsial tiap variabel pada setiap kelas, langkah selanjutnya adalah menghitung peluang posterior yang kemudian digunakan untuk menentukan klasifikasi kelas pada data *testing*.

$$P(X_1, X_2, \ldots, X_{6033} \mid Y = 0) = P(x_{1(1)} \mid Y = 0).P(x_{1(2)} \mid Y = 0) \ldots P(x_{1(6033)} \mid Y = 0).P(Y = 0)$$
$$= 0,7266 \text{ x } 0,5439 \text{ x } \ldots \text{ x } 0,3693 \text{ x } 0,4945$$
$$= 0,0715$$

$$P(X_1, X_2, \ldots, X_{6033} \mid Y = 1) = P(x_{1(1)} \mid Y = 1).P(x_{1(2)} \mid Y = 1) \ldots P(x_{1(6033)} \mid Y = 1).P(Y = 1)$$
$$= 0,6294 \text{ x } 0,4025 \text{ x } \ldots \text{ x } 0,8605 \text{ x } 0,5055$$
$$= 0,0547$$

Hasil perhitungan menunjukkan bahwa data testing pertama untuk fold ke-1 masuk dalam kelas normal, karena nilai peluang posterior pada kelas normal lebih tinggi jika dibandingkan dengan nilai *posterior* kelas tumor. Perhitungan nilai peluang posterior dilakukan untuk semua data testing di setiap fold. Hasil dari klasifikasi pada fold ke-1 dapat dilihat pada Tabel 4.19.

Tabel 4.19 Konfusi Matrix Klasifikasi Naïve Bayes Fold ke-1 Data Testing

Aktual	Prediksi	
	Normal	Tumor
Normal	4	1
Tumor	2	4

Tabel 4.19 menunjukkan bahwa fold ke-1 mempunyai jumlah data testing sebanyak 11 data. Orang yang tepat diklasifikasikan normal atau tidak terkena kanker prostat dan orang yang tepat diklasifikasikan terkena kanker prostat terdapat masing-masing 4 orang. Sedangkan seseorang yang terkena kanker tetapi diklasifikasikan tidak terkena kanker ada 2 orang dan seseorang yang tidak terkena kanker tetapi diklasifikasikan tumor ada 1 orang. Ketepatan klasifikasi pada fold ke-1 adalah 72,73% dengan nilai AUC 73,33%. Perhitungan ketepatan klasifikasi pada fold ke-1 dapat dilihat sebagai berikut.

$$\text{Akurasi} = \frac{8}{11} x100\% = 72,73\%$$

$$\text{AUC} = \left(\frac{1}{2} \text{x} \left(\frac{4}{4+1} + \frac{4}{2+4} \right) \right) \text{x}100\% = 73,33\%$$

Untuk melihat nilai ketepatan klasifikasi setiap fold dapat dilihat pada Tabel 4.20.

Tabel 4.20 Hasil Klasifikasi Naïve Bayes Tanpa Seleksi

Fold	Akurasi	AUC
1	0.73	0.73
2	0.45	0.42
3	0.6	0.6
4	0.5	0.5
5	0.5	0.5
6	0.6	0.6
7	0.6	0.6
8	0.7	0.7
9	0.7	0.7
10	0.8	0.8

Tabel 4.20 menunjukkan nilai ketepatan klasifikasi untuk setiap fold mempunyai hasil yang berbeda. Rata – rata nilai akurasi dengan klasifikasi naïve bayes adalah 61,82% dan nilai AUC sebesar 61,5%. Hasil dari nilai AUC dapat dikatakan bahwa klasifikasi dengan menggunakan metode naïve bayes masih kurang baik untuk data microarray kanker prostat.

4.5.2 Naïve Bayes Dengan Seleksi Variabel

Jumlah variabel yang digunakan setelah dilakukan seleksi variabel adalah 29 variabel. Hasil klasifikasi dengan variabel yang terseleksi telah ditabelkan pada Tabel 4.21.

Tabel 4.21 Hasil Klasifikasi Naïve Bayes Dengan Seleksi

Fold	Akurasi	AUC
1	0.91	0.92

2	1	1
3	1	1
4	1	1
5	0.9	0.9
6	1	1
7	1	1
8	0.8	0.8
9	1	1
10	0.9	0.9

Hasil klasifikasi dengan menggunakan variabel yang telah terseleksi mempuyai hasil klasifikasi yang lebih baik. Hal ini ditunjukkan pada Tabel 4.21 dari 10 fold terdapat 6 fold yang mempunyai hasil klasifikasi sangat baik dikarenakan nilai akurasinya sebesar 100%. Rata-rata nilai akurasi yang didapatkan dengan menggunakan 29 variabel adalah 95,09%. Sedangkan nilai AUC sebesar 95,17%, artinya dengan melakukan seleksi variabel pada penelitian ini dapat meningkatkan nilai akurasi.

4.6 Perbandingan Nilai Ketepatan Klasifikasi

Setelah melakukan klasifikasi pada data microarray kanker prostat dengan seluruh variabel ataupun dengan 29 variabel SVM, SVM-GA dan naïve bayes, langkah selanjutnya adalah membandingkan hasil dari klasifikasi beberapa metode tersebut dengan melihat nilai ketepatan klasifikasinya. Nilai akurasi yang telah didapatkan untuk setiap metode telah dirangkum Pada Tabel 4.22.

Tabel 4.22 Perbandingan Nilai Ketepatan Klasifikasi

Metode	Variabel	Data Testing	
		Akurasi	AUC
SVM	Tanpa Seleksi	92.09	93.96
	FCBF	95.09	96.07
SVM-GA	Tanpa Seleksi	91.18	93.24
	FCBF	95.09	96.07
Naïve Bayes	Tanpa Seleksi	61.82	61.50
	FCBF	95.09	95.17

Berdasarkan hasil analisis yang telah dilakukan, Tabel 4.23 menunjukkan bahwa nilai tertinggi didapatkan pada saat klasifikasi menggunakan SVM-GA dengan variabel yang telah diseleksi. Meskipun nilai ketepatan klasifikasi dengan data testing sama seperi pada saat menggunakan SVM *Grid Search* , namun apabila dilihat juga dari nilai ketepatan klasifikasi pada saat menggunakan data training maka SVM-GA yang mempunyai hasil klasifikasi lebih baik dibandingkan SVM *Grid Search.*

BAB V
KESIMPULAN DAN SARAN

5.1 Kesimpulan

Berdasarkan hasil analisis dan pembahasan pada bab sebelumnya, dapat disimpulkan sebagai berikut.

1. Nilai akurasi dengan menggunakan semua variabel untuk metode SVM adalah sebesar 92,09%, sedangkan untuk metode SVM-GA mendapatkan hasil nilai akurasi sebesar 91,18% dan dengan menggunakan metode naïve bayes hasil yang didapatkan sebesar 61,82%. Apabila dibandingkan dengan SVM dan SVM-GA, hasil klasifikasi naïve bayes dapat dikatakan kurang baik untuk mengklasifikasikan seseorang terkena kanker prostat ataupun tidak, hal ini dikarenakan nilai AUC yang didapat juga tidak jauh berbeda dengan nilai akurasi yaitu sebesar 61,50%.
2. Seleksi variabel menggunakan FCBF didapatkan hasil seleksi sejumlah 29 variabel. Klasifikasi menggunakan FCBF untuk metode SVM, SVM-GA dan naïve bayes mempunyai nilai akurasi yang sama yaitu sebesar 95,09%. Akan tetapi untuk nilai akurasi pada data training klasifikasi SVM-GA lebih baik dibandingkan SVM, hal ini dikarenakan terdapat kenaikan sebesar 2,72%. Sedangkan untuk metode naïve bayes nilai AUC yang didapatkan lebih rendah jika dibandingkan dengan SVM ataupun SVM-GA.
3. Metode SVM-GA lebih baik dibandingkan metode SVM dan naïve bayes. Adanya seleksi variabel dapat meningkatkan nilai akurasi untuk klasifikasi data microarray kanker prostat.

5.2 Saran

Berdasarkan penelitian yang telah dilakukan, saran yang diberikan oleh peneliti adalah menggunakan metode seleksi variabel lainnya untuk mendapatkan nilai akurasi yang lebih tinggi lagi dibandingkan dari hasil penelitian ini.

(Halaman ini sengaja dikosongkan)

DAFTAR PUSTAKA

Badan Pusat Statistika. (2015). Retrieved Maret 7, 2018, from Statistik Angka Harapan Hidup dari Tahun 1995-2015: https://www.bps.go.id/statictable/2014/09/22-/1517/angka-harapan-hidup-penduduk-beberapa-negara-tahun-1995-2015.html

Berson, A., & Smith, S. (2001). *Data Warehousing, Data Mining & OLAP*. New York: McGraw-Hill.

CancerHelps. (2014). *Bebas Kanker Itu Mudah.* Jakarta: FMedia.

Cho, S.-B., & Won, H.-H. (2003). *Macine* Learning in DNA Microarray Analysis For Cancer Classification. *Proceedings of The First Asia-Pacific Bioinformatics Conference on Bioinformatics* , Vol 19, 189-198.

Firdausanti, N. A. (2017). *Klasifikasi Kelas Risiko Pasien Pneumonia Menggunakan Metode Hybrid Analisis Diskriminan Linier-Particle Swarm Optimization (ADL-PSO) dan Naive bayes Classification.* Surabaya: Institut Teknologi Sepuluh Nopember.

Fu, L. (1994). *Neural Network in Computers Intelligence*. Singapura: McGraw-Hill.

Globocan-IARC. (2012). Retrieved Maret 7, 2018, from Estimated cancer incidence, mortality and prevalence worldwide in 2012. International Agency for Research on Cancer, World Health Organization: http://globocan.iarc.-fr/Pages/fact_shee-ts_cancer.aspx

Gorunescu, F. (2011). *Data Mining.* Berling: Springer.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The Elements of Statistical Learning.* New York: Springer.

Kementrian Kesehatan RI. (2014). Retrieved Maret 7, 2018, from Infodatin : Pusat data dan informasi kementrian kesehatan RI, situasi dan analisis lanjut usia: http://www.depkes.go-.id/resources/download/pusdatin/infodatin/infodatinlansia.pf

Kementrian Kesehatan RI. (2015). Retrieved Maret 7, 2018, from Profil kesehatan Indonesia tahun 2013: http://www-.depkes.go.id/resources/download/pusdatin/profilkesehatan-indonesia/profil-kesehatan-indonesia2014-.pdf

Kusumadewi, S., & Purnomo, H. (2005). *Penyelesaian Masalah Optimasi dengan Teknik-Teknik Heuristi*. Yogyakarta: Graha Ilmu.

Mubarok, M. S., Purbolaksono, M. D., & Adiwijaya. (2017). Implementasi Mutual Information dan Bayesian Network Untuk Klasifikasi Data Microarray. *e-Proceeding of Engineering* , Vol. 4, 3292.

Nuansa, E. P. (2017). *Analisis Sentimen Pengguna Twitter Terhadap Pemilihan Gubernur DKI Jakrata Dengan Metode Naive bayes Classification dan Support vector Macine.* Surabaya: Institut Teknologi Sepuluh Nopember.

Prasetyo, E. (2012). *Data Mining Konsep dan Aplikasi Menggunakan Matlab.* Yogyakarta: Andi.

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

Selvaraj, S., & Natarajan, J. (2011). Microarray Data Analysis and Mining Tools. *Bioinformation , 6(3)*, 95-99.

Singh, D., Febbo, P. G., Ross, K., Jackson, D. G., Manola, J., Ladd, C., et al. (2002). Gene Expression Correlates of Clinical Prostate Cancer Behavior. *Cancer Cell* , 203-209.

Sitohang, M. E. (2012). *Analisis Sinyal Electronic Nose Berbasis Wavelet Menggunakan Support vector Macine Untuk Identifikasi Jenis Teh Hitam.* Semarang: Universitas Diponegoro.

Sivanandam, S. N., & Deepa, S. N. (2008). *Introduction to Genetic Algorithm* . Berlin: Springer Verlag Heidelberg.

Vanitha, C. D., Devaraj, D., & Venkatesulu, M. (2015). Gene Expression Data Classification using *Support vector* and Mutual Information-Base Gene Selection. *Procedia Computer Science* , 47, 13-21.

Vapnik, V. N. (1999). *The Nature of Statistical Learning Theory 2nd Edition*. Berlin: Springererlag.

Vapnik, V., & Cortes, C. (1995). *Support vector* Networks. *Macine Learning* , 20, 273-297.

Wirasna, S. H., Adiwijaya, & Triantoro, D. (2017). Deteksi Kanker Berdasarkan Klasifikasi Microarray Data Menggunakan Principal Component Analysis dan Backpropagation Termodifikasi dengan Conjugate Gradient Powell-Beale. *e-Proceeding of Engineering* , Vol 4, 1247.

Yu, L., & Liu, H. (2003). Retrieved Maret 8, 2018, from Feature Selection for High Dimensional Data: A Fast Correlation-Based Filter Solution: www.hpl.hp.com-/conferences/icml-2003-/papers/144.pdf

(Halaman ini sengaja dikosongkan)

LAMPIRAN

Lampiran 1. Data Microarray Kanker Prostat

Obs	Y	X_1	…	X_{6033}
1	Normal	-0.9272	…	0.1940
2	Normal	-0.8359	…	0.0755
3	Normal	0.2361	…	-1.1544
4	Normal	-0.7486	…	-1.1215
5	Normal	0.1012	…	-1.1215
6	Normal	-0.3195	…	-1.0857
7	Normal	0.1393	…	-1.1778
8	Normal	-0.0743	…	-0.7027
9	Normal	0.6648	…	-1.1468
10	Normal	0.3130	…	-1.1547
11	Normal	-0.7807	…	-0.2405
⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮
⋮	⋮	⋮	⋮	⋮
6024	Tumor	-0.7361	…	-0.4140
6025	Tumor	-0.6900	…	0.2833
6026	Tumor	-0.8625	…	-0.6052
6027	Tumor	-0.8155	…	-0.1501
6028	Tumor	-0.7966	…	0.0081
6029	Tumor	-0.8124	…	-0.0010
6030	Tumor	-0.7988	…	-0.1452
6031	Tumor	-0.7830	…	0.0474
6032	Tumor	-0.8321	…	0.0359
6033	Tumor	-0.8630	…	-0.1462

Lampiran 2. *Syntax* K-*Fold Cross Validation*

```
library(MXM)
data=read.csv('E:/pros.csv', header=TRUE)
a=generatefolds(data$kelas, nfolds=10, stratified=TRUE, seed=12)
a
train=data[-a[[10]],]
test=data[a[[10]],]
write.csv(train,'E:/train10.csv')
write.csv(test,'E:/test10.csv')
```

Lampiran 3. *Syntax* Klasifikasi dengan *Naïve Bayes*

```
library(MXM)
library(e1071)
library(caret)
library(rminer)
data=read.csv('E:/pros.csv', header=TRUE)
a=generatefolds(data$kelas, nfolds=10, stratified=TRUE, seed=12)
AUC=rep(0,10)
for(i in 1:10)
{
  train=data[-a[[i]],]
  test=data[a[[i]],]
  model=naiveBayes(kelas~., data=train)
  pred=predict(model,test)
  tabel1=table(test$kelas,pred)

AUC[i]=0.5*((tabel1[1,1]/sum(tabel1[1,]))+(tabel1[2,2]/sum(tabel1[2,])))
}
pred
mean(AUC)
```

Lampiran 4. *Syntax* Tunning Parameter SVM

```
test=read.csv("E:/TA/test10.csv", header=TRUE)
train=read.csv("E:/TA/train10.csv", header=TRUE)
library(e1071)
ptm<-proc.time()
range_cost = 2^seq(-5,-1, by=1)
range_gamma = 2^seq(-15,-9, by=1)
hasil = matrix(0,1,4)
auc=function(model)
{
  pred=predict(model,test[,2:6034],type="kelas")
  Tabel=table(pred=pred, test[,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, 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
```

Lampiran 5. *Syntax* Optimasi SVM dengan *Genetic Algorithm*

```
ibrary(e1071)
library(GA)
test=read.csv("E:/TA/test10.csv", header=TRUE)
train=read.csv("E:/TA/train10.csv", header=TRUE)
xtest=test[,2:6034]
ytest=test[,1]
xtrain=train[,2:6034]
ytrain=train[,1]
ptm<-proc.time()
fitnessFunc=function(x)
{
  par_cost=x[1]
  par_gamma=x[2]
  model<-svm(ytrain~., data=train, cost=par_cost, gamma=par_gamma, cross=10,
scale=FALSE)
  return(model$tot.accuracy)
}
theta_min=c(p_cost=0.5, p_gamma=0)
theta_max=c(p_cost=8, p_gamma=0.001953125)
gaControl("real-valued"=list(selection="ga_rwSelection",
crossover="gareal_laCrossover", mutation="gareal_raMutation"))
fitnesvalue=c()
solutions=c()
result=ga(type="real-valued",
        fitness=fitnessFunc,
        names=names(theta_min),
        min=theta_min,
        max=theta_max,
        selection=gaControl("real-valued")$selection,
        crossover=gaControl("real-valued")$crossover,
        mutation=gaControl("real-valued")$mutation,
        popSize=100,
        maxiter=10,
        run=100,
        maxFitness=100,
        pcrossover=0.8,
        pmutation=0.01,
        monitor=plot)
summary(result)
```

Lampiran 6. Pembagian Data Menjadi 10 Fold

```
$`Fold 1`
 [1]  6  42  34  37  18  54  93 102  71  99  92
$`Fold 2`
 [1] 31 41 33 26 20 66 52 78 76 72 67
$`Fold 3`
 [1] 30 11  5 23 14 86 63 89 81 90
$`Fold 4`
 [1] 48 35  2 45 46 75 51 96 61 83
$`Fold 5`
 [1] 40 38 32  3 27 58 60 53 82 80
$`Fold 6`
 [1] 29 47 21 12 49 74 77 65 79 85
$`Fold 7`
 [1]  1 43 13 16  7 73 62 88 57 59
$`Fold 8`
 [1] 10  9 39  4 36 84 68 95 55 97
$`Fold 9`
 [1] 28 50 19 15 25 98 91 94 70 56
$`Fold 10`
 [1]  22   8  44  17  24  87  69 101 100  64
```

Lampiran 7. Pecarian Parameter Optimal SVM Grid Search

```
> hasil
[1] 5.000000e-01 4.882812e-04 8.181818e+01 8.588889e-01
> tune_par$best.parameters$cost
[1] 0.5
> tune_par$best.parameters$gamma
[1] 0.0004882812
> a=auc(model)
> a
[1] 81.81818
> 1- tune_par$best.performance
[1] 0.8588889
> tune_par$best.parameters
   cost        gamma
25  0.5 0.0004882812
```

Lampiran 6. Pecarian Parameter Optimal SVM Grid Search (Lanjutan)

```
> tune_par$performance
      cost        gamma     error dispersion
1  0.03125 3.051758e-05 0.5588889 0.16843099
2  0.06250 3.051758e-05 0.5588889 0.16843099
3  0.12500 3.051758e-05 0.5588889 0.16843099
4  0.25000 3.051758e-05 0.5588889 0.16843099
5  0.50000 3.051758e-05 0.4477778 0.15396453
6  0.03125 6.103516e-05 0.5588889 0.16843099
7  0.06250 6.103516e-05 0.5588889 0.16843099
8  0.12500 6.103516e-05 0.5588889 0.16843099
9  0.25000 6.103516e-05 0.4477778 0.17870514
10 0.50000 6.103516e-05 0.3855556 0.12217171
11 0.03125 1.220703e-04 0.5588889 0.16843099
12 0.06250 1.220703e-04 0.5588889 0.16843099
13 0.12500 1.220703e-04 0.5588889 0.16843099
14 0.25000 1.220703e-04 0.4477778 0.15396453
15 0.50000 1.220703e-04 0.2877778 0.15148556
16 0.03125 2.441406e-04 0.5477778 0.18754423
17 0.06250 2.441406e-04 0.5477778 0.18754423
18 0.12500 2.441406e-04 0.5255556 0.19231105
19 0.25000 2.441406e-04 0.3833333 0.11249143
20 0.50000 2.441406e-04 0.1755556 0.05587562
21 0.03125 4.882812e-04 0.5255556 0.20608374
22 0.06250 4.882812e-04 0.5255556 0.20608374
23 0.12500 4.882812e-04 0.5255556 0.20608374
24 0.25000 4.882812e-04 0.2388889 0.15281986
25 0.50000 4.882812e-04 0.1411111 0.09911128
26 0.03125 9.765625e-04 0.5144444 0.20727840
27 0.06250 9.765625e-04 0.5144444 0.20727840
28 0.12500 9.765625e-04 0.5144444 0.20727840
29 0.25000 9.765625e-04 0.3600000 0.21968927
30 0.50000 9.765625e-04 0.1622222 0.12083387
31 0.03125 1.953125e-03 0.5255556 0.18504068
32 0.06250 1.953125e-03 0.5255556 0.18504068
33 0.12500 1.953125e-03 0.5255556 0.18504068
34 0.25000 1.953125e-03 0.5255556 0.18504068
35 0.50000 1.953125e-03 0.3277778 0.15811388
```

Lampiran 8. Model SVM dengan Seleksi Variabel

=== Classifier model (full training set) ===
SMO
Kernel used:
RBF Kernel: K(x,y) = exp(-0.00195*(x-y)^2)
Classifier for classes: normal, tumor
Number of support vectors: 90
b = 0,5042

No	y	Alpha	1	2	3	27	28	29
1	+	2	0.51	0.42	0.11	0.77	0.23	0.88
2	-	2	0.29	0.13	0.46	0.15	0.44	0.10
3	-	2	0.33	0.08	0.66	0.10	0.09	0.05
4	-	2	0.36	0.54	1.00	0.15	0.43	0.10
5	-	2	0.43	0.08	0.10	0.25	0.10	0.06
6	+	2	0.46	0.70	0.28	0.08	0.21	0.04
7	+	2	0.25	0.61	0.52	0.76	0.26	0.41
8	+	2	0.60	0.48	0.00	0.71	0.15	0.42
9	-	2	0.49	0.04	0.64	0.05	0.32	0.25
10	+	2	0.55	0.07	0.69	0.09	0.24	0.33
80	+	2	0.68	0.67	0.05	0.19	0.19	0.89
81	+	2	0.49	0.04	0.29	0.05	0.05	0.34
82	+	2	0.49	0.75	0.29	0.36	0.30	0.15
83	-	2	0.43	0.07	0.78	0.09	0.39	0.18
84	-	2	0.43	0.07	0.50	0.08	0.26	0.13
85	-	2	0.25	0.18	0.40	0.22	0.50	0.16
86	+	2	0.51	0.95	0.32	0.27	0.19	0.17
87	-	2	0.44	0.05	0.44	0.07	0.51	0.16
88	-	2	0.24	0.50	0.58	0.17	0.16	0.11
89	-	2	0.11	0.59	0.55	0.17	0.17	0.12
90	-	2	0.49	0.04	0.64	0.05	0.05	0.11

Lampiran 9. Model SVM-GA dengan Seleksi Variabel

=== Classifier model (full training set) ===
SMO
Kernel used:
RBF Kernel: K(x,y) = exp(-0.0163*(x-y)^2)
Classifier for classes: normal, tumor
Number of support vectors: 46
b = - 0,6364

No	y	alpha	1	2	3	27	28	29
1	+	4.68	0.40	0.19	0.22	0.23	0.23	0.50
2	-	0.0016	0.37	0.12	0.83	0.14	1.00	0.09
3	+	4.68	0.41	0.75	0.46	0.15	0.15	0.09
4	+	4.68	0.54	0.04	0.58	0.27	0.05	0.24
5	-	4.68	0.25	0.18	0.52	0.22	0.50	0.15
6	-	4.68	0.41	0.08	0.66	0.10	0.10	0.05
7	+	4.68	0.46	0.70	0.40	0.08	0.21	0.03
8	+	4.68	0.50	0.08	0.35	0.10	0.10	0.27
9	-	3.89	0.17	0.54	0.47	0.20	0.20	0.14
10	+	4.68	0.49	0.04	0.41	0.05	0.05	0.32
36	+	4.68	0.51	0.95	0.44	0.27	0.19	0.17
37	-	4.68	0.16	0.14	0.78	0.17	0.58	0.11
38	-	4.68	0.44	0.05	0.57	0.07	0.51	0.16
39	+	4.68	0.60	0.08	0.84	0.20	0.10	0.32
40	+	2.00	0.49	0.75	0.41	0.36	0.30	0.14
41	-	4.68	0.43	0.07	0.91	0.09	0.39	0.17
42	-	4.68	0.42	0.05	0.54	0.06	0.40	0.15
43	-	4.68	0.44	0.68	0.67	0.05	0.23	0.29
44	-	4.68	0.24	0.50	0.71	0.17	0.16	0.11
45	-	4.68	0.11	0.59	0.67	0.17	0.17	0.11
46	-	4.68	0.49	0.04	0.77	0.05	0.05	0.11

Lampiran 10. Surat Pernyataan Data

SURAT PERNYATAAN

Saya yang bertanda tangan di bawah ini, mahasiswa Program Sarjana Departemen Statistika FMKSD ITS.

Nama : Violita Pertiwi
NRP : 062116 4500 0020

menyatakan bahwa data yang digunakan dalam Tugas Akhir ini, merupakan data sekunder yang diambil dari publikasi yaitu:

Sumber	: *Journal* "*Gene Expression Correlates Of Clinical Prostate Cancer Behavior*" Pada Maret 2002 oleh Dinesh Singh dkk di Florida
Keterangan	: Data *Microarray* "*Prostate Cancer*"

Surat pernyataan ini dibuat dengan sebenarnya. Apabila terdapat pemalsuan data maka mahasiswa bersedia menerima sanksi sesuai aturan yang berlaku.

Mengetahui,

Pembimbing Tugas Akhir	Surabaya, 31 Juli 2018
Irhamah, M.Si, Ph.D	.Violita Pertiwi
NIP. 19780406 200112 2 002	NRP. 062116 4500 0020

(Halaman ini sengaja dikosongkan)

BIODATA PENULIS

Penulis memiliki nama lengkap Violita Pertiwi dan dilahirkan di Surabaya, 02 November 1995 dari pasangan Drs. Yahya Fajar dan Listari Puspa Dewi. Penulis merupakan anak kedua dari dua bersaudara, dengan kakak perempuan yang bernama Virga Fatari. Penulis bertempat tinggal di Jalan Wonorejo Asri 1/27 Kec. Rungkut, Kel. Wonorejo. Penulis telah menempuh pendidikan formal mulai dari TK Putra Berlian, SDN Penjaringan Sari II, SMPN 23 Surabaya, dan SMAN 14 Surabaya. Setelah lulus dari SMA, penulis melanjutkan studinya di Diploma III Departemen Statistika ITS pada tahun 2013 dan melanjutkan ke Jenjang Sarjana di Departemen Statistika ITS melalui jalur regular pada tahun 2016. Selama perkuliahan penulis aktif mengikuti kegiatan kepanitiaan di KM ITS. Penulis pernah bergabung dalam organisasi kemahasiswaan, yakni sebagai anggota DPM FMIPA ITS periode 2014/2015. Selain aktif dalam organisasi, selama masa perkuliahan penulis memiliki pengalaman kerja sebagai surveyor di PT Mitra Pinasthika Mulia (MPM) dan Astra Honda Motor (AHM). Selain itu, penulis juga memiliki pengalaman kerja sebagai Asisten Dosen mata kuliah Metode Multivariat Terapan. Penulis juga mendapatkan kesempatan Kerja Praktek di Kantor Wilayah Direktorat Jenderal Pajak Jawa Timur I dan Kantor Bank Indonesia Kota Surabaya. Komunikasi lebih lanjut dengan penulis dapat melalui email violitapertiwi@gmail.com.