IMPLEMENTASI METODE *SUPPORT VECTOR MACHINE* DAN *NAÏVE BAYES* PADA DATA EKSPRESI GEN *MICROARRAY*

(Studi Kasus: Klasifikasi Data Ekspresi Gen *Induced Sputum* Pada Pasien Penderita Penyakit Asma GSE76262)

TUGAS AKHIR

Diajukan Sebagai Salah Satu Syarat Untuk Memperoleh Gelar Sarjana Program Studi Statistika

Disusun Oleh:

Muhammad Faskul Fatan

19611111

PROGRAM STUDI STATISTIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS ISLAM INDONESIA

YOGYAKARTA

2023

HALAMAN PERSETUJUAN PEMBIMBING

TUGAS AKHIR

.

Judul	:	Implementasi Metode *Support Vector Machine* dan *Naïve Bayes* pada Data Ekspresi Gen *Microarray* (Studi Kasus: Klasifikasi Data Ekspresi Gen *Induced Sputum* Pada Pasien Penderita Penyakit Asma GSE76262)
Nama Mahasiswa	:	Muhammad Faskul Fatan
NIM	:	19611111

TUGAS AKHIR INI TELAH DIPERIKSA DAN DISETUJUI UNTUK DIUJIKAN

Yogyakarta, 4 April 2023

Kepala Program Studi	Pembimbing
(Dr. Atina Ahdika, S.Si., M.Si)	**(Dr. Techn. Rohmatul Fajriyah, S.Si., M.Si.)**

HALAMAN PENGESAHAN

TUGAS AKHIR

IMPLEMENTASI METODE SVM DAN *NAÏVE BAYES* PADA DATA EKSPRESI GEN *MICROARRAY*

(Studi Kasus: Klasifikasi Data Ekspresi Gen *Induced Sputum* Pada Pasien Penderita Penyakit Asma GSE76262)

Nama Mahasiswa : Muhammad Faskul Fatan

NIM : 19611111

TUGAS AKHIR INI TELAH DIUJIKAN
PADA TANGGAL: (11 April 2023)

Nama Penguji:	**Tanda Tangan**
1. Dina Tri Utari, S.Si., M.Sc.	.....
2. Muhammad Hasan Sidiq Kurniawan., M.Sc.,	.....
3. Dr. Techn. Rohmatul Fajriyah, S.Si., M.Si.	.....

Mengetahui,
Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam

UNIVERSITAS ISLAM INDONESIA YOGYAKARTA FAKULTAS MIPA

Prof. Riyanto, S.Pd., M.Si., Ph.D.

KATA PENGANTAR

بِسْمِ اللَّهِ الرَّحْمَٰنِ الرَّحِيمِ

Assalamu'alaikum Wr.Wb.

Alhamdulillah, puji dan syukur atas kehadirat Allah SWT yang telah melimpahkan rahmat, nikmat dan karunia-Nya, sehingga penulis mampu dapat menyelesaikan tugas akhir yang berjudul "Implementasi Metode *Support Vector Machine* dan *Naïve Bayes* pada Data Ekspresi Gen *Microarray* dengan studi kasus: Klasifikasi Data Ekspresi Gen Induced Sputum Pada Pasien Penderita Penyakit Asma GSE76262)" dengan baik. Shalawat serta salam senantiasa tercurahkan kepada Nabi Muhammad SAW beserta keluarga, sahabat dan umatnya hingga akhir zaman. Semoga syafaatnya sampai kepada kita semua umatnya di hari akhir kelak. Dalam pelaksanaan penelitian dan penyusunana laporan Tugas Akhir, penulis dengan segala kekurangan mendapatkan wawasan, bimbingan, masukan, arahan, motivasi dan kesempatan untuk dapat menyelesaikan laporan ini. Oleh karena itu, penulis ingin mengucapkan terimakasih sebesar-besarnya kepada:

1. Kedua Orang tua tercinta bapak Muhammad Iksan, dan ibu Nurhayati yang selalu memberikan doa, dukungan, serta kasih sayang yang menjadi semangat untuk menyelesaikan skripsi ini.
2. Bapak Prof. Riyanto, S.Pd., M.Si., Ph.D., selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Islam Indonesia
3. Bapak Dr. Edy Widodo, S.Si., M.Si., selaku Ketua Program Studi Statistika Universitas Islam Indonesia beserta seluruh jajarannya.
4. Dr. Techn. Rohmatul Fajriyah, S.Si., M.Si., selaku dosen pembimbing yang telah mengajari, menjelaskan, dan selalu menyempatkan waktu dan tenaganya dalam memberikan bimbingan dan arahan kepada penulis selama penyusunan tugas akhir ini.
5. Alfan dan Lintang selaku teman satu bimbingan, yang selalu bersama-sama membantu dan menemani selama proses penyusunan tugas akhir ini.

6. Teman-teman dan rekan seperjuangan saya di kampus Universitas Islam Indonesia yang senantiasa berbagi suka cita, serta menemani masa perkuliahan hingga penulis dapat menyelesaikan tugas akhir ini.
7. Semua pihak yang telah membantu yang tidak dapat penulis sebutkan satu per satu, terima kasih.

Semoga Allah SWT membalas segala kebaikan kalian semua dengan segala anugrah, rahmat, dan Hidayah-Nya.

Penulis menyadari sepenuhnya bahwa Tugas Akhir ini masih jauh dari kata sempurna, oleh karena itu segala kritik dan saran yang sifatnya membangun selalu penulis harapkan. Semoga Tugas Akhir ini dapat bermanfaat bagi penulis khususnya dan bagi semua yang membutuhkannya. Akhir kata, semoga Allah SWT selalu melimpahkan rahmat serta hidayah-Nya kepada kita semua, Amin amin ya robbal alamiin.

Wassalamualaikum Wr.Wb.

Yogyakarta, (4 Aprill 2023)

(Muhammad Faskul Fatan)

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

DAFTAR LAMPIRAN

PERNYATAAN

Dengan ini saya menyatakan bahwa dalam Tugas Akhir ini tidak terdapat karya yang sebelumnya pernah diajukan untuk memperoleh gelar kesarjanaan di suatu Perguruan Tinggi dan sepanjang pengetahuan saya juga tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan orang lain, kecuali yang diacu dalam naskah ini dan disebutkan dalam daftar pustaka.

Yogyakarta, 04 April 2023

(Muhammad Faskul Fatan)

INTISARI

IMPLEMENTASI METODE *SUPPORT VECTOR MACHINE* DAN *NAÏVE BAYES* PADA DATA EKSPRESI GEN *MICROARRAY*

(Studi Kasus: Klasifikasi Data Ekspresi Gen *Induced Sputum* Pada Pasien Penderita Penyakit Asma GSE76262)

Muhammad Faskul Fatan

Program Studi Statistika, Fakultas Matematika dan Ilmu Pengetahuan Alam

Universitas Islam Indonesia

Asma adalah penyakit yang terjadi karena adanya penyempitan saluran udara karena infeksi atau peradangan. dikematian tahunan di seluruh dunia akibat asma diperkirakan mencapai 250.000 jiwa, karena hal tersebut perlu dilakukan tindakan pemeriksaan untuk pasien penderita penyakit asma dengan mengambil cairan lendir pekat atau dahak pada penderita asma dengan menggunakan teknik yaitu *induced sputum* atau penginduksian dahak. Teknik induksi sputum bertujuan untuk mendapatkan sputum yang memadai dari saluran napas bawah. Pada penelitian ini penginduksian sputum dilakukan pada pasien penderita asma kategori berat (*severe*), sedang (*moderate*), dan *healthy control*. Digunakan cabang ilmu bioinformatika dalam menganalisis permasalahan ini. Pada penelitian ini dilakukan analisis klasifikasi pada data microarray hasil dari ekspresi gen pada pasien penderita penyakit asma yang dilakukan penginduksian sputum dengan kode series GSE76262. Digunakan metode SVM dan *Naïve Bayes* untuk melakukan klasifikasi pada dataset yang digunakan. Diperoleh nilai akurasi untuk masing-masing metode tersebut, dimana dengan metode SVM diperoleh nilai akurasi 78.57%, dan untuk metode *naïve bayes* didaptkan nilai akurasi sebesar 71.43%. Dilakukan penanganan *imbalanced* data dengan SMOTE, dan dianalisis kembali menggunakan metode yang sama, sehingga diperoleh hasil klasifikasi berupa nilai akurasi untuk SVM dengan SMOTE sebesar 98.25, dan untuk metode *naïve bayes* dengan SMOTE sebesar 68.42%. Diketahui bahwa metode SVM dengan SMOTE merupakan metode yang paling baik dalam mengklasifikasikan data ekspresi gen pada pasien asma. Selain itu juga diperoleh nilai AUC dari metode terbaik yaitu SVM dengan SMOTE, diperoleh nilai AUC sebesar 99.13%, nilai tersebut termasuk dalam kategori *excellent classifier* atau metode tersebut merupakan metode yang sangat baik dan akurat dalam melakukan klasifikasi

Kata Kunci: Asma, Induksi Sputum, Klasifikasi, SVM, *Naïve Bayes*.

ABSTRACT

IMPLEMENTATION OF SUPPORT VECTOR MACHINE AND NAÏVE BAYES METHODS ON MICROARRAY GENE EXPRESSION DATA

(Case Study: Classification of Induced Sputum Gene Expression Data in Patients with Asthma GSE76262)

Muhammad Faskul Fatan

Department of Statistics, Faculty of Mathematics and Natural Sciences

Islamic University of Indonesia

Asthma is a disease that occurs due to narrowing of the airways due to infection or inflammation. Annual deaths worldwide due to asthma are estimated at 250,000 people, because of this it is necessary to carry out examination measures for patients with asthma by taking concentrated mucus or sputum in asthmatics using a technique that is induced sputum or sputum induction. The sputum induction technique aims to obtain adequate sputum from the lower airway. In this study, sputum induction was carried out on patients with asthma in the severe, moderate, and healthy control categories. The branch of bioinformatics is used to analyze this problem. In this study, classification analysis was carried out on microarray data resulting from gene expression in patients with asthma who were sputum induced with the series code GSE76262. SVM and Naïve Bayes methods were used to classify the dataset used. The accuracy value for each method was obtained, where the SVM method obtained an accuracy value of 78.57%, and for the naïve bayes method an accuracy value of 71.43% was obtained. Handling imbalanced data with SMOTE, and re-analyzed using the same method, so that the classification results are obtained in the form of an accuracy value for SVM with SMOTE of 98.25, and for the naïve bayes method with SMOTE of 68.42%. It is known that the SVM method with SMOTE is the best method in classifying gene expression data in asthma patients. In addition, the AUC value of the best method, namely SVM with SMOTE, obtained an AUC value of 99.13%, this value is included in the excellent classifier category or the method is a very good and accurate method in classifying asthma patients.

Keywords**:** *Ashtma, Induced Sputum, Classification, SVM, Naïve Bayes.*

BAB I
PENDAHULUAN

1.1 Latar Belakang Masalah

Asma adalah suatu kondisi yang disebabkan oleh penyempitan saluran udara karena infeksi atau peradangan. Asma mengandung banyak sel inflamasi seperti *eosinofil, sel mast, leukotrien* dan lain-lain. Ini terkait dengan peradangan kronis Hipersensitivitas saluran napas menyebabkan sering mengi atau *wheezing*, sesak napas, dada terasa berat dan batuk terutama pada malam hari dan dini hari (Novita & Zabit, 2014). Berdasarkan data yang diambil dari laporan *Global Initiatif for Asthma* (GINA) pada tahun 2012, diperkirakan terdapat 300 juta orang manusia yang mengidap penyakit asma diseluruh dunia saat ini, dimana kematian tahunan di seluruh dunia akibat asma diperkirakan mencapai 250.000 jiwa dan kematian tampaknya tidak berkorelasi baik dengan prevalensi (GINA, 2012).

Induced sputum atau *sputum* yang telah diinduksi merupakan suatu metode yang dirancang untuk mengumpulkan sampel yang memadai dari saluran napas individu yang tidak dapat mengeluarkan dahak secara spontan (Surjanto & Niwan, 2009). Metode pemeriksaan sputum ini berguna untuk mengevaluasi peradangan saluran napas pada pasien asma dan berbagai permasalahan pernapasan lainnya. eosinofil dahak atau *sputum* yang diinduksi dan oksida nitrat yang dihembuskan merupakan kandidat yang paling banyak diteliti untuk digunakan di arena klinis, akan tetapi terdapat ruang untuk banyak perbaikan dalam penerapannya dan *biomarker* lain mungkin dapat lebih bermanfaat atau bahkan lebih baik. Penerapan sputum yang diinduksi secara luas pada penyakit asma, dan melintasi spektrum keparahan penyakit telah memberikan wawasan tentang hubungan antara fungsi saluran napas dan peradangan saluran pernapasan, mengusulkan fenotipe penyakit baru dan menentukan fenotipe mana yang merespons terapi saat ini, dan mungkin yang paling penting disediakan alat tambahan untuk memandu manajemen klinis pasien asma.

Pada era kemajuan teknologi seperti sekarang, penerapan teknologi dalam dunia biomedis sendiri sudah berkembang pesat, salah satu cabang ilmu yang memadukan disiplin ilmu antara biologi molekul, matematika, dan teknik informasi adalah bioinformatika. Bionformatika merupakan disiplin yang relatif muda yang berkaitan dengan penyimpanan, pengambilan, dan analisis data biologis dengan alat informatika (Tramontano, 2018). Bioinformatika merupakan cabang dari ilmu biologi yang menganalisis informasi yang terkandung di dalam makromolekul biologis secara kuantitatif menggunakan bantuan komputer (Victor & Langkah, 2017). Pada penerapannya bioinformatika dapat diaplikasikan dalam berbagai bidang, seperti bidang klinis, dimana pada bidang klinis ini, bioinformatika dapat diaplikasikan untuk memanajemen data-data klinis dari pasien melalui *Electrical Medical Record* (EMR) yang dikembangkan oleh Clement J. McDonald, lalu penerapan dalam bidang lainnya yaitu, bioinformatika dapat digunakan untuk mengidentifikasi *agent* penyakit baru, diagnosis penyakit baru, dan penemuan obat-obatan (Aprijani & Elfaizi, 2004).

Bioinformatika merupakan bagian yang terintegrasi dari teknologi sekuensing generasi berikutnya atau yang dikenal dengan *Next-Generating Sequencing* (NGS). Analisis sekuens genom menggunakan pendekatan NGS adalah proses mengubah materi yang tidak berarti secara biologis (sampel DNA atau RNA) menjadi informasi yang dikodekan (kode biner dan kode basa nukleotida) dan mengubahnya menjadi informasi biologis yang berarti (IRRIB, 2017). Bioinformatika memegang peranan yang sangat penting dalam setiap langkah analisis NGS. Salah satu pengaplikasian dari teknologi pengurutan sel atau *cell squencing* adalah DNA *microarray*. DNA *microarray* adalah teknologi yang mampu menghasilkan informasi genetik dengan menggunakan *probe* oligonukleotida pada *array* dengan densitas tinggi, sehingga dapat digunakan sebagai salah satu metode untuk mendeteksi dan mengklasifikasikan suatu jaringan penyakit pada manusia.

Penggunaan teknologi *microarray* dalam penerapan bioinformatika dapat digunakan dengan melakukan teknik komputasi seperti metode *machine learning*. *Machine learning* dapat digunakan untuk menganalisis pemilihan gen atau protein

yang memiliki sifat terkait dan untuk mengklasifikasikan jenis sampel ekspresi gen dalam data *microarray* (Yang dkk, 2010). *Machine learning* dapat menjadi sebuah metode dalam penerapan teknik komputasi dikarenakan pada data biologis dibutuhkan data yang sangat besar dan membutuhkan kompleksitas tinggi, oleh karena itu algoritma dari *machine learning* dapat dimanfaatkan sebagai bentuk pelatihan dan pengenalan fitur utama dan untuk klasifikasi kelompok pada data biologis. Salah satu metode *machine learning* yang saat ini sedang berkembang dan dapat digunakan dalam pengolahan data dan analisis klasifikasi pada bidang bioinformatika adalah *Support Vector Machine* (SVM) dan *Naïve Bayes Classifier* (NBC).

Support Vector Machine atau SVM adalah metode *machine learning* yang bekerja berdasarkan prinsip *Structural Risk Minimization* (SRM) dengan tujuan menemukan *hyperplane* terbaik dari banyaknya jumlah kelas yang digunakan (Nugroho dkk, 2003). SVM merupakan metode yang ampuh sebagai solusi dalam mengatasi permasalahan pada analisis klasifikasi dan regresi. Dibandingkan dengan metode *machine learning* lainnya, SVM sangat kuat dalam mengenali pola halus dalam kumpulan data yang kompleks (Aruna S., 2011). SVM memiliki banyak fitur yang membuatnya menarik untuk analisis ekspresi gen, termasuk fleksibilitasnya dalam memilih fungsi kemiripan, keragaman solusi ketika berhadapan dengan kumpulan data yang besar, kemampuan untuk menangani ruang fitur yang besar, dan kemampuan untuk mengidentifikasi pencilan. karena fitur-fitur di atas SVM diadopsi dalam penelitian ini untuk klasifikasi data gen (Devi & Venkatesulu, 2015).

Selain dari *support vector machine*, salah satu metode *machine learning* lain yang dapat digunakan dalam bioinformatika adalah *naïve bayes*. *Naïve Bayes Classifier* adalah pengklasifikasi jaringan Bayesian sederhana yang dibangun di atas asumsi kuat bahwa atribut yang berbeda adalah independen satu sama lain berdasarkan kelasnya (Friedman, 1997). Terlepas dari kesederhanaannya, pengklasifikasi *Naïve Bayes* terbukti sangat efektif dibandingkan dengan pengklasifikasi lain yang lebih canggih (Hall, 2007). Oleh karena itu, tidak mengherankan jika pengklasifikasi *Naïve Bayes* menjadi populer dalam

menyelesaikan berbagai masalah klasifikasi termasuk analisis data *microarray* (Rickard Sandberg, 2001).

Penerapan *machine learning* pada cabang ilmu bioinformatika merupakan salah satu hal yang dapat dilakukan apabila ingin melakukan analisis data dan pengklasifikasian, sebagaimana yang sudah dijelaskan sebelumnya, penerapan klasifikasi dengan menggunakan metode *support vector machine* (SVM) dan *naïve bayes* merupakan metode yang cocok dan dapat digunakan dalam mengklasifikasikan data *microarray*, dimana SVM dalam analisis ekspresi gen, memiliki sejumlah fitur menarik yang meliputi pilihan fungsi kesamaan yang mempunyai fleksibilitas tinggi, penyebaran solusi yang baik saat menghadapi data yang besar, kemampuan mengatasi dimensi ruang fitur yang besar, serta kemampuan mengidentifikasi outlier. Sedangkan untuk *naïve bayes* merupakan metode yang memiliki efisiensi yang baik secara komputasi dan membutuhkan jumlah data pelatihan yang relatif kecil untuk memperkirakan parameter. Hal ini bermanfaat untuk data *microarray*, yang seringkali berisi sejumlah besar gen dan sampel yang terbatas, serta *naïve bayes* dapat secara baik menangani data *microarray* biasanya memiliki ruang fitur dimensi tinggi, di mana jumlah gen jauh melebihi jumlah sampel. Naive Bayes dapat menangani data dimensi tinggi dengan baik dan tidak terlalu rentan terhadap *overfitting* dibandingkan dengan metode klasifikasi lainnya. Oleh karena fitur-fitur tersebut. SVM dan *Naïve Bayes* diadopsi dalam penelitian ini untuk melakukan klasifikasi untuk data ekspresi gen.

Sehingga berdasarkan uraian yang telah disampaikan sebelumnya, maka akan dibuat suatu penelitian dengan judul "Implementasi Metode SVM, dan *Naive Bayes* pada Data Ekspresi Gen" dengan studi kasus: "Klasifikasi Data Ekspresi Gen *Induced Sputum* pada Pasien Penderita Penyakit Asma (GSE76262)". Dari penelitian ini diharapkan suatu hasil yang dapat mengetahui metode manakah yang paling baik dari SVM dan NBC untuk mengklasifikasikan kelas sputum atau dahak yang telah diinduksi pada pasien dengan gangguan pernafasan atau asma, dengan kelas "Sputum Severe (Berat), "Sputum Moderate (Sedang)", "Sputum Healthy Control" dengan melihat hasil dari beberapa metrik klasifikasi, seperti *accuracy, precision, recall, spesificity, f-score,* dan nilai AUC.

1.2 Rumusan Masalah

1. Bagaimana gambaran data ekspresi gen *induced sputum* pada pasien dengan gangguan pernafasan atau penyakit asma pada data *microarray*?
2. Bagaimana hasil klasifikasi pada data ekspresi gen *induced sputum* pada pasien penderita penyakit asma dengan menggunakan metode *Support Vector Machine* dan *Naive Bayes*?
3. Bagaimana perbandingan hasil implementasi dari metode klasifikasi SVM dan NBC?
4. Bagaimana evaluasi terhadap hasil klasifikasi dengan menggunakan metode terbaik?

1.3 Batasan Masalah

Batasan masalah yang terdapat dalam penelitian ini adalah sebagai berikut:

1. Data yang digunakan adalah data sekunder berupa *Microarray gene expression* yang diambil dari *website* NCBI dengan *series* GSE76262 yang merupakan data *induced sputum* pada pasien yang menderita gangguan pernapasan atau asma
2. Data terdiri atas 3 kelas yaitu sampel dengan kelas *induced sputum severe*, *induced sputum moderate*, *induced sputum healthy control*
3. Metode yang digunakan untuk melakukan klasifikasi adalah metode *Support Vector Machine* (SVM) dan metode *Naive Bayes Classifier* (NBC).
4. *Software* yang digunakan dalam penelitian ini adalah aplikasi R Studio versi 4.2.2 dengan menggunakan beberapa *packages* bioinformatika.

1.4 Tujuan Penelitian

1. Mengetahui gambaran umum data ekspresi gen *induced sputum* pada pasien dengan gangguan pernafasan atau penyakit asma pada data *microarray*
2. Mengetahui hasil klasifikasi yang dihasilkan dari penerapan metode Support Vector Machine (SVM) dan *Naïve Bayes Classification* (NBC).
3. Mengetahui hasil perbandingan implementasi dari metode klasifikasi SVM dan NBC
4. Mengetahui hasil evaluasi klasifikasi dengan menggunakan metode terbaik

1.5 Manfaat Penelitian

Penelitian ini dibuat dengan harapan dapat memberikan beberapa manfaat bagi banyak pihak, diantaranya adalah sebagai berikut:

1. Bagi Penulis

   Sebagai implementasi pengetahuan yang diperoleh dari bangku perkuliahan dan memberikan sebuah *insight* baru dalam penelitian berdasarkan data ekspresi gen dalam bidang bioinformatika.

2. Bagi Pembaca

   - Menginformasikan klasifikasi data ekspresi gen *induced sputum* pada kasus pasien penderita penyakit asma.
   - Memberikan gambaran penggunaan metode SVM dan NBC dalam klasifikasi data ekspresi gen *induced sputum* pada kasus pasien penderita penyakit asma.

3. Bagi Penelitian Selanjutnya

   Untuk penelitian selanjutnya, hasil penelitian ini dapat digunakan sebagai referensi dalam penelitian sejenis atau dapat dikembangan dan dianalisa lebih dalam dengan menambahkan beberapa faktor dalam penelitian terkait, serta membandingkan atau mengkomparasikan beberapa metode klasifikasi lainnya yang diharapkan dapat memperoleh hasil kinerja yang lebih baik.

BAB II
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Bagian ini memuat studi atau penelitian sebelumnya yang relevan dengan topik penelitian yang sedang dibahas. Bagian ini sangat penting untuk menyajikan temuan-temuan yang telah dicapai pada penelitian sebelumnya, serta kemajuan dan penerapan terbaru dari teori yang akan diteliti.

2.1.1 Penelitian Menggunakan *Support Vector Machine* (SVM)

(Devi & Venkatesulu, 2015) dengan penelitiannya yang judul "*Gene Expression Data Classification using Support Vector Machine and Mutual Information-based Gene Selection*" menjelaskan tentang *support vector machine* sebagai algoritma *machine learning* dalam menentukan pendekatan yang paling optimal dalam melakukan klasifikasi dengan menggunakan dua *datasets microarray*. Dataset 1 berupa data kanker usus besar, lalu dataset 2 adalah data tentang kanker limfoma, atau kanker kelenjar getah bening. Digunakan 4 kernel dasar dalam SVM (*Linier, Quadratic, Polynomial, Radial Basis Function* (RBF)) sebagai perbandingan hasil dari penelitian untuk kedua dataset tersebut, sehingga diperoleh hasil bahwa SVM dengan Kernel Linier merupakan metode yang paling baik dibandingkan kernel lainnya, dengan nilai akurasi untuk dataset kanker usus besar sebesar 0.6774 atau 67.74%, sedangkan untuk dataset kedua atau data kanker kelenjar getah bening, didapatkan nilai akurasi nya sebesar 0.9777 atau 97.77%

Selanjutnya sebuah penelitian dari (Zhao, et al., 2020) dengan judul "*Applications of Support Vector Machine in Genomic Prediction in Pig and Maize Populations*". Pada penelitian ini membahas tentang penerapan *support vecotr machine* dalam memprediksi *dataset* genom pada babi dan jagung. Digunakan 5 dataset pada data genom babi yang diambil dari 5 *traits* atau sifat babi yang berbeda-beda, sedangkan pada data genom jagung menggunakan 3 dataset, dari 3 sifat waktu pembungaan pada jagung. Dari dataset genom babi dan jagung tersebut

selanjutnya dilakukan perbandingan hasil dari 4 kernel pada SVM, sehingga diperoleh hasil bahwa pada dataset genom babi, kernel dengan nilai akurasi terbaik yaitu kernel *Radial Basis Function* (RBF), sedangkan pada dataset jagung, kernel linier merupakan kernel dengan akurasi tertinggi. Berdasarkan hasil dari kedua kernel terbaik tersebut, selanjutnya akan dilakukan pengujian dengan menggunakan metode prediksi yang paling umum digunakan dalam memprediksi dataset genom, dengan menggunakan *Bayesian multiple regression model* (BayesR) dan *Genomic best linear unbiased prediction* (GBLUP). Berdasarkan hasil dari perbandingan dengan ke-empat parameter tersebut, diperoleh hasil bahwa SVM mampu memberikan prediksi dengan hasil yang paling baik untuk 2 dari 8 total dataset, dan tetap mampu memberikan hasil prediksi yang baik untuk sisa dataset lainnya. Oleh karena itu SVM merupakan metode yang terbilang cukup baik dalam memprediksi data genom dari hewan dan tumbuhan

Penelitian selanjutnya dilakukan oleh (Prasojoe & Setyorini, 2021) dengan judul "Uji Konsep Paralel SVM dengan Dekomposisi SMO Pada Data Set Cancer Microarray". Dilakukan pengujian konsep *Parallel Support Vector Machine* (PSVM) menggunakan dekomposisi SMO dalam mendeteksi kanker serta mengklasifikasikannya menggunakan data microarray. Pada penelitian ini teknik yang diterapkan adalah *Sequential Minimal Optimization* (SMO) yang menggunakan *lagrange multipliers* sebagai penyelesaian pada masalah *quadratic programming* (QP) yang ditemukan selama pengujian. Dalam proses pengujian konsep dekomposisi SMO, dilakukan pemecahan dataset dalam beberapa subset, yang selanjutnya dilakukan pelatihan SMO pada masing-masing subset dan menggabungkan hasil dari pelatihan subset-subset tersebut ke dalam satu bentuk model klasifikasi SMO. Setelah itu dilakukan tahap evaluasi hasil menggunakan perbandingan akurasi dari *perfromance* Dekomposisi SMO dan non-Dekomposisi SMO, didaptkan hasil dari akurasi untuk Dekomposisi SMO sebesar 75%, dan non-Dekomposisi SMO sebesar 63%.

(Ramdaniah, 2019)pada penelitiannya yang berjudul "*Performance evaluation of naive Bayes and support vector machine in type 2 Diabetes Mellitus gene expression microarray data*" membahas tentang pengevaluasian kinerja dari

algoritma *machine learning* dalam mengklasifikasikan data *microarray* ekspresi gen. Algoritma yang digunakan dalam penelitian ini adalah *Naive Bayes* dan *Support Vector Machine* (SVM). SVM menggunakan banyak fungsi kernel seperti Linear, Radial Basis Function (RBF), Polynomial, dan Sigmoid. Information gain digunakan untuk memilih fitur-fitur pada dataset GSE18732 dengan memilih 10, 20, 30, 40, dan 50 fitur teratas. Performa algoritma dievaluasi dan dibandingkan dengan menggunakan 30% set pengujian dan 20% set pengujian. Hasil penelitian menunjukkan bahwa SVM yang menggunakan kernel Polynomial memiliki kinerja yang paling baik jika dibandingkan dengan algoritma lainnya. Kernel polynomial mencapai akurasi 98,15% dengan menggunakan 30% *testting dataset* dan mencapai akurasi 100% dengan menggunakan 20% *testting dataset*.

2.1.2 Penelitian Menggunakan *Naïve Bayes Classifier*

Pada penelitian (Purnama, Astuti, & Adiwijaya, 2021) yang berjudul "Analisis Perbandingan Klasifikasi Microarray menggunakan *Naïve Bayes dan Support Vector Machine* (SVM) untuk Deteksi Kanker dengan *Feature Extraction* PCA". *Microarray* data digunakan sebagai bahan pengamatan yang digunakan untuk dianalisa, sehingga diperoleh suatu permasalahan yang ingin dianalisis yaitu, apakah orang atau sampel yang dianalisis tersebut terdiagnosa kanker atau tidak kanker. Analisis pada data *microarray* seringkali terjadi beberapa permasalahan, salah satunya adalah jumlah variabel ataupun atribut yang perbedaannya jumlah nya yang signifikan dibandingkan dengan jumlah sampel, sehingga pada proses ini perlu dilakukan metode reduksi dimensi. Digunakan salah satu teknik reduksi dimensi, yaitu Principal Component Analysis (PCA) dengan 2 algoritma *machine learning* yang akan digunakan untuk melakukan klasifikasi, yaitu *Naïve Bayes* dan SVM. Dilakukan perbandingan dari hasil analisis klasifikasi kedua metode tersebut dengan melihat nilai akurasi dari hasil penelitian. Berdasarkan hasil penelitian tersebut diperoleh kesimpulan akhir bahwa bahwa 4 dari 5 data kanker mendapatkan akurasi sebesar 77-96% sedangkan 1 data lainnya yaitu data kanker payudara mendapatkan akurasi paling kecil yaitu sebesar 54.6%.

Selanjutnya penelitian dari (Aini, Yulita, & Achmad, 2018) dengan judul" Seleksi Fitur Information Gain untuk Klasifikasi Penyakit Jantung Menggunakan Kombinasi Metode *K-Nearest Neighbor* dan *Naïve Bayes*", Penelitian ini membahas tentang penerapan seleksi fitur *Information Gain* dengan menggunakan algoritma *machine learning* dari *K-Nearest Neighbor (KNN)* dan *Naïve Bayes* untuk mengatasi masalah efektivitas dan akurasi dalam klasifikasi penyakit jantung. Algoritma *Information Gain* digunakan untuk mereduksi dimensi atribut untuk mendapatkan atribut yang relevan. Setelah proses seleksi fitur Information Gain selesai, proses selanjutnya adalah melakukan klasifikasi dengan menggunakan KNN untuk atribut numerik dan Naïve Bayes untuk atribut kategorik. Hasil dari penelitian ini menunjukkan nilai akurasi sebesar 92.31% saat pengujian distribusi kelas seimbang menggunakan 6 fitur dengan nilai K=25 dan saat pengujian distribusi kelas tidak seimbang pada 4 fitur dengan nilai K=35. Berdasarkan hasil tersebut dapat disimpulkan bahwa algoritma seleksi fitur Information Gain dengan kombinasi KNN dan Naïve Bayes dapat digunakan untuk klasifikasi penyakit jantung.

(Nor Azam, Zakaria, Hassan, & Zulkifle, 2022) dalam penelitiannya yang berjudul "Classification of Psoriasis Microarray Data using Machine Learning" membahas tentang Psoriasis yang merupakan kondisi kulit genetik papuloskuamosa kronis yang umum terjadi dan mempengaruhi orang-orang dari segala usia. Penelitian ini dilakukan untuk mengidentifikasi jenis lesi kulit pada pasien dengan menggunakan nilai optimal dari hasil klasifikasi algoritma *naïve bayes* dan SVM dengan melakukan tuning parameter berdasarkan kernel and, prior, dan var_smoothing, menggunakan algoritma optimasi Grid Search. Dataset yang digunakan adalah ekspresi gen kulit psoriasis yang diperoleh dari NCBI GEO Database. Dari hasil tuning parameter, SVM dengan kernel linear dan sigmoid berhasil meningkatkan performanya dari uji coba awal. Sementara itu, parameter Naïve Bayes default merupakan parameter yang optimal dalam mengklasifikasikan penyakit psoriasis dan tetap mempertahankan hasil performa awal. Kesimpulannya, memang benar bahwa setiap parameter mempengaruhi performa model klasifikasi. Pada akhir penelitian, model SVM yang telah dilakukan *tunning parameter* dengan

kernel sigmoid dan parameter yang signifikan terpilih sebagai machine learning yang tepat dalam mengklasifikasikan ekspresi gen penyakit psoriasis dengan tingkat akurasi sebesar 96%.

Sebuah penelitian dari (Suharman & Hartono, 2022) yang berjudul "Klasifikasi Kematangan Manggis Berdasarkan Fitur Warna dan Tekstur Menggunakan Algoritma Naive Bayes" Penelitian ini dilakukan untuk mengklasifikasikan kematangan buah manggis dengan menggunakan algoritma Naive Bayes berdasarkan warna dan tekstur. Fitur-fitur warna dan tekstur seperti kontras, korelasi, energi, homogenitas, entropi, standar deviasi, rata-rata, varians, skewness, dan kurtosis diekstraksi dari berbagai jenis gambar seperti RGB, grayscale, HSV, dan CIELAB. Setelah itu, dilakukan seleksi fitur dengan menggunakan algoritma MRMR. Metode klasifikasi yang digunakan adalah Naive Bayes dengan model yang menggunakan 13 parameter seperti *mean R, mean G, standar deviasi G, mean Saturation, mean Hue, standar deviasi Hue, standar deviasi Value, mean a*, mean b*, standar deviasi a*, standar deviasi b*, varian a**, dan kontras. Didapatkan tingkat akurasi untuk kelas matang sebesar 95,7% dengan sensitivitas 93,3%, spesifisitas 96,8%, dan presisi 93,3%.

2.1.3 Penelitian Tentang Asma

Sebuah penelitian dari (Akbar & WU, 2019) dengan judul "*Machine Learning Classifiers for Asthma Disease Prediction: A Practical Illustration*" yang bertujuan untuk memprediksi eksaserbasi parah yang disebabkan oleh asma yang tidak terkontrol. Digunakan 110 catatan tentang Asma dengan 7390 sampel yang dicari *dataset* stranskriptomik yang tersedia untuk umum dari *Gene Expression Omnibus* (GEO) pada situs NCBI (https://www.ncbi.nlm.nih.gov/geo/). Dataset yang digunakan tersebut kemudian dilakukan beberapa tahap pemrosesan data seperti *preprocessing*, *filtering* untuk mendapatkan kumpulan data tentang pasien asma yang siap dianalisis dengan klasifikasi klinis peserta yang ditentukan, dan ekspresi gen epitel bronkial menggunakan microarray. Empat algoritme klasifikasi pembelajaran mesin bernama *Naïve Bayes,* J48*, RandomForest, dan Random tree* digunakan dalam eksperimen ini untuk memprediksi penyakit asma pada tahap

awal. Performa keempat algoritme dievaluasi pada berbagai ukuran. Akurasi diukur pada kelas yang diklasifikasikan dengan benar dan salah. Setelah percobaan, diperoleh hasil masing-masing dari ke empat algoritma klasifikasi tersebut diperoleh nilai akurasi untuk klasifikasi menggunakan *naïve bayes* sebesar 98.75%, algoritma J48 memperoleh nilai akurasi sebesar 98.75%, *random forest* sebesar 97.7%, dan *random tree* sebesar 97%

Table 2.1 Penelitian Terdahulu

No	Tahun	Peneliti	Judul	Metode	Hasil Penelitian
1	2015	Devi Arockia Vanitha, Venkatesulu	*Gene Expression Data Classification using Support Vector Machine and Mutual Information-based Gene Selection*	*Support Vector Machine* dan *Mutual Information*	Diperoleh hasil bahwa SVM dengan Kernel Linier merupakan metode yang paling baik dibandingkan kernel lainnya, dengan nilai akurasi untuk dataset kanker usus besar sebesar 0.6774 atau 67.74%, sedangkan untuk dataset kedua atau data kanker kelenjar getah bening, didapatkan nilai akurasi nya sebesar 0.9777 atau 97.77%
2	2018	Aini Safitri, Yulita Sari, Achmad Arwan	Seleksi Fitur *Information Gain* untuk Klasifikasi Penyakit Jantung Menggunakan Kombinasi Metode *K-Nearest Neighbor* dan *Naïve Bayes*	*K-Nearest Neighbor* dan *Naïve Bayes*	Penelitian ini menghasilkan akurasi sebesar 92.31% pada pengujian distribusi kelas seimbang menggunakan 6 fitur dengan nilai K=25 dan pada pengujian distribusi kelas tidak seimbang pada 4 fitur dengan nilai K=35. Dapat disimulkan bahwa algoritma seleksi fitur information gain bersama *KNN* dan *Naïve Bayes* efektif untuk klasifikasi penyakit jantung

No	Tahun	Peneliti	Judul	Metode	Hasil Penelitian
3	2019	Ramdaniah Ramdaniah, Armin Lawi, Syafruddin Syarif	*Performance evaluation of naive Bayes and support vector machine in type 2 Diabetes Mellitus gene expression microarray data*	*Naive Bayes* dan *support vector machine*	Hasil penelitian menunjukkan bahwa SVM yang menggunakan kernel Polynomial memiliki kinerja yang paling baik jika dibandingkan dengan algoritma lainnya. Kernel polynomial mencapai akurasi 98,15% dengan menggunakan 30% testting dataset dan mencapai akurasi 100% dengan menggunakan 20% testting dataset.
4	2020	Wei Zhao, Xueshuang Lai, Dengying Liu, Zhenyang Zhang, Peipei Ma, Qishan Wang, Zhe Zhang, and Yuchun Pan	*Applications of Support Vector Machine in Genomic Prediction in Pig and Maize Populations*	*Support Vector Machine*	Berdasarkan hasil dari perbandingan dengan ke-empat parameter tersebut (SVM kernel linier, SVM kernel RBF, BayesR, dan GBLUP) diperoleh hasil bahwa SVM mampu memberikan prediksi dengan hasil yang paling baik untuk 2 dari 8 total dataset, dan tetap mampu memberikan hasil prediksi yang baik untuk sisa dataset lainnya. Oleh karena itu SVM merupakan metode yang terbilang cukup baik dalam memprediksi data genom dari hewan dan tumbuhan

No	Tahun	Peneliti	Judul	Metode	Hasil Penelitian
5	2021	Vina Mutiara Purnama, Widi Astuti, Adiwijaya	Analisis Perbandingan Klasifikasi Microarray menggunakan *Naïve Bayes dan Support Vector Machine* (SVM) untuk Deteksi Kanker dengan *Feature Extraction* PCA	*Naïve Bayes* dan *Support Vector Machine*	Berdasarkan penelitian tersebut didapatkan nilai akurasi dari hasil penelitian ini menunjukkan bahwa 4 dari 5 data kanker yang digunakan dalam penelitian, diperoleh akurasi sebesar 77-96%, sedangkan 1 data lainnya yaitu data kanker payudara mendapatkan akurasi paling kecil yaitu sebesar 54.6%.
6	2022	Rahmat Ramadan Prasojoe, Setyorini	Uji Konsep Paralel SVM dengan Dekomposisi SMO Pada *Data Set Cancer Microarray*	*Support Vector Machine*	Berdasarkan penelitian ini didapatkan hasil dari *dataset* yang telah dilakukan pemecahan menjadi beberapa *subse datat*, dan hasil pelatihan masing-masing *subset* tersebut digabungkan dalam satu bentuk klasifikasi SMO. Sehingga diperoleh *performance* klasifikasi berupa nilai akurasi dari dekomposisi SMO sebesar 75%, dan non-dekomposisi SMO sebesar 63%, serta waktu pelatihan dekomposisi SMO 5.7 kali lebih cepat daripada non-dekomposisi SMO

No	Tahun	Peneliti	Judul	Metode	Hasil Penelitian
7	2022	Raihan Abimanyu Suharman, Hartono Hartono	Klasifikasi Kematangan Manggis Berdasarkan Fitur Warna dan Tekstur Menggunakan Algoritma *Naive Bayes*	*Naïve Bayes*	Berdasarkan penelitian tentang klasifikasi kematangan buah manggis dengan menggunakan algoritma *Naive Bayes* diperoleh tingkat akurasi yang tinggi, yaitu sebesar 95,7%, dengan sensitivitas, spesifisitas, dan presisi untuk kelas matang sebesar 93,3%, 96,8%, dan 93,3%. Untuk kelas mentah, sensitivitas, spesifisitas, dan presisi masing-masing mencapai 100%, sedangkan untuk kelas semi-matang, sensitivitas, spesifisitas, dan presisi masing-masing adalah 93,3%, 96,9%, dan 93,3%.
8	2022	Siti Nor Zulaika Nor Azam, Noor Hidayah Zakaria, Rohayanti Hassan;, Farizuwana Akma Zulkifle	Classification of Psoriasis Microarray Data using Machine Learning	*Support Vector Machine* dan *Naïve Bayes*	Berdasarkan penelitian menggunakan SVM (kernel linear dan sigomoid) serta menggunakan metode *naïve* bayes, diperoleh hasil bahwa SVM kernel sigmoid yang telah dilakukan *tunning parameter* sehingga memperoleh parameter yang signifikan, terpilih sebagai machine learning yang tepat dan terbaik dalam mengklasifikasikan data ekspresi gen pada penyakit psoriasis dengan tingkat akurasi sebesar 96%.

No	Tahun	Peneliti	Judul	Metode	Hasil Penelitian
9	2019	Akbar Wasif, WU WEI-PING, Mushtaq Muhammad	Machine Learning Classifiers for Asthma Disease Prediction: A Practical Illustration	*Naïve Bayes,* J48, *Random Forest*, *Random Tree*	Berdasarkan hasil penelitian untuk memprediksi eksaserbasi parah dikarenakan asma yang tidak terkontrol, dengan melihat catatan kesehatan pasien. Digunakan empat algoritma *machine learning* dalam melakukan prediksi klasifikasi, yaitu (*naïve bayes*, J48, *random forest*, *random tree*) didapatkan hasil *machine learning* yang paling baik dalam mengklasifikasikan data pasien penyakit asma ialah pada metode *naive bayes* dengan nilai akurasi sebesar 98.75%

BAB III
LANDASAN TEORI

3.1 Bioinformatika

Bioinformatika merupakan cabang ilmu baru dari gabungan multidispilin ilmu seperti ilmu biologi, medis, teknik informasi, matematika, statistika, dan beberapa cabang ilmu terkait yang lain. Penerapan bioinformatika dilakukan dengan menggabungkan berbagai teknik dari bidang matematika terapan, statistika, informatika, ilmu komputer, sistem informasi, intelegensi buatan, kimia, dan biokimia untuk menyelesaikan masalah dalam bidang biologi pada tingkat molekuler. (Santoso, Mariyah, Yuniarto, Pramana, & Nooraeni, 2018). Penerapan bioinformatika telah membantu mengatasi berbagai masalah di berbagai bidang, termasuk di bidang klinis. Sebagai contoh, penerapan bioinformatika dalam bidang klinis meliputi berbagai hal, beberapa hal tersebut seperti identifikasi gen penyakit baru, diagnosis penyakit, dan juga bisa digunakan sebagai bahan anlisis untuk pembuatan obat-obatan

Salah satu teknik analisis dalam bioinformatika adalah analisis ekspresi gen, yang melibatkan pengukuran level sejumlah gen menggunakan metode tertentu seperti *microarray*. (Raza & Khalid, 2010)

3.1.1 Microarray

Mesin Microarray RNA merupakan sebuah alat yang diciptakan pada akhir tahun 1990-an untuk mengukur ekspresi gen (Gonzalo Sanz & Sánchez, 2018). Saat ini, teknologi *Microarray* telah berkembang pesat dan hal tersebut ditandai dengan *microarray* jutaan probe yang bisa terkandung dalam teknologi ini, dimana pada *microarray* terdapat DNA dan RNA termasuk intron, ekson, dan beberapa aspek lain dari gen-gen yang ingin dianalisis pada *microarray*. Analisis data *microarray* menjadi salah satu analisis yang telah digunakan pada berbagai persoalan dalam analisis biologis, beberapa penelitiannya mencakup, pemilihan gen ekspresi gen yang berbeda, pembangunan prognosis, serta penemuan cluster dalam data.

Tahapan-tahapan dalam melakukan analisis *microarray* ialah sebagai berikut:

- Eksplorasi Data
- Kontrol Kualitas
- Normalisasi
- Analisis Statistik
- Signifikansi Biologis atau Analisis Jalur.

Terdapat berbagai media analisis ataupun *software* yang dapat digunakan untuk menganalisis data *microarray*, salah satunya adalah perpustakaan *Bioconductor* (http://bioconductor.org) yang menggunakan bahasa statistik R.

3.1.2 Ekspresi Gen

Menurut (Perdew, Vanden Heuvel, & Peters, 2007) Ekspresi gen adalah sintesis produk gen fungsional menggunakan informasi yang disediakan oleh asam deoksiribonukleat (DNA). Dalam proses ekspresi gen, asam ribonukleat (RNA) disintesis dari DNA melalui proses transkripsi. Ekspresi gen didefinisikan sebagai proses di mana informasi yang dikodekan dalam gen digunakan untuk mengarahkan sintesis produk gen fungsional (García-Sánchez & Marqués-García, 2016). Sebagai suatu proses, ini dapat menjelaskan mengapa organisme yang mengandung sebagian besar DNA yang sama tetapi bisa menunjukkan jenis dan fungsi sel yang berbeda (Gibney & Nolan, 2010). Regulasi gen sangat penting dalam diferensiasi seluler pada organisme multisel, karena dapat berkontribusi pada fungsi dan struktur sel tertentu, dan merupakan bagian integral dari perkembangan organisme. Semua hal di atas membuktikan bahwa selain informasi genetik yang diwariskan, fungsi dan struktur sel dipengaruhi oleh informasi yang tidak dikodekan dalam urutan DNA. Informasi ini juga disebut informasi epigenetik (Gibney & Nolan, 2010).

3.2 Asma

Asma adalah inflamasi kronik saluran napas, meskipun penilaian derajat asma didasarkan pada gejala klinik dan faal paru, bukan inflamasi. Gejala asma bergantung pada persepsi penderita dan faal paru tergantung pada effort dan teknik

melakukannya. Inflamasi sudah terjadi pada asma dini dan ringan. Inflamasi mukosal terjadi sebelum disfungsi paru. Jarak antara inflamasi mukosal dan disfungsi paru belum diketahui. Inflamasi juga terjadi pada asma episodik bahkan pada saat tidak ada gejala. Limfosit T, sel mast dan eosinofil terlibat dalam inflamasi saluran napas, dengan limfosit T CD4 atau T helper yang berperan terutama. Inflamasi saluran napas merupakan faktor utama pada patogenesis asma. Meskipun biopsi bronkus dan bronchoalveolar lavage (BAL) dapat digunakan untuk menilai inflamasi saluran napas, cara ini merupakan tindakan invasif dan tidak nyaman untuk penderita. Sputum dapat digunakan sebagai spesimen untuk pemeriksaan, namun tidak semua pasien dapat membatuk sputum yang cukup untuk digunakan dalam pemeriksaan, sehingga perlu dilakukan induksi sputum. Spesimen harus diambil dari saluran napas bawah dengan kualitas yang baik dan sebaiknya sebelum pemberian antibiotik.

3.3 Induced Sputum

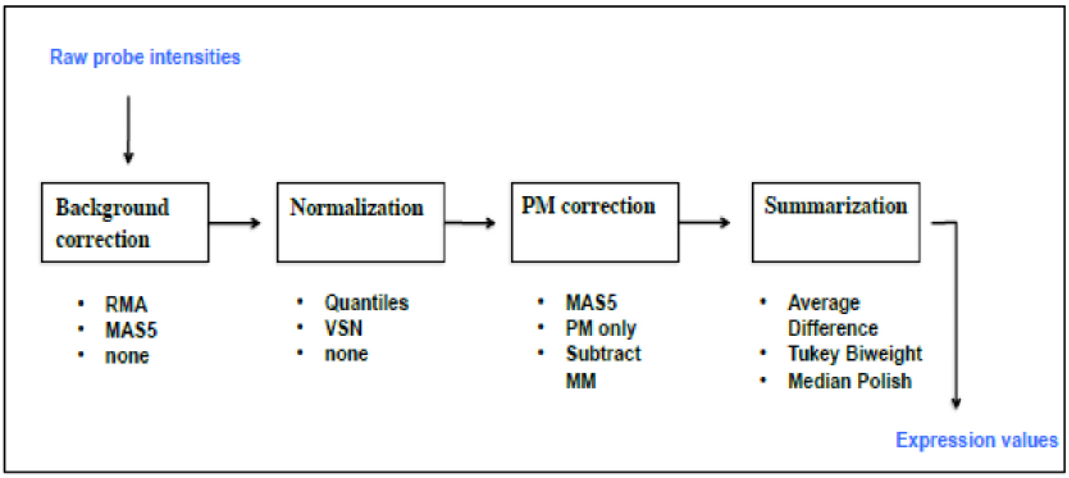
Pada abad ke-19, kegunaan pemeriksaan sputum pertama kali diusulkan setelah ditemukannya eosinofilia sputum pada penderita asma. Pada tahun 1992, Pin dan rekan-rekannya mengembangkan teknik untuk menginduksi produksi sputum pada penderita asma dengan menggunakan nebulisasi salin hipertonik. Teknik ini berhasil mendapatkan sputum yang cukup dari saluran napas bawah. Berbagai penelitian kemudian menunjukkan bahwa induksi sputum aman dan dapat dilakukan pada orang sehat maupun penderita asma ringan hingga berat. Sputum yang dihasilkan mencerminkan kondisi bronkus. Meskipun sputum terinduksi berkorelasi dengan bronchial washing dan BAL, tetapi ukurannya lebih kecil dari pada biopsi bronkus. Saat ini, belum ada metode induksi sputum yang standar, tetapi berbagai metode umumnya menerapkan prinsip-prinsip berikut:

1. Pemberian salbutamol sebagai bronkodilator awal
2. Monitoring fungsi paru-paru
3. Nebulisasi dengan nebulizer ultrasonik
4. Menggunakan konsentrasi cairan salin umumnya 3%, 4%, atau 5%.

Meskipun induksi sputum dapat menyebabkan bronkokonstriksi sebagai efek samping, ini dapat dicegah dengan pemberian salbutamol sebagai bronkodilator sebelum pemberian cairan salin. Selain itu, penggunaan salin hipertonik lebih efektif dalam menginduksi pengeluaran sputum daripada salin normal, dan tidak ada perbedaan hasil komposisi sel yang dihasilkan dari perbedaan konsentrasi salin. Penggunaan nebulizer ultrasonik juga lebih berhasil daripada nebulizer jet.

3.4 Pre-processing

Preprocessing adalah suatu proses yang bertujuan untuk menghilangkan pengaruh non-biologis pada data sehingga hasil analisis menjadi lebih akurat. Setiap tahap dari proses *preprocessing* memiliki pilihan yang berbeda-beda yang dapat dilakukan. Dengan melakukan proses *preprocessing*, kita dapat memperoleh data yang lebih bersih dan mengurangi kesalahan yang dapat terjadi selama proses analisis. Hal ini sangat penting karena dapat mempengaruhi hasil akhir dari analisis dan memastikan keakuratan dan keandalan hasil tersebut. (Serin, 2011).



Gambar 3.1 Alur *Pre-processing*

Pada gambar 3.5 digambarkan bahwa proses *preprocessing* terbagi atas tiga tahap yaitu *background correction, normalization,* dan *summerization*yang memiliki fungsi sebagai berkut (Bolstad, 2004):

1. *Background correction* adalah suatu metode yang digunakan untuk menghilangkan noise latar belakang pada data dan menyesuaikan sinyal

cross-hybridization yang disebabkan oleh pengikatan DNA non-spesifik pada *array*.

2. *Normalization* adalah proses yang digunakan untuk menghilangkan variansi non-biologis yang tidak diinginkan pada data *microarray*. Proses ini bertujuan agar data dapat dibandingkan secara akurat antara sampel-sampel yang berbeda.
3. *Summarization* adalah proses penggabungan beberapa intensitas probe menjadi probe set yang menghasilkan nilai ekspresi untuk setiap gen. Nilai ini mewakili rata-rata intensitas probe dan membantu mempermudah interpretasi hasil analisis.

3.5 *Filtering*

Filtering adalah proses pemilihan *subset probe* yang digunakan dalam analisis, dengan tujuan menghilangkan atau menyertakan beberapa *probe* yang diperlukan saja. *Filtering* merupakan salah satu langkah pada anlisis *microarray* yang dapat digunakan jika ingin meningkatkan kualitas model serta memberikan pemodelan yang lebih efisien dikarenakan beberapa data yang tidak dibutuhkan akan dihilangkan dengan menggunakan fungsi *nsfilter* (non-specific filter). Fungsi tersebut merupakan sarana ataupun metode dalam dalam *filtering* fitur dari data *expression set* secara keseluruhan. Fungsi *nsfilter* memiliki beberapa perintah yang digunakan, salah satunya adalah `var.cutoff` yang merupakan perintah untuk menghapus beberapa data ataupun variabel yang memiliki nilai IQR dibawah *quartile*, sehingga beberapa gen tertentu yang memiliki nilai IQR dibawah kuartil akan difilter atau dihilangkan. Selanjutnya terdapat perintah `Require.entrez TRUE` yang merupakan perintah sebagai proses filterisasi anotasi *Entrez Gene ID*, dimana yang digunakan hanya berupa d sistem pengenalan ID pusat. Perintah selanjutnya yaitu `Remove.DupEntrez TRUE` memiliki fungsi untuk menghapus ID gen *Entrez* yang memiliki kesamaan atau terduplikat dengan ID gen yang lain. Terakhir, *Feature.exclude* digunakan sebagai alat penyaring *probe* control, seperti AFFX, sehingga *probe-probe* yang tidak diinginkan tersebut tidak akan terikut sebagai data untuk dianalisis (Gentleman dkk, 2019).

3.6 *Feature Selection*

Feature selection atau yang dikenal dengan *gene selection* adalah teknik untuk memilih gen yang paling relevan dalam suatu analisis klasifikasi, sehingga membuat lebih ringkas waktu untuk melakukan pengklasifikasian dan meningkatkan akurasi (Karabulut dkk, 2012). Dalam melakukan feature selection menggunakan bantuan packages *multtest* dengan perintah `mt.teststat` yang mampu menghitung statistik uji diantara nya yaitu uji t-test, wicoxon, uji F, untuk setiap baris kerangka data. Pada penelitian ini digunakan ui F karena memiliki lebih dari dua sampel. Uji F menurut Kuncoro (2013) digunakan untuk menguji signifikan tidaknya pengaruh variabel independen secara simultan terhadap variabel dependen.

3.7 Statistika Deskriptif

Pada dasarnya, statistik dapat dikelompokkan menjadi dua jenis, yaitu statistik deskriptif dan statistik inferensial. Statistik deskriptif digunakan untuk merangkum karakteristik suatu data dengan memuat berbagai ukuran penting. (Walpole, 2011). Beberapa ukuran yang termasuk dalam statistik deskriptif meliputi mean, median, dan standar deviasi. Mean dan median digunakan sebagai ukuran pemusatan data, sedangkan standar deviasi digunakan untuk mengukur seberapa jauh data tersebar dari nilai rata-rata atau mean. (Johnson & Bhattacharyya, 2010).

Analisis klasifikasi menggunakan statistik deskriptif untuk membandingkan karakteristik dari kelompok atau kelas yang berbeda (Hair, Black, Babin, & Anderson, 2010). Contohnya, kita dapat menggunakan statistik deskriptif untuk membandingkan rata-rata, standar deviasi, dan distribusi data dari dua kelompok yang berbeda untuk menentukan apakah terdapat perbedaan yang signifikan antara kelompok-kelompok tersebut.

3.8 Support Vector Machine (SVM)

Metode klasifikasi SVM menerapkan prinsip *Structural Risk Minimization* (SRM) untuk menemukan hyperplane terbaik yang memisahkan dua kelas berbeda dalam input space. SVM juga disebut sebagai pengklasifikasi margin maksimum,

karena secara simultan mengurangi kesalahan klasifikasi empiris dan memaksimalkan margin geometrik (Nugroho dkk, 2003). Metode ini pertama kali ditemukan dan dikembangkan dalam kerangka pembelejaran statistik oleh Vapnik pada tahun 1992 dan sejak saat itu SVM mulai menarik minat yang tinggi dalam komunitas penelitian *machine learning*. Pada awalnya SVM dirancang hanya untuk *linear classifier*, akan tetapi seiring perkembangannya SVM dapat digunakan dalam memecahkan permasalahan *non-linear classifier*.

Pada bukunya yg berjudul "Teknik pemanfaatan data untuk keperluan bisnis" (Santosa, 2007) menjelaskan bahwa *hyperplane* SVM klasifikasi linier dapat dituliskan dengan menggunakan perhitungan berikut:

$$f(x) = W^T x + b \quad (3.1)$$

SVM menggunakan konsep *hyperplane* terbaik untuk membedakan dua kelas yaitu kelas positif (+1) dan kelas negatif (-1), dimana w adalah bobot vektor dan b adalah nilai bias. Fungsi klasifikasi SVM didefinisikan oleh ekspresi:

Apabila suatu *pattern* x_i merupakan golongan *negative class* atau kelas -1, akan menjadi *pattern* yang memenuhi pertidaksamaan berikut:

$$w.x_i + b \leq -1 \quad (3.2)$$

Sedangkan *Pattern* yang masuk ke dalam golongan *positive class* atau kelas +1 dirumuskan sebagai berikut:

$$w.x_i + b \geq +1 \quad (3.3)$$

Keterangan:

w : vektor bobot

x : nilai masukan atribut

b : bias

Suatu *hyperplane* terbaik dapat diperoleh dari *hyperplane* yang memiliki margin maksimum atau optimal, optimal margin tersebut dapat diperoleh ketika jarak suatu *hyperplane* sudah semaksimal mungkin dengan *patternt* terdekatnya, yaitu $\frac{1}{\|w\|}$, dengan $\|w\|$ merupakan *norm* dari *weight* vector w. Bentuk pencarian optimal *hyperplane* seperti yang dijelaskan sebelumnya, sering kali disebut sebagai masalah *quadratic programming* (QP), dimana titik minimal dari

$$min\ \tau(w) = \frac{1}{2}\ \|\ w\ \|^2 \quad (3.4)$$

Akan ditemukan dengan batasan persamaan

$$y_i(x_i.w + b) - 1 \geq 0, i = 1,2,3, \ldots, l \quad (3.5)$$

Setelahnya akan digunakan teknik komputasi *lagrange multipliers* untuk menyelesaikan permasalahan diatas, dengan menggunakan persamaan seperti dibawah ini

$$L(w,b,a) = \frac{1}{2}\ \|\ w\ \|^2 - \sum_{i=1}^{l} a_i\ y_i\ (x_i.w + b) - 1, i = 1,2,3, \ldots, l \quad (3.6)$$

Dimana $a_i \geq 0$ adalah nilai koefisien *lagrange multipliers*. Selanjutnya meminimumkan nilai L terhadap w dan b, sehingga diperoleh:

$$\sum_{i=1}^{l} a_i\ y_i\ = 0 \quad (3.7)$$

$$w = \sum_{i=1}^{l} a_i\ y_i\ x_i \quad (3.8)$$

Setelah itu, dilakukan modifikasi pada persamaan 3.5 agar dapat memaksimalkan nilai a_i dalam masalah maksimalisasi, sehingga diperoleh persamaan baru:

$$L(a) = \sum_{i=1}^{n} a_i - \frac{1}{2} \sum_{i=1.j=1}^{n} a_i\ a_j y_i y_j x_i x_j \quad (3.9)$$

$$\sum_{i=1}^{n} a_i y_i = 0, a_i\ \geq 0 \quad (3.10)$$

Dari hasil tahapan sebelumnya, nilai a_i yang < 0 disebut sebagai support vector, sedangkan nilai a_i yang lainnya = 0. Fungsi keputusan yang dihasilkan hanya bergantung pada nilai support vector. Pada umumnya, kebanyakan masalah tidak dapat diselesaikan dengan data yang terpisah secara linear, oleh karena itu SVM menggunakan fungsi kernel untuk menyelesaikan masalah nonlinear. Fungsi kernel dapat memetakan sampel data ke dalam ruang dimensi yang lebih tinggi, sehingga dapat menyelesaikan kasus di mana hubungan antara kelas dan atributnya tidak linear. SVM memiliki empat jenis kernel yang dapat digunakan untuk menyelesaikan masalah linear dan non-linear.

1. Kernel Linear

$$K(x, x_k) = x_k^T x \quad (3.11)$$

2. Kernel Polynomial

$$K(x, x_k) = (x_k^T x + 1)^d \quad (3.12)$$

3. Kernel Gaussian Radial Basis Function (RBF)

$$K(x, x_k) = \exp\{-\| x - x_k \|_2^2 / \sigma^2\} \quad (3.13)$$

4. Sigmoid

$$K(x, x_k) = tanh\{k x_k^T x | \theta]\quad (3.14)$$

Performa model SVM tergantung pada penggunaan fungsi kernel dan parameter C serta gamma yang digunakan. Penggunaan nilai C yang besar akan mengurangi kesalahan klasifikasi pada data. Sementara, parameter gamma digunakan pada kernel RBF untuk mentransformasi data train ke dalam ruang fitur, yang kemudian dioptimalkan menggunakan metode Lagrange Multipliers. Hal ini menghasilkan nilai α yang digunakan untuk menentukan support vector serta memperkirakan koefisien bobot (w) atau bias (b) pada model klasifikasi. (Handayani & Jamal, 2016).

3.9 Naïve Bayes

Algoritma *Naive Bayes* merupakan metode klasifikasi berdasarkan teorema *Bayes* yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, di mana perhitungannya didasarkan pada probabilitas dan statistik (Informatikalogi, 2017). Algoritma ini melakukan estimasi peluang di masa depan berdasarkan pengalaman sebelumnya. *Naive Bayes* mengasumsikan semua atribut bersifat independen (tidak saling berpengaruh). Asumsi independensi yang kuat (naif) tersebut menjadikan syarat peluang yang sederhana dan mudah dihitung. Kaitan antara teorema *Bayes* dengan klasifikasi yaitu hipotesis atau dugaan dalam teorema *Bayes* merupakan label kelas yang menjadi target dalam klasifikasi (Wasiati & Wijayanti, 2014). Berikut adalah persamaan dari teorema *Bayes* (Bustami, 2014):

$$P(B|A) = \frac{P(B \cap A)}{P(A)} \quad (3.15)$$

$$P(B \cap A) = P(B|A)\, P(A) \quad (3.16)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{P(B|A)\, P(A)}{P(B)} \quad (3.17)$$

Sehingga dari persamaan diatas akan menjadi seperti persamaan berikut:

$$P(Y|X) = \frac{P(X|Y)P(Y)}{P(X)} \quad (3.18)$$

Keterangan:

X = data dengan *class* yang belum diketahui

Y = hipotesis data X merupakan suatu kelas spesifik

$P(Y|X)$ = probabilitas hipotesis Y berdasarkan kondisi X (*posterior probability*)

$P(Y)$ = probabilitas hipotesis Y (*prior probability*)

$P(X|Y)$ = probabilitas X berdasarkan kondisi pada hipotesis Y

$P(X)$ = probabilitas X

Perhitungan klasifikasi dengan teorema *Bayes* diatas membutuhkan sejumlah petunjuk untuk menentukan kelas apa yang lebih cocok untuk sebuah sampel yang dianalisis. Penjabaran detail klasifikasi dengan teorema *Bayes* dapat dilihat pada persamaan berikut (Bustami, 2014):

$$P(Y_j|X_1, \dots, X_n) = \frac{P(X_1,\dots,X_n|Y)P(Y_j)}{P(X_1,\dots,X_n)} \quad (3.19)$$

Keterangan:

P = Peluang

Y_j = sub kelas Y yang dicari

X = variabel $X_1, \dots, X_n$ dengan karakteristik petunjuk untuk klasifikasi

Probabilitas hipotesis Y berdasarkan kondisi X biasa disebut dengan *posterior probability,* probabilitas X berdasarkan kondisi pada hipotesis Y disebut juga dengan *likelihood,* probabilitas hipotesis Y disebut juga dengan *prior probability*, sedangkan probabilitas X disebut juga dengan *evidence.* Oleh karena itu, rumus pada persamaan (3.16) diatas dapat juga ditulis seperti berikut (Bustami, 2014):

$$posterior = \frac{likelihood \times prior}{evidence} \quad (3.20)$$

Oleh karena klasifikasi *Naive Bayes* mempunyai asumsi independensi yang kuat, maka masing-masing petunjuk $(X_1, \dots, X_n)$ saling bebas (independen).

Berdasarkan asumsi tersebut, maka berlaku persamaan sebagai berikut (Bustami, 2014):

$$P(X_i|X_j) = \frac{P(X_i \cap X_j)}{P(X_j)} = \frac{P(X_i).P(X_j)}{P(X_j)} = P(X_i) \quad (3.21)$$

Di mana $i \neq j$, sehingga

$$P(X_i|Y_j, X_j) = P(X_i|Y_j) \quad (3.22)$$

Atau dapat dituliskan sebagai berikut:

$$P(Y_j|X_1, \dots, X_n) = P(Y_j) \prod_{i=1}^{n} P(X_i|Y_j) \quad (3.23)$$

Kemudian dapat dijabarkan sebagai berikut:

$$P(Y_j|X) = P(X_1|Y_j).P(X_2|Y_j) \dots P(X_n|Y_j).P(Y_j) \quad (3.24)$$

Persamaan (3.21) diatas merupakan model dari teorema *Naive Bayes* yang selanjutnya akan digunakan untuk proses klasifikasi. Berikut adalah tahapan dalam kalsifikasi *Naive Bayes*:

- Langkah 1: Menghitung nilai *prior probability* atau *class probability* pada masing-masing variabel kelas, dengan mengggunakan rumus peluang kejadian sebagai berikut (Informatikalogi, 2017):

  $$prior = P(Y_j) = \frac{n(Y_j)}{n(S)} \quad (3.25)$$

  Dengan $0 \leq P(Y) \leq 1$.

  Keterangan:

  $n(Y_j)$ = banyak anggota dalam kejadian sub kelas Y

  $n(S)$ = banyak anggota dalam himpunan ruang sampel

- Langkah 2: Menghitung nilai *conditional probability* atau nilai probabilitas setiap variabel independen terhadap variabel kelas. Jika variabel independen berbentuk bilangan kontinu, maka diperlukan pemodelan ulang bentuk data dengan menghitung nilai *mean* (rata-rata) dan sd (standar deviasi) masing-masing variabel.

  Rumus *mean* data tunggal (Rumus Statistik , 2013):

  $$Mean(\bar{X}) = \frac{\sum_{i=1}^{n} X_i}{n} \quad (3.26)$$

  Keterangan:

  $\bar{X}$ = rata-rata sampel

X_i = nilai X ke i

n = banyak sampel

Rumus standar deviasi data tunggal (Rumus Statistik, 2013):

$$S = \sqrt{\frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n-1}} \quad (3.27)$$

Keterangan:

S = standar deviasi (simpangan baku) sampel

X_i = nilai X ke i

n = banyak sampel

Setelah nilai *mean* dan standar deviasi didapatkan, maka selanjutnya dapat dihitung nilai *likelihood* atau $P(X_i|Y_j)$ dengan menggunakan rumus Densitas Gauss sebagai berikut (Informatikalogi, 2017)::

$$P(X_i|Y_j) = \frac{1}{\sqrt{2\pi S_{ij}}} e^{-\frac{(X_i - \bar{X}_{ij})^2}{2S_{ij}^2}} \quad (3.28)$$

Keterangan:

P = peluang

X_i = nilai X ke i

Y_j = sub kelas Y yang dicari

π = nilai konstanta yaitu 3,14

$\bar{X}_{ij}$ = rata-rata X ke i berdasarkan sub kelas Y

S_{ij} = standar deviasi X ke i berdasarkan sub kelas Y

- Langkah 3: menentukan prediksi kelas Y dengan menghitung nilai *posterior probability*, di mana kelas dengan nilai *posterior probability* tertinggi merupakan kelas yang sesuai untuk sampel yang dianalisis. Perhitungan nilai *posterior probability* dapat dilihat pada persamaan berikut (Informatikalogi, 2017)::

$$P(Y_j|X_i) = P(Y_j) \prod_{i=1}^{n} P(X_i|Y_j) \quad (3.29)$$

Keterangan:

P = Peluang

Y_j = sub kelas Y yang dicari

X_i = nilai X ke i

3.10 Penanganan Data *Imbalanced*

Masalah umum yang terjadi pada metode klasifikasi data *mining* adalah ketidakseimbangan data, yang disebut sebagai data yang tidak seimbang dikarenakan terdapat perbedaan jumlah atau memiliki perbedaan yang signifikan antara kelasnya (Chawla, 2002). Data yang tidak seimbang dapat menghasilkan klasifikasi yang tidak seimbang, dimana kelas mayoritas (yang paling sering muncul) lebih dominan dibandingkan dengan kelas minoritas (yang muncul lebih sedikit). Untuk mengatasi ketidakseimbangan kelas, teknik resampling seperti *Synthetic Minority Over Sampling Technique* (SMOTE) dapat digunakan.

SMOTE adalah teknik resampling yang menciptakan sampel sintetis untuk kelas minoritas dengan jumlah data yang ditentukan, sehingga jumlah data untuk setiap kelas menjadi sama (Chawla, 2002). Misalnya diketahui jumlah suatu data sebanyak 100 dengan 4 kelas, maka dengan teknik SMOTE dihasilkan jumlah data masing-masing kelas sebanyak 25 (100 dibagi 4). SMOTE dilakukan dengan menambah jumlah data pada kelas minoritas dengan cara membangkitkan data baru berdasarkan k tetangga terdekat. Data pada kelas minoritas dilakukan *oversampling* dengan mengambil data pada kelas minoritas dan menambah sampel sintetis di sepanjang garis yang menghubungkan salah satu atau semua k tetangga terdekat data kelas minoritas tersebut. Jumlah tetangga k dipilih secara acak. Formula untuk membangkitkan data sintetis dengan SMOTE adalah sebagai berikut:

$$X_{new} = X_i + (\hat{X}_k - X_i)\,\delta \qquad (3.30)$$

Dimana:

X_{new} = data sintesis baru

X_i = data dari kelas minoritas

$\hat{X}_k$ = data dari k tetangga terdekat dengan X_i

δ = bilangan acak dari 0 dan 1

Pendekatan SMOTE tidak hanya menangani pada kumpulan data dengan fitur yang memiliki skala nominal (SMOTE-N) saja, akan tetapi pendekatan SMOTE dapat pula menangani semua kumpulan data dengan fitur yang memiliki

skala campuran, yakni nominal dan kontinu (SMOTE-NC). Perhitungan tetangga terdekat kelas minoritas untuk fitur nominal (SMOTE-N) dilakukan dengan menggunakan *value difference metric* (VDM) (chawla, et al 2002).

$$\Delta(X, Y) = W_x W_y \sum_{i=1}^{N} \delta(X_i, X_y)^r \quad (3.31)$$

Keterangan:

$W_x W_y$ = bobot amatan

N = banyaknya fitur penjelas

r = bernilai 1 apabila jarak *manhattan* atau 2 apabila jarak *Euclid*

$\delta(X_i, X_y)$ = jarak antar fitur nominal

Tahapan metode yang dilakukan adalah:

1. Menghitung jarak antar amatan pada kelas minor menggunakan rumus VDM
2. Menentukan nilai k yaitu 5 dan persentase oversampling sebesar 4200
3. Dipilih satu contoh dari kelas minor secara acak.
4. Menentukan amatan k tetangga terdekat dengan mengurut jarak contoh terpilih dengan semua amatan pada kelas minor.
5. Data sintesis dibuat dengan menentukan nilai per peubah penjelasnya. Nilai tersebut diperoleh dari mayoritas nilai pada k tetangga terdekat. Jika semua peubah telah dibuat maka diperoleh satu amatan baru.
6. Langkah 3 hingga 5 dilakukan berulang hingga banyaknya oversampling yang diinginkan telah tercapai.

3.11 Evaluasi Metode dan Tabel Klasifikasi

Hasil pengujian yang diperoleh dari metode klasifikasi perlu dilakukan evaluasi, yaitu bertujuan untuk mengetahui seberapa besar performa yang dihasilkan dari metode klasifikasi yang digunakan. Pengukuran performa klasifikasi dapat menggunakan *Confusion Matrix*, di mana keluaran yang dihasilkan dapat berupa dua kelas atau lebih. *Confusion Matrix* merupakan tabel perbandingan antara nilai aktual dengan nilai prediksi. Berikut adalah contoh *Confusion Matrix* empat kelas yang dapat dilihat pada tabel 3.1 berikut (Khalimi, 2020)

Tabel 3.1 Contoh *Confusion Matrix* 4 Kelas

Prediksi	Aktual			
	A	B	C	D
A	2	0	1	0
B	3	5	1	0
C	1	0	5	0
D	0	0	0	6

Berdasarkan contoh *Confusion Matrix* pada tabel 3.1 diatas, maka dapat dilakukan perhitungan nilai akurasi. Akurasi merupakan tingkat kedekatan nilai prediksi dengan nilai sebenarnya. Namun penggunaan nilai akurasi kurang tepat apabila digunakan pada kasus kelas yang *imbalancedd.* Menurut Purwa (2019), ukuran performa klasifikasi yang dapat digunakan untuk kasus kelas *imbalancedd* diantaranya yaitu sensitivitas (*recall*), spesifitas, dan *G-Mean* (*Geometric Mean*). *Recall* yaitu rasio atau proporsi *true positive* yang diklasifikasi dengan benar . Spesifitas yaitu rasio atau proporsi *true negative* yang diklasifikasi dengan benar. Sedangkan *G-Mean* yaitu tingkat keseimbangan antara proporsi *true positive* (*recall*) dan proporsi *true negative* (spesifitas). Semakin tinggi nilai *G-Mean* maka menunjukkan hasil prediksi klasifikasi yang semakin baik atau lebih seimbang. Berikut adalah perhitungan nilai akurasi, *recall,* spesifitas, dan *G-Mean* (Prasetyowati & Ramadhani, 2018):

- Akurasi $= \frac{TP+TN}{\text{Jumlah Data}} \times 100\% = \frac{2+5+5+6}{24} \times 100\% = 75\%$
- *Recall* A $= \frac{TP}{(TP+FN)} = \frac{2}{(2+4)} = 0{,}33$

  Recall B $= \frac{TP}{(TP+FN)} = \frac{5}{(5+0)} = 1$

  Recall C $= \frac{TP}{(TP+FN)} = \frac{5}{(5+2)} = 0{,}71$

  Recall D $= \frac{TP}{(TP+FN)} = \frac{6}{(6+0)} = 1$

  Recall $= \frac{Recall\ \text{A}+Recall\ \text{B}+Recall\ \text{C}+Recall\ \text{D}}{\text{Jumlah Kelas}}$

  $= \frac{0{,}33+1+0{,}71+1}{4} \times 100\% = 76\%$
- Spesifitas A $= \frac{TN}{(TN+FP)} = \frac{16}{(16+1)} = 0{,}94$

Spesifitas B $= \frac{TN}{(TN+FP)} = \frac{13}{(13+4)} = 0{,}76$

Spesifitas C $= \frac{TN}{(TN+FP)} = \frac{13}{(13+1)} = 0{,}93$

Spesifitas D $= \frac{TN}{(TN+FP)} = \frac{12}{(12+0)} = 1$

$$\text{Spesifitas} = \frac{\text{Spesifitas A+Spesifitas B+Spesifitas C+Spesifitas D}}{\text{Jumlah Kelas}}$$

$$= \frac{0{,}94+0{,}76+0{,}93+1}{4} \times 100\% = 91\%$$

Terdapat pengukuran lain selain yang telah disebutkan di atas yaitu AUC, yang berfungsi sebagai ukuran statistik untuk mengevaluasi kinerja klasifikasi biner, memilih model terbaik, dan menentukan model yang paling efisien. Berbeda dengan metrik *sensitivity* (*recall*) yang hanya mengevaluasi proporsi positif yang benar diidentifikasi, AUC bertujuan untuk mengevaluasi kinerja diskriminatif dengan memperkirakan probabilitas *output* dari sampel yang dipilih secara acak dari populasi positif atau negatif. AUC sering digunakan untuk mengukur kualitas classifier probabilitas, sehingga rentang nilai AUC selalu berkisar antara 0 hingga 1. Semakin tinggi nilai AUC, semakin kuat klasifikasi yang digunakan. Berikut disajikan tabel kriteria penilaian dari AUC berdasarkan nilai akurasinya

Tabel 3.2 Kriteria Penilaian AUC

Nilai AUC	Kategori Klasifikasi
0.91 – 1.00	*Excellent*
0.81 – 0.90	*Good*
0.71 – 0.80	*Fair*
0.61 – 0.70	*Poor*
≤ 0.60	*Failure*

Sumber: (Defiyanti & Jajuli, 2015)

BAB IV
METODOLOGI PENELITIAN

4.1 Populasi Penelitian

Data yang digunakan dalam penelitian ini merupakan data sekunder yang diunduh melalui website NCBI https://www.ncbi.nlm.nih.gov/ pada 10 Februari 2023. Data tersebut merupakan data microarray yang tersedia di server dan memuat informasi bioteknologi seperti DNA, protein, senyawa aktif, dan taksonomi. Data microarray yang digunakan adalah seri GSE76262_RAW yang terdiri dari 139 sampel.

4.2 Tempat dan Waktu Penelitian

Penelitian ini dibuat dan dilakukan di Kabupaten Sleman, DIY, dengan sebagian besarnya dilakukan di Fakultas MIPA, Universitas Islam Indonesia, dimana penelitian ini dilakukan terhitung sejak proses pengambilan data, analisis data, dan penyusunannya dimulai dari tanggal 10 Februari - 03 Maret 2023

4.3 Variabel Penelitian

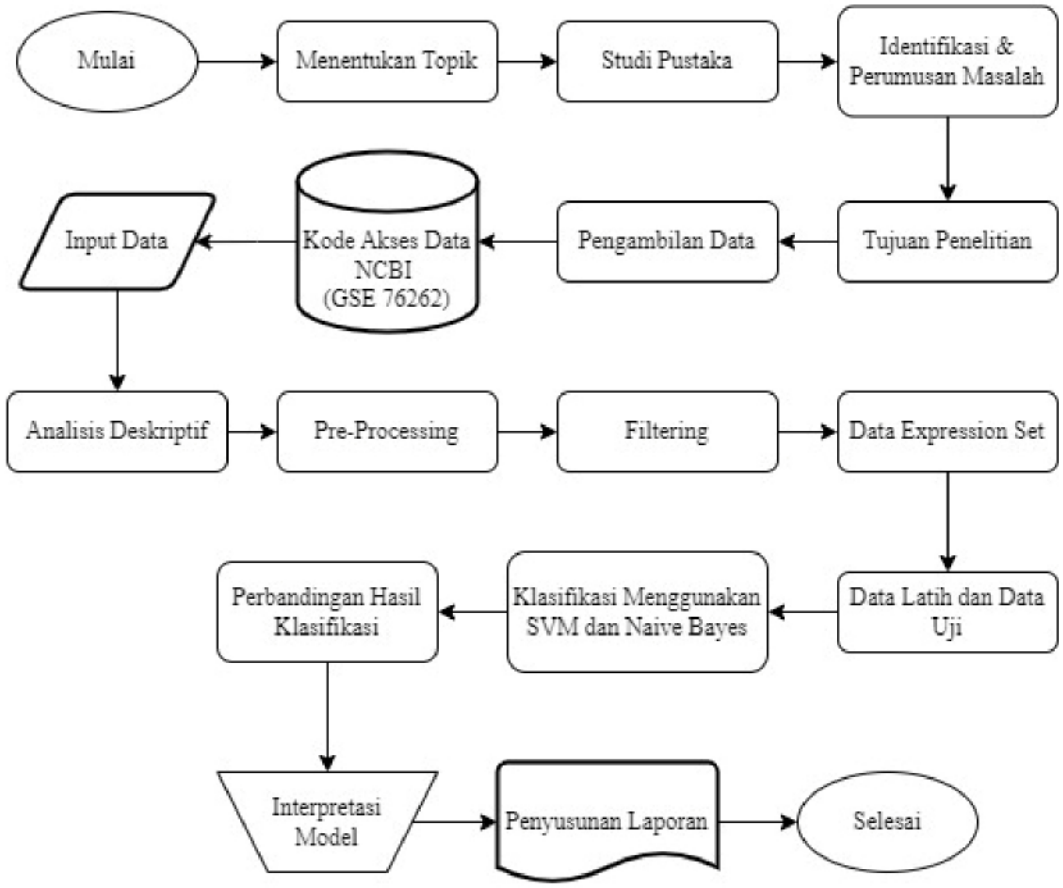
Adapun macam-macam variabel yang digunakan pada penelitian ini adalah sebagai berikut:

Tabel 4.1 Definisi Variabel Penelitian

Nama Variabel	Definisi Operasional Variabel
Gen	Gen yang digunakan pada penelitian ini adalah hasil dari ekspresi gen sputum yang telah diinduksi, pada manusia atau pasien yang menderita penyakit pernafasan atau asma, dengan jumlah gen sebanyak 139 sampel.
Jaringan (*Tissue*)	Variabel yang berupa sekumpulan sel yang menyusun setiap tubuh manusia. variabel ini selalu berbeda pada tiap-tiap dataset yang digunakan

Nama Variabel	Definisi Operasional Variabel
Jenis Kelamin	Variabel yang menjelaskan tentang identitas jenis kelamin pasien yang telah dilakukan induksi sputum pada pasien penyakit asma.
Umur	Variabel usia merupakan usia pasien yang tercatat dalam dataset GSE76262
Kelas Sputum	Variabel yang menjelaskan tentang kategori atau kelas data ekspresi gen dari sel *sputum* (dahak) yang diinduksi pada pasien penyakit asma, dimana terdapat 3 kelas yaitu, penginduksian sputum berat, sedang, dan kontrol sehat.

4.4 Langkah Penelitian



Gambar 4.1 *Flowchart* Tahapan Penelitian

Berdasarkan *flowchat* pada gambar diatas, dapat diketahui bahwa tahapan-tahapan dalam penelitian ini yaitu sebagai berikut:

1. Tahapan pertama yang dilakukan oleh peneliti adalah menentukan topik yang akan dipakai, dalam hal ini peneliti menggunakan topik mengenai bioinformatika.
2. Kemudian dilanjutkan mencari studi pustaka dengan mencari beberapa literasi pada penelitian-penelitian sebelumnya terkait bioinformatika yang kemudian akan dijadikan sebagai referensi dalam penelitian ini.
3. Setelah mendapatkan referensi dari beberapa penelitian-penelitian terdahaulu, kemudian dilakukan identifikasi serta perumusan masalah. Dalam hal ini peneliti mengangkat judul penelitian "Implementasi Metode *Support Vector Machine* dan *Naïve Bayes* Pada Data Ekspresi Gen *Microarray*" dengan studi kasus terkait klasifikasi data ekspresi gen

induced sputum pada pasien penderita penyakit asma, kemudian ditentukanlah tujuan dari penelitian ini.

4. Selanjutnya melakukan pengambilan data pada *website National Center of Biotechnology Information* (NCBI) dan didapatkan data terkait data *microarray* yang memuat informasi bioteknologi seperti DNA, protein, senyawa aktif, dan taksonomi, yang diberi akses seri GSE76262_RAW yang terdiri dari 139 sampel.
5. Tahapan selanjutnya yaitu memasuki tahapan peng-*input*-an data, yang dilakukan dengan bantuan *software Rstudio*, sebelum meng-*input* data, diperlukan meng-*install* beberapa *packages* yang akan digunakan, seperti `affy`, `GEOquery`, `Biobase`, `affyPLM`, `ul33x3pcdf`, `ul33x3p.db`, `genefilter`, `AnnotationDbi`, `BiocManager`, `biomaRT`, `hgul33plus2.db`. setelah itu dilakukan *input* data dengan sintak yang telah dilampirkan pada Lampiran 1.
6. Setelah *input* data, lalu peneliti melakukan analisis deskriptif pada data tersebut, untuk duketahui terkait gambaran umum pada data, dengan menggunakan sintak yang telah dilampirkan pada Lampiran 1.
7. Tahap selanjutnya dilakukan *pre-processing* data guna untuk memastikan kualitas data yang lebih baik dan mencegah adanya kesalahan pada tahap analisis selanjutnya.
8. Setelah dilakukan *pre-processing* data, kemudian dilakukan *filtering* pada data tersebut. Tahap ini dilakukan dengan cara 2 (dua) tahap, yang pertama yaitu tahap *filtering,* kemudian dilanjutkan dengan tahap kedua yaitu tahap *feature selection* yang kemudian didapatkan *data expression set*.
9. Tahap selanjutnya yaitu memasuki metode klasifikasi, langkah awal sebelum dilakukan klasifikasi yaitu dilakukan pembagian data menjadi 2 (dua) yang biasa dinamakan data latih atau data *training* dan data uji atau data *testing*. Pada tehap ini peneliti melakukan pembagian data *training* dan data *testing* dengan perbandingan yaitu sebesar 80%:20%.
10. Selanjutnya dilakukan klasifikasi dengan menggunakan 2 (dua) metode yang nantinya akan dibandingan dengan menggunakan nilai akurasinya.

Kedua metode tersebut yaitu metode *Support Vector Machine* dan metode *Niave Bayes*.

11. Setelah dilakukan klasifikasi dengan 2 (dua) metode tersebut, kemudian dilakukan perbandingan antara kedua metode tersebut untuk didapatkan model terbaik dengan melihat nilai akurasi dan nilai AUC nya. Selanjutnya dilakukan interpretasi hasil dari klasifikasi tersebut.

4.5 Metode Analisis Data

Metode analisis data yang digunakan pada penelitian ini adalah klasifikasi menggunakan 2 metode klasifikasi yaitu *Support Vector Machine* (SVM), dan *Naïve Bayes*. Analisis ini digunakan untuk mengklasifikasikan gen hasil ekspresi *gen sputum severe, sputum moderate*, dan *sputum healthy control*. Software yang digunakan untuk penelitian ini adalah software R studio dengan versi 4.2.2

BAB V
HASIL DAN PEMBAHASAN

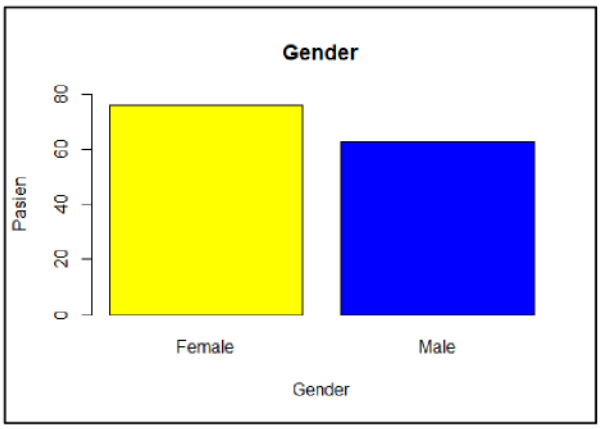
5.1 Analisis Deskriptif

Pada penelitian ini data yang digunakan adalah data ekspresi gen dari penginduksian sputum dalam proyek prediksi hasil penyakit pernafasan manusia atau penyakit asma yang didapatkan pada website NCBI dengan series GSE76262 dan pada *platform* GPL13158.

Tabel 5.1 *Dataset* GSE76262

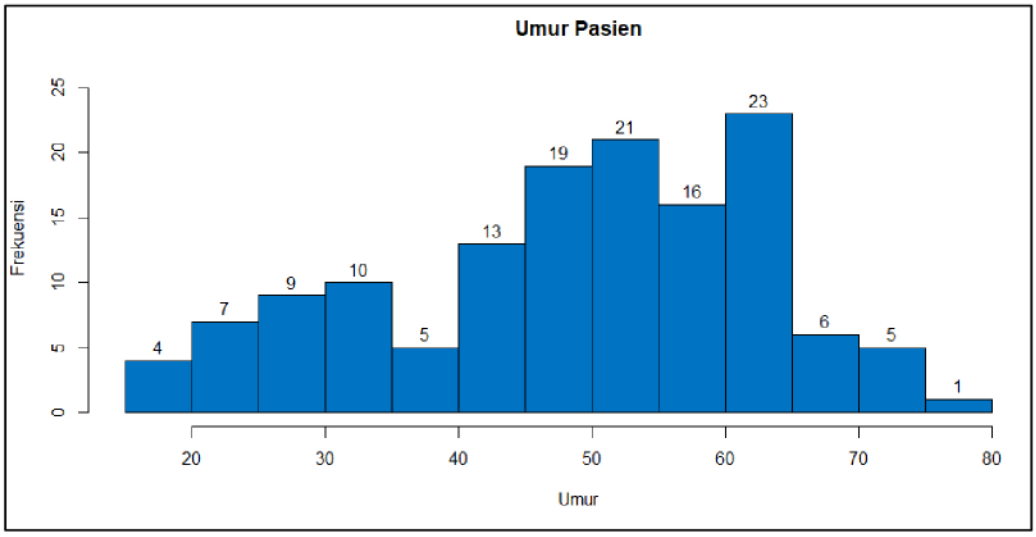
Informasi	Frekuensi
Gen	54715
Sampel	139
Kelas	3

Berdasarkan Tabel 5.1 diketahui bahwa data terdiri atas 139 sampel dengan jumlah gen sebanyak 54715 gen dan terdiri atas 3 kelas yaitu ekspresi gen *sputum severe, sputum moderate, dan sputum healthy control*. Sampel pada data ekspresi gen dengan series GSE76262 diambil dari Department of Computing, Imperial College London. Dari 139 data yang ada, terbagi atas 3 kelas, dengan jumlah sampel untuk *sputum severe* sebanyak 93 pasien, jumlah sampel *sputum moderate* sebanyak 25 pasien, dan sampel *sputum healthy control* sebanyak 21 pasien



Gambar 5.1 *Barplot* Jenis Kelamin

Berdasarkan Gambar 5.1 diketahui bahwa dari 139 sampel sebanyak 76 sampel berjenis kelamin perempuan dan sisanya sebanyak 63 sampel berjenis kelamin laki-laki. Dari 76 sampel perempuan terdapat sebanyak 57 pasien penderita penyakit asma yang termasuk kedalam kategori berat atau *severe asthma*, lalu terdapat 13 pasien yang menderita penyakit asma dengan kategori sedang atau *moderate asthma*, dan 6 sampel atau pasien dengan kategori *healthy control* atau dalam masih termasuk dalam kategori sehat. Sedangkan dari 63 sampel atau pasien yang berjenis kelamin laki-laki, terdapat 36 pasien penyakit asma yang termasuk kedalam kategori berat, 12 sampel termasuk kedalam kategori sedang, dan 15 sisanya termasuk kedalam pasien dengan hasil kontrol yang sehat.

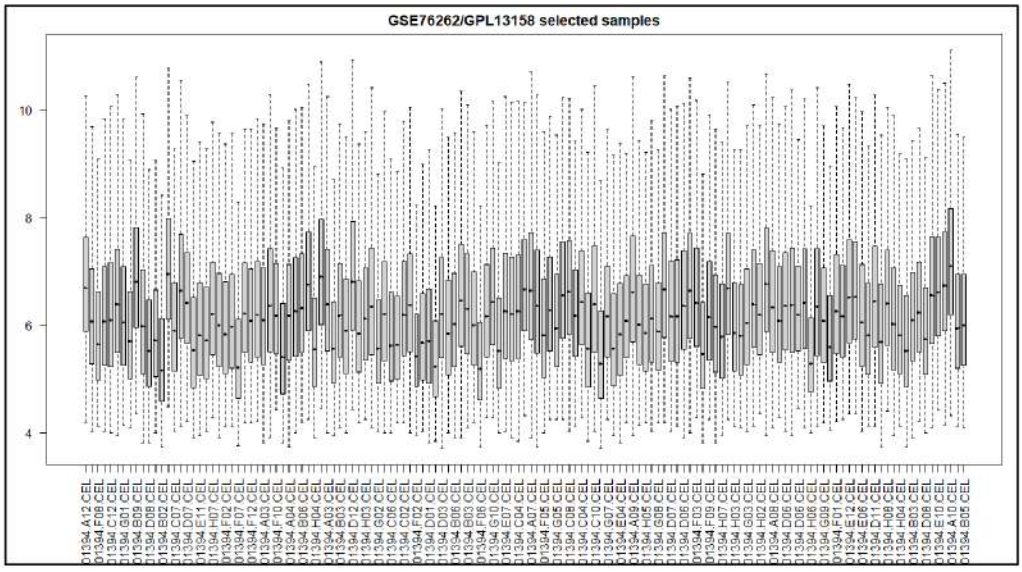


Gambar 5.2 Histogram Umur Pasien

Berdasarkan Gambar 5.2 diketahui bahwa sampel dari data ekspresi gen GSE76262 berada pada rentang usia 15 tahun sampai dengan 80 tahun. Berdasarkan gambar tersebut juga dapat diketahui bahwa setelah hasil pemeriksaan, pasien yang menderita penyakit asma paling banyak berada di rentang umur 40 tahun sampai 60 tahun, dan juga terdapat beberapa yang menderita penyakit asma pada usia lanjut atau usia >70 tahun, hal ini menandakan bahwa penyakit asma lebih cenderung diderita oleh pasien yang telah dewasa dan usia lanjut, sedangkan untuk pasien dengan rentang umur 15-39 tahun jumlah penderita penyakit asma tidak sebanyak dengan pasien yang berusia 40-80 tahun

5.2 Pre-Processing Data

Tahap *pre-processing* ini dilakukan dengan menggunakan bantuan *package affyPLM* sebagai alat untuk meringkas tingkatan pada data ekspresi gen dan menghapus bentuk non-biologis serta *noise* dalam data. Digunakan *boxplot* sebagai grafik untuk mengetahui hasil atau *output* dari *preprocessing* yang dilakukan. Berikut merupakan tampilan *boxplot* sebelum dilakukan *preprocessing*



Gambar 5.3 *Boxplot* sebelum dilakukan *pre-processing*

Berdasarkan *boxplot* diatas diketahui bahwa pada data pengamatan sebelum dilakukan *preprocessing*, sebaran data pada masing-masing variabel tidak seragam. Hal tersebut juga bisa dilihat pada nilai *quartil* ke-2 atau *median* yang nilai untuk masing-masing variabelnya tidak seragam dan garis *median* tidak berada di tengah kotak, yang berarti bahwa sebaran data atau bentuk data yang digunakan tidak normal atau tidak simetris. Selain dari itu juga *output* pada *boxplot* tersebut bisa saja terjadi dikarenakan masih adanya variansi ataupun faktor non biologis pada dataset yang digunakan.

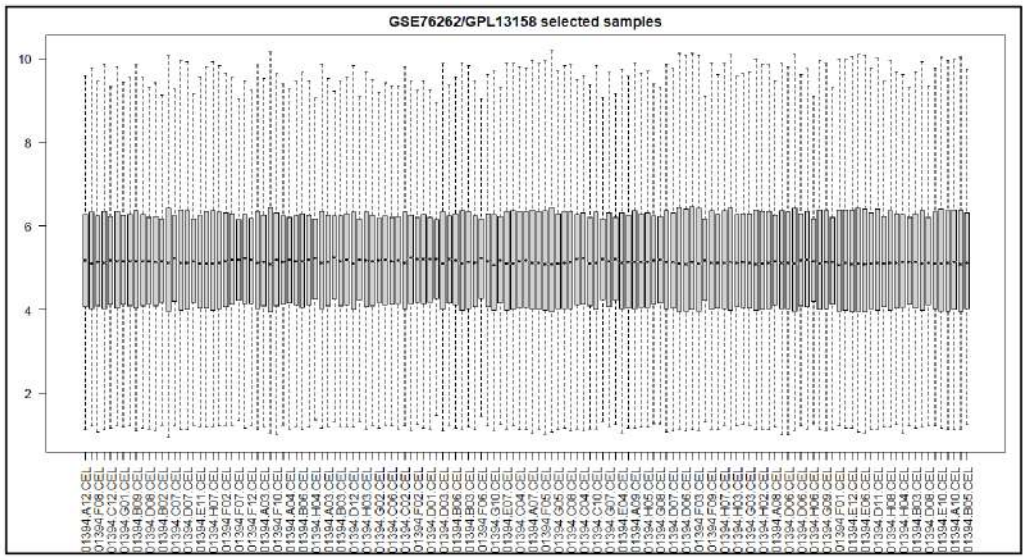
Dikarenakan masih terdapat sebaran data yang tidak normal, dan untuk faktor dan variansi non-biologis yang belum dihilangkan pada data, maka dilakukan tahap *preprocessing* ini dengan menggunakan fungsi `threestep()` yang diambil dari *package affyPLM*, fungsi tersebut merupakan satu metode *pre-processing* yang

terdiri dari tiga langkah yaitu penyesuaian *background correction*, *normalization*, dan *summarization* data.

- *Background correction* merupakan tahap pertama dari fungsi *three-step* pada *preprocessing,* dengan menggunakan metode RMA2. RMA merupakan singkatan dari *Robust multi-array average* yang bertujuan untuk menghilangkan kebisingan dan hibridisasi non-spesifik pada data *microarray,* serta bertujuan untuk mengetahui intensitas distribusi antar *probe*.
- *Normalization*, tahap selanjutnya dari *preprocessing* ialah *normalization* dengan menggunakan metode *quantile*, tahap ini dilakukan sebagai upaya untuk menyetarakan data dengan menghilangkan variansi non biologis yang keberadaannya tidak memiliki keterkaitan dengan proses biologis yang sedang diteliti, dimana variansi non biologis ini sering muncul disebabkan oleh beberapa faktor, seperti proses pengambilan sampel yang dilakukan dalam beberapa tahap sehingga bisa saja terdapat perbedaan perlakuan ataupun perbedaan waktu dan suhu pada masing-masing sampel, sehingga hal-hal tersebut bisa jadi memiliki potensi sebagai penyebab munculnya faktor-faktor non-biologis pada data yang ingin diteliti.
- *Summarization*, pada tahap ini dilakukan *summarization* dengan menggunakan metode *median polish* yang merupakan proses penggabungan beberapa intensitas *probe* sehingga menjadi *probe set*, yang kemudian *probe set* tersebut akan menghasilkan nilai ekspresi gen.

Setelah dilakukan tahap *pre-processing* maka selanjutnya akan ditampilkan kembali bentuk *boxplot* dari hasil data yang telah dilakukan *pre-processing*. Sebelum menampilkan hasil dengan menggunakan *boxplot,* data ekspresi gen yang diperoleh dalam tahap *pre-processing* masih berupa data yang berbentuk *expression set*, sehingga untuk dapat menampilkan hasilnya dalam bentuk *boxplot* data tersebut perlu diubah terlebih dahulu dari bentuk *expression set* menjadi bentuk *data frame*. Berikut ditampilkan grafik yang menunjukkan perbedaan hasil dalam bentuk *boxplot* untuk data *microarray* yang sebelum dilakukan

preprocessing dan data yang telah dilakukan *preprocessing* dengan bentuk data yang telah diubah dari sebelumnya *data expression set* menjadi bentuk *data frame*.



Gambar 5.4 *Boxplot* setelah dilakukan *pre-processing*

Dalam Gambar 5.3 dan Gambar 5.4, terlihat perbedaan yang sangat signifikan antara data ekspresi gen *induced sputum* pada pasien penyakit asma, sebelum dan setelah tahap *pre-processing* menggunakan metode *three steps*. Tapahan *pre-processing* ini digunakan untuk menghapus nilai non biologis, sehingga yang tertera pada *boxplot* setelah dilakukan *pre-processing* hanya data-data yang bersifat biologis saja. Perbedaan hasil pada kedua *boxplot* ini disebabkan oleh metode normalisasi yang digunakan pada tahap *pre-processing*, yang dapat membuat data menjadi lebih normal dan nilai *median* untuk masing-masing variabel pada *boxplot* menjadi seragam dibandingkan dengan yang sebelum dilakukan *pre-processing*.

5.3 Filtering

Pada proses *filtering*, langkah awal adalah menentukan gen yang akan digunakan dalam analisis. Terdapat dua tahap dalam proses *filtering*, yaitu *filtering* dan feature selection. Untuk tahap *filtering* data, digunakan fungsi `nsfilter` (*Non-Specified Filtering*) untuk menghilangkan variabel yang memiliki nilai *interquartile range* (IQR) yang tinggi sehingga dihasilkan data dengan dimensi 10379x139.

Setelah proses *filtering*, dilakukan tahap *feature selection* menggunakan fungsi *multtest*. Fungsi ini awalnya berasal dari *ttest* yang digunakan untuk membandingkan dua sampel dari populasi yang memiliki variansi yang sama. Karena data yang digunakan dalam penelitian ini terdiri dari tiga sampel kelas, maka digunakan fungsi *F-test*. Pada tahap *multtest*, diasumsikan bahwa klasifikasi data terdistribusi secara normal, yang dapat dikonfirmasi dengan perintah `qqnorm`. Hasil dari proses *feature selection* ini menghasilkan data dengan dimensi 139x3330, seperti yang terlihat pada tabel.

Tabel 5.2 Sampel dan Gen Setelah dilakukan *Filtering dan Feature Selection*

	Nsfilter	Multtest
Sampel	139	139
Gen	10379	3330

5.4 Membagi Data Kedalam Bentuk *Data Train, & Data Test*

Dari tahap filtering didapatkan data yang siap dianalisis berdimensi 139x3330, dari data tersebut kemudian dipecah menjadi dua bagian, dengan rasio untuk data *training* (latih) sebesar 80%, dan data *testing* (uji) sebesar 20% dari total data yang siap dianalisis untuk masing-masing kelas atau kategori, yang kemudian data yang telah dibagi tersebut akan digunakan untuk melakukan analisis dengan menggunakan beberapa metode klasifikasi (*Support Vector Machine* & *Naïve Bayes*). Hasil yang diperoleh dari pembagian data menjadi data *train* dan data *test* tersebut dapat dilihat pada tabel dibawah ini

Tabel 5.3 Pembagian Data Training dan Data Testing

Klasifikasi	Jumlah	Data *Training* 80%	Data *Testing* 20%
Severe	93	74	19
Moderate	25	20	5
Healthy Control	21	17	4
Total	**139**	**111**	**28**

Berdasarkan tabel 5.3 diatas, didapatkan perbandingan data *training* dan data *testing* sebesar 80%:20%, yaitu dari total 139 data yang diperoleh dan dapat

dianalisis, digunakan sebanyak 111 data sebagai data *training* dan 28 data sisanya digunakan untuk data *testing*.

5.5 Support Vector Machine

Setelah selesai melewati tahap-tahap yang telah dijelaskan sebelumnya seperti, *pre-processing, filtering,* dan pembagian data, sehingga didapatkan data yang telah siap digunakan dan dapat dilakukan analisis klasifikasi, di mana akan dilakukan analisis klasifikasi yang pertama dengan menggunakan metode *support vector machine* atau SVM. Pada analisis klasifikasi menggunakan metode SVM terdapat empat kernel yang dapat digunakan untuk mengklasifikasikan data yaitu *kernel linear, Polynomial, Sigmoid dan RBF*. Masing-masing kernel mempunyai parameter yang berbeda dalam setiap pembuatan modelnya. Berikut ini adalah hasil nilai akurasi dari masing-masing kernel pada percobaan klasifikasi menggunakan data ekspresi gen *induced sputum severe, sputum moderate, dan sputum healthy control*.

5.5.1 Kernel *Linear*

Dilakukan *tune parameter* untuk memperoleh parameter terbaik yang akan digunakan dalam melakukan klasifikasi menggunakan *support vector machine*. Parameter yang digunakan pada SVM kernel *linear* yaitu *cost,* dengan nilai *cost* berkisar antara (0.01, 0.01, 0.1, 1, 10, 100, 200, 300)

Tabel 5.4 *Tune parameter* kernel *linear*

No	*Cost*	*Error*
1	0.001	0.2363636
2	0.01	0.2363636
3	0.1	0.2363636
4	1	0.2363636
5	10	0.2363636
6	100	0.2363636
7	200	0.2363636
8	300	0.2363636

Berdasarkan hasil *tune parameter* diatas, diketahui parameter yang paling baik dari SVM kernel *polynomial* adalah *Cost* = 0.1 dengan nilai *error* = 0.236363. Dilakukan pembentukan model SVM kernel *linear* dengan data *training* menggunakan parameter terbaik dari hasil *tune parameter*. Berikut adalah parameter model SVM kernel *linear*:

Tabel 5.5 Parameter SVM kernel *linear*

Parameter		
SVM-Type	:	C-*Classification*
SVM-Kernel	:	*Linear*
Cost	:	0.1
Number of Support Vector	:	89

Selanjutnya digunakan parameter terbaik tersebut dalam membuat *confusion matrix* dari model SVM kernel *linear* untuk mendapatkan nilai akurasi klasifikasi dengan menggunakan data *testting*.

Tabel 5.6 *Confusion matrix* kernel *linear*

Prediksi	Aktual		
	Severe	Moderate	Healthy Control
Severe	18	3	0
Moderate	1	2	2
Healthy Control	0	0	2

Berdasarkan tabel 5.6 diatas, terdapat 28 sampel yang didapatkan dari hasil pembagian data *testting* sebesar 20% pada dataset dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 18 sampel sesuai dengan kategori aslinya(true positive), lalu 3 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe*, dan 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe* (false positive).

- Pada kategori *moderate*, model memprediksi sebanyak 0 sampel di mana model memprediksi kategori *severe* tetapi kategori sebenarnya *moderate*, 2 sampel di mana model memprediksi dengan benar kategori *moderate* dan kategori sebenarnya juga *moderate* (true positive), dan 2 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate* (false negative).
- Pada kategori *healthy control*, terdapat 0 kasus di mana model memprediksi kategori *severe* tetapi kategori sebenarnya adalah *healthy control* (false positive), 0 kasus di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *healthy control* (false positive), dan 1 kasus di mana model memprediksi dengan tepat kategori *healthy control* dan kategori sebenarnya juga *healthy control* (true positive)

Berdasarkan hasil dari penjelasan *confusion matrix* tersebut, dapat dilakukan perhitungan untuk mengetahui nilai akurasi dari klasifikasi SVM kernel *linear*, dengan perhitungan sebagai berikut:

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{22}{28} = 0.7857 \times 100\%\ = 78.57\%$$

5.5.2 Kernel *Polynomial*

Dilakukan *tune parameter* untuk memperoleh parameter terbaik dari SVM kernel *polynomial* yang akan digunakan untuk melakukan klasifikasi. Parameter yang digunakan yaitu *cost* dan *gamma,* dengan nilai *cost* berkisar antara (0.1, 1, 10, 100, 200, 300) dan *gamma* sebesar (0.1, 1, 2, 3, 4, 5). Sehingga diperoleh hasil keseluruhan dari *tune parameter* sebagai berikut.

Tabel 5.7 *Tune parameter* kernel polynomial

No	*Cost*	*Gamma*	*Error*	No	*Cost*	*Gamma*	*Error*
1	0.1	0.1	0.2772727	19	0.1	3	0.2772727
2	1	0.1	0.2772727	20	1	3	0.2772727
3	10	0.1	0.2772727	21	10	3	0.2772727
4	100	0.1	0.2772727	22	100	3	0.2772727

5	200	0.1	0.2772727	23	200	3	0.2772727
6	300	0.1	0.2772727	24	300	3	0.2772727
7	0.1	1	0.2772727	25	0.1	4	0.2772727
8	1	1	0.2772727	26	1	4	0.2772727
9	10	1	0.2772727	27	10	4	0.2772727
10	100	1	0.2772727	28	100	4	0.2772727
11	200	1	0.2772727	29	200	4	0.2772727
12	300	1	0.2772727	30	300	4	0.2772727
13	0.1	2	0.2772727	31	0.1	5	0.2772727
14	1	2	0.2772727	32	1	5	0.2772727
15	10	2	0.2772727	33	10	5	0.2772727
16	100	2	0.2772727	34	100	5	0.2772727
17	200	2	0.2772727	35	200	5	0.2772727
18	300	2	0.2772727	36	300	5	0.2772727

Berdasarkan hasil *tune parameter* diatas, diketahui parameter yang paling baik dari SVM kernel *polynomial* adalah *Cost*=10 dan *gamma*=1. Dilakukan pembentukan model SVM kernel *polynomial* dengan data *training* menggunakan *cost* dan *gamma* terbaik dari hasil *tune parameter*. Berikut adalah parameter model SVM kernel *polynomial*:

Tabel 5.8 Parameter model SVM kernel *polynomial*

Parameter		
SVM-Type	:	C-Classification
SVM-Kernel	:	Polynomial
Cost	:	10
Gamma	:	0.1
Number of Support Vector	:	93

Selanjutnya digunakan parameter terbaik tersebut dalam membuat *confusion matrix* dari model SVM kernel *polynomial* untuk mendapatkan nilai akurasi klasifikasi dengan menggunakan data *testting*. Sehingga diperoleh *confusion matrix* nya sebagai berikut.

Tabel 5.9 *Confusion matrix* SVM kernel *polynomial*

Prediksi	Aktual

	Severe	Moderate	Healthy Control
Severe	19	3	1
Moderate	0	2	2
Healthy Control	0	0	1

Berdasarkan tabel 5.9 diatas, terdapat 28 sampel yang didapatkan dari hasil pembagian data testing sebesar 20% pada dataset dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 19 sampel sesuai dengan kategori aslinya, lalu 3 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe*, dan 1 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe*.
- Pada kategori *moderate*, model memprediksi sebanyak 0 sampel di mana model memprediksi kategori *severe* tetapi kategori sebenarnya *moderate*, 2 sampel di mana model memprediksi dengan benar kategori *moderate* dan kategori sebenarnya juga *moderate* (true positive), dan 2 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate* (false negative).
- Pada kategori *healthy control*, terdapat 0 kasus di mana model memprediksi kategori *severe* tetapi kategori sebenarnya adalah *healthy control* (false positive), 0 kasus di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *healthy control* (false positive), dan 1 kasus di mana model memprediksi dengan tepat kategori *healthy control* dan kategori sebenarnya juga *healthy control* (true positive)

Berdasarkan hasil dari penjelasan *confusion matrix* tersebut, dapat dilakukan perhitungan untuk mengetahui nilai akurasi dari klasifikasi SVM kernel *polynomial*, dengan perhitungan sebagai berikut

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{22}{28} = 0.7857 \times 100\%\ = 78.57\%$$

5.5.3 Kernel Sigmoid

Dilakukan *tune parameter* untuk memperoleh parameter terbaik dari SVM kernel *sigmoid* yang akan digunakan untuk melakukan klasifikasi. Parameter yang digunakan yaitu *cost* dan *gamma,* dengan nilai *cost* berkisar antara (0.1, 1, 10, 100, 200, 300) dan *gamma* sebesar (0.1, 1, 2, 3, 4, 5). Sehingga diperoleh hasil keseluruhan dari *tune parameter* sebagai berikut.

Tabel 5.10 *Tune parameter* kernel *sigmoid*

No	*Cost*	*Gamma*	*Error*	No	*Cost*	*Gamma*	*Error*
1	0.1	0.1	0.3310606	19	0.1	3	0.3310606
2	1	0.1	0.3492424	20	1	3	0.3219697
3	10	0.1	0.3227273	21	10	3	0.3128788
4	100	0.1	0.3227273	22	100	3	0.3128788
5	200	0.1	0.3318182	23	200	3	0.3128788
6	300	0.1	0.3318182	24	300	3	0.3128788
7	0.1	1	0.3310606	25	0.1	4	0.3310606
8	1	1	0.3219697	26	1	4	0.3219697
9	10	1	0.3227273	27	10	4	0.3128788
10	100	1	0.3318182	28	100	4	0.3128788
11	200	1	0.3318182	29	200	4	0.3128788
12	300	1	0.3318182	30	300	4	0.3128788
13	0.1	2	0.3310606	31	0.1	5	0.3310606
14	1	2	0.3310606	32	1	5	0.3219697
15	10	2	0.3310606	33	10	5	0.3128788
16	100	2	0.3310606	34	100	5	0.3128788
17	200	2	0.3227273	35	200	5	0.3128788
18	300	2	0.3227273	36	300	5	0.3128788

Berdasarkan hasil *tune parameter* diatas, diketahui parameter yang paling baik dari SVM kernel *sigmoid* adalah *Cost*=10 dan *gamma*=3. Selanjutnya dilakukan pembentukan model SVM kernel *sigmoid* dengan data *training* menggunakan *cost* dan *gamma* terbaik dari hasil *tune parameter*. Berikut adalah parameter model untuk SVM kernel *sigmoid*:

Tabel 5.11 *parameter* SVM kernel *sigmoid*

Parameter		
SVM-*Type*	:	*C-Classification*
SVM-Kernel	:	*Sigmoid*
Cost	:	10
Gamma	:	3
Number of Support Vector	:	60

Selanjutnya digunakan parameter terbaik tersebut dalam membuat *confusion matrix* dari model SVM kernel *sigmoid* untuk mendapatkan nilai akurasi klasifikasi dengan menggunakan data *testting*. Sehingga diperoleh *confusion matrix* sebagai berikut.

Tabel 5.12 *Confusion matrix* SVM kernel *sigmoid*

Prediksi	Aktual		
	Severe	Moderate	Healthy Control
Severe	16	5	4
Moderate	2	0	0
Healthy Control	1	0	0

Berdasarkan tabel 5.12 diatas, terdapat 28 sampel yang didapatkan dari hasil pembagian data testing sebesar 20% pada *dataset*, dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 16 sampel sesuai dengan kategori aslinya (true positive), lalu 5 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe*, dan 4 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe*.
- Pada kategori *moderate*, model memprediksi sebanyak 2 sampel kategori *severe* dan kategori sebenarnya *moderate*, 0 sampel di mana model memprediksi dengan benar kategori *moderate* dan kategori sebenarnya juga *moderate*, dan 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate* (false positive).

- Pada kategori *healthy control*, terdapat 1 kasus di mana model memprediksi kategori *severe* tetapi kategori sebenarnya adalah *healthy control* (false negative), 0 kasus di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *healthy control* (false positive), dan 0 kasus di mana model memprediksi dengan tepat kategori *healthy control* dan kategori sebenarnya juga *healthy control*

Berdasarkan hasil dari penjelasan *confusion matrix* tersebut, dapat dilakukan perhitungan untuk mengetahui nilai akurasi dari klasifikasi SVM kernel *sigmoid*, dengan menggunakan perhitungan sebagai berikut

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{16}{28} = 0.5714 \times 100\%\ = 57.14\%$$

5.5.4 Kernel Radial

Dilakukan *tune parameter* untuk memperoleh parameter terbaik dari SVM kernel *Radial basis function* (RBF) yang akan digunakan untuk melakukan klasifikasi. Parameter yang digunakan yaitu *cost* dan *gamma,* dengan nilai *cost* berkisar antara (0.1, 1, 10, 100, 200, 300) dan *gamma* sebesar (0.1, 1, 2, 3, 4, 5). Sehingga diperoleh hasil keseluruhan dari *tune parameter* sebagai berikut.

Tabel 5.13 *Tune parameter* SVM kernel RBF

No	*Cost*	*Gamma*	*Error*	No	*Cost*	*Gamma*	*Error*
1	0.1	0.1	0.2772727	19	0.1	3	0.2772727
2	1	0.1	0.2772727	20	1	3	0.2772727
3	10	0.1	0.2772727	21	10	3	0.2772727
4	100	0.1	0.2772727	22	100	3	0.2772727
5	200	0.1	0.2772727	23	200	3	0.2772727
6	300	0.1	0.2772727	24	300	3	0.2772727
7	0.1	1	0.2772727	25	0.1	4	0.2772727
8	1	1	0.2772727	26	1	4	0.2772727
9	10	1	0.2772727	27	10	4	0.2772727
10	100	1	0.2772727	28	100	4	0.2772727
11	200	1	0.2772727	29	200	4	0.2772727

No	*Cost*	*Gamma*	*Error*	No	*Cost*	*Gamma*	*Error*
12	300	1	0.2772727	30	300	4	0.2772727
13	0.1	2	0.2772727	31	0.1	5	0.2772727
14	1	2	0.2772727	32	1	5	0.2772727
15	10	2	0.2772727	33	10	5	0.2772727
16	100	2	0.2772727	34	100	5	0.2772727
17	200	2	0.2772727	35	200	5	0.2772727
18	300	2	0.2772727	36	300	5	0.2772727

Berdasarkan hasil *tune parameter* diatas, diketahui parameter yang paling baik dari SVM kernel *RBF* adalah *Cost*=10 dan *gamma*=1. Dilakukan pembentukan model SVM kernel *RBF* dengan data *training* menggunakan *cost* dan *gamma* terbaik dari hasil *tune parameter*. Berikut adalah parameter model SVM kernel *RBF*:

Tabel 5.14 Parameter SVM kernel RBF

Parameter		
SVM-Type	:	*C-Classification*
SVM-Kernel	:	*Radial*
Cost	:	0.1
Gamma	:	1
Number of Support Vector	:	111

Selanjutnya digunakan parameter terbaik tersebut dalam membuat *confusion matrix* dari model SVM kernel *radial basis function* (RBF) untuk mendapatkan nilai akurasi klasifikasi dengan menggunakan data *testting*. Sehingga diperoleh *confusion matrix* nya sebagai berikut.

Tabel 5.15 *Confusion matrix* kernel RBF

Prediksi	Aktual		
	Severe	Moderate	Healthy Control
Severe	19	5	4
Moderate	2	0	0
Healthy Control	1	0	0

Berdasarkan tabel 5.9 diatas, terdapat 28 sampel yang didapatkan dari hasil pembagian data testing sebesar 20% pada *dataset*, dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 19 sampel sesuai dengan kategori aslinya (true positive), lalu 5 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe*, dan 4 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe*.
- Pada kategori *moderate*, model memprediksi sebanyak 0 sampel kategori *severe* dan kategori sebenarnya *moderate*, 0 sampel di mana model memprediksi kategori *moderate* dan kategori sebenarnya juga *moderate*, lalu 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate*.
- Pada kategori *healthy control*, model memprediksi sebanyak 0 sampel kategori *severe* tetapi kategori sebenarnya *moderate*, 0 sampel di mana model memprediksi kategori *moderate* dan kategori sebenarnya juga *moderate*, lalu 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate*.

Berdasarkan hasil dari penjelasan *confusion matrix* tersebut, dapat dilakukan perhitungan untuk mengetahui nilai akurasi dari klasifikasi SVM kernel RBF, dengan menggunakan perhitungan sebagai berikut

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{19}{28} = 0.6786 \times 100\%\ = 67.86\%$$

5.5.5 Perbandingan hasil masing-masing kernel SVM

Setelah diperoleh hasil akurasi klasifikasi untuk masing-masing kernel pada SVM, yaitu kernel *linear, polynomial, sigmoid*, dan RBF, maka akan dilakukan perbandingan hasil dari masing-masing kernel tersebut untuk mengetahui kernel pada SVM yang memiliki nilai akurasi klasifikasi terbaik. Berikut disajikan nilai akurasi untuk masing-masing kernel SVM

Tabel 5.16 Perbandingan Nilai Akurasi Pada masing-masing Kernel

Kernel	Akurasi
Linear	78.57%
Polynomial	78.57%
Sigmoid	57.14%
RBF	67.86%

Berdasarkan Tabel 5.16 diketahui bahwa model SVM dengan kernel linear dan polynomial merupakan 2 model dengan nilai akurasi yang terbaik dibandingkan model SVM dengan kernel yang lain seperti Sigmoid dan RBF, di mana diperoleh nilai tingkat akurasi terbaik sebesar 78,57% untuk metode Kernel Linear, dan Polynomial

Setelah diperoleh nilai akurasi terbaik dari masing-masing kernel pada klasifikasi menggunakan SVM, dan didapatkan kernel linear dan polynomial yang merupakan kernel dengan nilai akurasi terbaik yaitu sebesar 78.57%, maka selanjutnya akan dilakukan analisis klasifikasi dengan menggunakan metode *Naïve Bayes*.

5.6 Naive Bayes

Setelah didapatkan hasil dari analisis klasifikasi dengan menggunakan metode *support vector machine*, selanjutnya akan ditampilkan hasil analisis klasifikasi dengan menggunakan metode *naïve bayes*. Analisis *naïve bayes classifier* (NBC) menggunakan nilai peluang bersyarat dalam menentukan kelasnya, oleh karena itu, klasifikasi pada analisis NBC sebelumnya harus memiliki nilai *prior probability*.

Prior probability adalah komponen utama pada konsep Naive Bayes, *Prior probability* merupakan tahapan untuk mencari nilai probabilitas pada masing-masing pengamatan yang akan menghasilkan klasifikasi, juga dapat dikatakan bahwa nilai prior adalah nilai suatu peluang kejadian. Nilai tersebut merupakan nilai proporsi probabilitas masing-masing kelas pada data. Nilai *prior probability* didapatkan dari banyaknya jumlah kategori kelas positif yang dibagi dengan total

dari data *training* yaitu sebanyak 111 data. Nilai *prior probability* untuk masing-masing kelas atau kategorinya dapat dilihat pada tabel 5.5 dibawah ini.

Tabel 5.17 Nilai *Prior Probability* Masing-Masing Kelas

Nilai	Kelas		
	Sputum_Severe	*Sputum_Moderate*	*Sputum Healthy Control*
Prior Probability	$\frac{70}{111} = 0{,}6667$	$\frac{20}{111} = 0{,}1802$	$\frac{17}{111} = 0{,}1531$

Berdasarkan tabel 5.5 diatas, didapatkan nilai *prior probability* masing-masing kelas atau kategori pada pasien penyakit asma, yaitu kategori *sputum severe* (berat) sebesar 66.67%, *sputum moderate* (sedang) sebesar 18.02%, dan *sputum healthy control* (kontrol yang sehat) sebesar 15.31%. Berdasarkan perhitungan probabilitas masing-masing kelas tersebut dapat dilihat bahwa kategori *sputum severe* atau pasien yang menderita penyakit asma dengan kategori dahak berat mempunyai probabilitas paling tinggi dibanding kelas lainnya dengan nilai *prior probability* sebesar 66.67%.

Setelah itu akan ditampilkan *confusion matrix* untuk hasil prediksi klasifikasi dari *Naive Bayes* menggunakan keseluruhan data *testing* yang merupakan hasil pembagian dari dataset sebesar 20%. Berikut merupakan tampilan *confusion matrix* tersebut

Tabel 5.18 *Confusion Matrix Data Testting Naïve Bayes Classifier*

Prediksi	Aktual			
	Severe	Moderate	Healthy Control	Total
Severe	16	2	0	18
Moderate	1	1	1	3
Healthy Control	2	2	3	7
Total	19	5	4	28

Berdasarkan tabel 5.6 diatas dapat diperoleh nilai *posterior probability* yang merupakan probabilitas suatu hipotesis berdasarkan hasil data baru yang telah

diprediksi. Berikut merupakan hasil dari *posterior probability* untuk masing-masing kelas, dengan menggunakan perhitungan sebagai berikut.

- *Posterior Severe*= $\frac{16}{19} \times \frac{19}{28} : \frac{18}{28} = 0.8889$

  $\frac{2}{5} \times \frac{5}{28} : \frac{18}{28} = 0.1111$

  $\frac{0}{19} \times \frac{4}{28} : \frac{18}{28} = 0$

Berdasarkan hasil dari nilai *posterior probability* untuk kelas severe, dapat diketahui bahwa model *naïve* bayes dapat memprediksi dengan benar model kelas *severe* sesuai dengan kelasnya yaitu sebesar 0.8889, lalu memprediksi model kelas *moderate* tetapi sebenarnya kelas *severe* sebesar 0.1111, dan tidak terdapat kesalahan prediksi dari kelas *severe* terhadap kelas *healthy control.*

- *Posterior Moderate*= $\frac{1}{19} \times \frac{19}{28} : \frac{3}{28} = 0.3333$

  $\frac{1}{5} \times \frac{5}{28} : \frac{3}{28} = 0.3334$

  $\frac{1}{4} \times \frac{4}{28} : \frac{3}{28} = 0.3333$

Berdasarkan hasil dari *posterior probability* untuk kelas *moderate* diperoleh hasil klasifikasi prediksi dari *naïve bayes*, dimana model memprediksi kelas *severe* dengan kelas aslinya *moderate* sebesar 0.3333, lalu model dapat memprediksi kelas *moderate* sesuai dengan kelas aslinya yaitu sebesar 0.3334, dan memprediksi kelas *healthy control* tetapi kelas sebenarnya *moderate* sebesar 0.3333

- *Healthy control:* $\frac{2}{19} \times \frac{19}{28} : \frac{7}{28} = 0.2857$

  $\frac{2}{5} \times \frac{5}{28} : \frac{7}{28} = 0.2857$

  $\frac{3}{4} \times \frac{4}{28} : \frac{7}{28} = 0.4286$

Berdasarkan hasil dari nilai *posterior probability* untuk kelas *healthy control*, diketahui hasil bahwa model memprediksi kelas *severe* sementara kelas aslinya *healthy control* ialah sebesar 0.2857, lalu memprediksi hasil untuk kelas *moderate* sementara kelas aslinya *healthy control* sebesar 0.2857, dan memprediksi dengan benar kelas *healthy control* sesuai dengan kelas aslinya yaitu sebesar 0.4286

Selanjutnya akan dilakukan interpretasi hasil dari *confusion matrix* pada tabel 5.6 diatas, terdapat 28 data yang didapatkan dari hasil pembagian data testing sebesar 20%, dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 16 sampel sesuai dengan kategori aslinya, lalu 2 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe* (false positive), dan tidak ada atau 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe* (true negative).
- Pada kategori *moderate*, model memprediksi sebanyak 1 sampel di mana model memprediksi kategori *severe* tetapi kategori sebenarnya *moderate* (false negative), 1 sampel di mana model memprediksi dengan benar kategori *moderate* dan kategori sebenarnya juga *moderate* (true positive), dan 1 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate* (false negative).
- Pada kategori *healthy control*, terdapat 2 kasus di mana model memprediksi kategori *severe* tetapi kategori sebenarnya adalah *healthy control* (false negative), 2 kasus di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *healthy control* (false negative), dan 3 kasus di mana model memprediksi dengan tepat kategori *healthy control* dan kategori sebenarnya juga *healthy control* (true positive)

Berdasarkan pada penjelasan diatas, maka dapat diketahui hasil dari perhitungan kinerja klasifikasi dari model *naïve bayes* dengan melihat nilai akurasi dari model, yang merupakan parameter seberapa jauh model dapat

mengklasifikasikan data dengan benar sesuai dengan data aktualnya, menggunakan perhitungan sebagai berikut:

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{20}{28} = 0.7143 \times 100\%\ = 71.43\%$$

Selain dari nilai akurasi tersebut, juga diperoleh nilai kinerja klasifikasi lainnya, yang bisa dilihat pada tabel dibawah ini

Tabel 5.19 Nilai Kinerja Klasifikasi *Naïve Bayes*

Akurasi	71,43%
Recall	59.73%
Specificity	84.14%

Berdasarkan tabel 5.19, analisis klasifiasi dengan menggunakan metode *naïve bayes*, diperoleh hasil akurasi, yang merupakan kedekatan nilai prediksi dengan data aktual, yaitu sebesar 71.43%, dan juga diperoleh nilai metrik klasifikaksi lainnya seperti, *recall* atau *sensitivity* sebesar 59.73%, dan *specificity* sebesar 84.14%.

5.7 Klasifikasi dengan SMOTE

Metode klasifikasi dengan menggunakan SVM dan *Naïve Bayes* mampu menghasilkan nilai akurasi klasifikasi yang cukup baik untuk masing-masing metode tersebut, akan tetapi pada data yang digunakan merupakan data yang termasuk dalam unsur *imbalanced*, dikarenakan ada perbedaan jumlah sampel yang cukup jauh antara kelas yang satu dengan kelas yang lainnya, oleh karena itu diperlukan *resampling* data dengan menggunakan teknik *Synthetic Minority Over-Sampling Technique* (SMOTE) untuk mendapatkan jumlah sampel yang seimbang antar kelas pada data, dan hasil prediksi yang lebih baik dibanding pada data yang tidak seimbang atau *imbalanced*. Berikut ditampilkan ulang dataset yang digunakan dalam penelitian.

Tabel 5.20 Data *Training* dan Data *Testting*

Klasifikasi	Jumlah	Data *Training* 80%	Data *Testing* 20%
Severe	93	74	19

Klasifikasi	Jumlah	Data *Training* 80%	Data *Testing* 20%
Moderate	25	20	5
Healthy Control	21	17	4
Total	**139**	**111**	**28**

Synthetic Minority Over-Sampling Technique (SMOTE) merupakan teknik *resampling* dengan cara *oversampling* terhadap kelas minoritas, SMOTE adalah algoritma yang melakukan augmentasi data dengan membuat titik data sintetik berdasarkan titik data asli. SMOTE dapat dilihat sebagai versi lanjutan dari *oversampling*, atau sebagai algoritme spesifik untuk augmentasi data. Sehingga setelah dilakukan teknik SMOTE pada data *imbalanced*, diperoleh suatu ringkasan jumlah data yang dihasilkan dengan teknik SMOTE seperti berikut:

Tabel 5.21 Jumlah data menggunakan SMOTE

Klasifikasi	Jumlah	Data *Training* 80%	Data *Testing* 20%
Severe	93	74	19
Moderate	94	75	19
Healthy Control	94	75	19
Total	**281**	**224**	**57**

Diperoleh data baru dengan jumlah sampel yang relatif sama setelah dilakukan *resampling data* dengan menggunakan SMOTE, dimana hasil tersebut diperoleh dengan berdasar pada konsep SMOTE yang membangkitkan data sintetis baru dengan melakukan *oversampling* pada kelas minoritas, dimana kelas minoritas yang dimaksud jika berdasarkan pada data yaitu, kelas moderate dan *healthy control*, sehingga dilakukan *oversampling* pada kelas tersebut dengan persentase *oversampling* sebesar 380% untuk kelas *moderate*, dan 450% untuk kelas *healthy control*. Berdasarkan penambahan data sintetis dengan menggunakan teknik SMOTE tersebut, diperoleh jumlah sampel baru untuk data *moderate* yang awalnya sebesar 25 sampel, menjadi 94 sampel setelah diterapkan teknik SMOTE, dan kelas *healthy control* yang sebelumnya hanya 21 sampel, setelah dilakukan SMOTE bertambah menjadi 94 sampel.

Berdasarkan data baru yang dihasilkan setelah dilakukan penanganan *imbalanced* dengan SMOTE, selanjutnya akan dilakukan analisis klasifikasi kembali dengan menggunakan dua metode klasifikasi sebelumnya yaitu SVM, dan *Naïve Bayes* menggunakan data baru pada tabel 5.9 tersebut.

5.8 SMOTE dengan SVM

Setelah dilakukan proses penangan *imbalancedd* data dengan menggunakan SMOTE dan didapatkan data yang sudah *balanced* untuk digunakan dalam melakukan klasifikasi kembali, diperoleh hasil klasifikasi dengan menggunakan metode SVM untuk masing-masing kernel pada SVM berupa nilai akurasi yang menunjukkan seberapa baik model dapat mengklasifikasikan data yang telah dilakukan penanganan *imbalancedd* data tersebut. Sehingga diperoleh hasilnya seperti pada tabel dibawah ini:

Tabel 5.22 Akurasi SVM dengan Menggunakan SMOTE

Kernel	Akurasi
Linear	98.25%
Polynomial	96.49%
Sigmoid	63.16%
RBF	45.61%

5.9 SMOTE dengan Naïve Bayes

Penanganan *imbalanced* data dengan menggunakan SMOTE menghasilkan dataset baru yang lebih *balance* atau seimbang dibanding sebelumnya, sehingga akan dilakukan klasifikasi kembali dengan menggunakan *naïve bayes* berdasarkan data yang telah seimbang tersebut. Sehingga diperoleh hasil klasifikasi dengan *Naïve Bayes Classifier* ialah sebagai berikut.

Tabel 5.23 Kinerja Klasifikasi *Naïve Bayes* dengan SMOTE

Akurasi	71,43%
Recall	59.73%
Specificity	84.14%

5.10 Membandingkan Model Klasifikasi (SVM, & *Naïve Bayes*)

Setelah memperoleh hasil klasifikasi dari metode yang digunakan, yaitu metode SVM, dan *Naïve Bayes*, selanjutnya akan dilakukan perbandingan untuk mengetahui model klasifikasi terbaik dari data ekspresi sputum yang telah diinduksi pada pasien yang menderita penyakit asma, dengan melihat dari nilai akurasi untuk masing-masing metodenya, dimana untuk hasil perbandingan akurasi dari kedua model yang digunakan (SVM, dan *Naïve* Bayes) tersebut dapat dilihat pada tabel dibawah ini

Tabel 5.24 Nilai Akurasi Metode SVM, dan *Naïve Bayes*

Metode Klasifikasi	Nilai Akurasi
Support Vector Machine	78.57%
Naïve Bayes	71.43%
SVM dengan SMOTE	98.25%
Naïve Bayes dengan SMOTE	68.42%

Berdasarkan dari hasil pada tabel diatas, metode yang dapat mengklasifikasikan data yang digunakan sehingga diperoleh nilai akurasi yang paling baik, dimana nilai akurasi merupakan kedekatan nilai prediksi dengan data aktual, adalah dengan menggunakan metode *support vector machine* dengan penanganan *imbalanced data* menggunakan SMOTE, dimana dengan menggunakan metode tersebut, diperoleh nilai akurasi sebesar 98.25%, nilai tersebut lebih besar dibandingkan dengan hasil dari klasifikasi untuk metode lainnya.

5.11 Hasil dari Klasifikasi Model Terbaik (SVM dengan SMOTE)

Berdasarkan pada *subbab* sebelumnya ataupun hasil pada tabel 5.8, diketahui bahwa model yang dapat mengklasifikasikan data ekspresi sputum yang telah diinduksi pada pasien yang menderita penyakit asma secara baik dan menghasilkan akurasi dengan nilai tertinggi, adalah metode *support vector machine*, dengan nilai akurasi sebesar 98.25%, dimana nilai tersebut tergolong baik bagi suatu model untuk dapat mengklasifikasikan data yang digunakan. Berikutnya akan ditampilkan hasil dari *confusion matrix* untuk klasifikasi dengan menggunakan metode *support*

vector machine, dimana hasil dari *confusion matrix* ini digunakan untuk mengetahui nilai-nilai dari metrik evaluasi, dimana beberapa contohnya seperti, *accuracy, recall/sensitivity, specivicity,* dan *precission*

Tabel 5.25 *Confusion Matrix Data Testting* SVM

Prediksi	Aktual		
	Severe	Moderate	Healthy Control
Severe	18	0	0
Moderate	1	19	0
Healthy Control	0	0	19

Berdasarkan tabel 5.6 diatas, terdapat 57 sampel yang didapatkan dari hasil pembagian data testing sebesar 20% pada dataset yang dilakukan penanganan *imabalanced* dataa, dan akan digunakan untuk melihat perbandingan dari hasil prediksi dan data aktual, dengan penjelasan sebagai berikut:

- Pada kategori *severe*, model dapat memprediksi dengan benar sebanyak 18 sampel sesuai dengan kategori aslinya, lalu 1 sampel di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *severe* (false positive), dan tidak ada atau 0 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *severe* (true negative).
- Pada kategori *moderate*, model memprediksi sebanyak 1 sampel di mana model memprediksi kategori *severe* tetapi kategori sebenarnya *moderate* (false negative), 2 sampel di mana model memprediksi dengan benar kategori *moderate* dan kategori sebenarnya juga *moderate* (true positive), dan 2 sampel di mana model memprediksi kategori *healthy control* tetapi kategori sebenarnya *moderate* (false negative).
- Pada kategori *healthy control*, terdapat 0 kasus di mana model memprediksi kategori *severe* tetapi kategori sebenarnya adalah *healthy control* (false positive), 0 kasus di mana model memprediksi kategori *moderate* tetapi kategori sebenarnya *healthy control* (false positive), dan 2 kasus di mana

model memprediksi dengan tepat kategori *healthy control* dan kategori sebenarnya juga *healthy control* (true positive)

Selanjutnya dari hasil klasifikasi menggunakan *support vector machine* yang terdapat pada *confusion matrix* yang telah dibuat sebelumnya pada tabel 5.9, dapat diketahui hasil dari perhitungan kinerja klasifikasi dari model *support vector machine* dengan melihat hasil dari beberapa *performance metrics* atau metrik evaluasi klasifikasi. Pertama akan ditampilkan nilai akurasi dari hasil klasifikasi dengan menggunakan metode SVM pada data *testting* dengan menggunakan perhitungan seperti berikut:

$$Accuracy = \frac{Jumlah\ Prediksi\ Benar}{Jumlah\ Data\ Uji} \times\ 100\% = \frac{56}{57} = 0.9825 \times 100\%\ = 98.25\%$$

Nilai akurasi dari hasil klasifikasi menggunakan metode *support vector machine* adalah sebesar 98.25%, nilai tersebut didapatkan dari 56 sampel yang diprediksi benar sesuai dengan data aslinya (*True Positive*), yang dibagi dengan banyaknya data yang digunakan dalam pengujian atau data *testting* sebanyak 57 sampel.

Setelah diketahui nilai akurasi dari metode klasifikasi *support vector machine* tersebut, selanjutnya akan ditampilkan hasil dari perhitungan kinerja dari model dengan menggunakan beberapa perhitungan metrik penting lainnya seperti, *precision, recall/sensitivity, spesivicity,* F1-*Score*, dan AUC

- *Precision Severe* $= \frac{TP}{(TP+FP)} = \frac{18}{(18+0+0)} = 1.00$

  Precision Moderate $= \frac{TP}{(TP+FP)} = \frac{19}{(19+1+0)} = 0.95$

  Precision Healthy control $= \frac{TP}{(TP+FP)} = \frac{19}{(19+0+0)} = 1.00$

  Precision $= \frac{Precision\ severe+precision\ moderate+precision\ healthy\ control}{\text{Banyak Kelas}}$

  $= \frac{1+0.95+1}{3} \times 100\% = 0.9833 = 98.33\%$

- *Recall Severe* $= \frac{TP}{(TP+FN)} = \frac{18}{(18+1+0)} = 0.9474$

  Recall Moderate $= \frac{TP}{(TP+FN)} = \frac{19}{(19+0+0)} = 1.00$

  Recall Healthy control $= \frac{TP}{(TP+FN)} = \frac{19}{(19+0+0)} = 1.00$

Recall $= \frac{Recall\ severe + Recall\ moderate + Recall\ healthy\ control}{\text{Banyak Kelas}}$

$= \frac{0.9474+1+1}{3} \times 100\% = 0.9825 = 98.25\%$

- *Specificity Severe* $= \frac{TN}{(TN+FP)} = \frac{37}{(37+0)} = 1.00$

  Specificity Moderate $= \frac{TN}{(TN+FP)} = \frac{38}{(38+1)} = 0.9737$

  Specificity Healthy control $= \frac{TN}{(TN+FP)} = \frac{37}{(37+0)} = 1.00$

  Specificity $= \frac{Specificity\ severe + Specificity\ moderate + Specificity\ healthy\ control}{\text{Banyak Kelas}}$

  $= \frac{1+0.9737+1}{3} \times 100\% = 0.9912 = 99.12\%$

- F1-*Score* $= \frac{2 * Precision * recall}{precision + recall}$

  $= \frac{2 * 0.9833 * 0.9825}{0.9833 + 0.9825} \times 100\% = 0.9829 = 98.29\%$

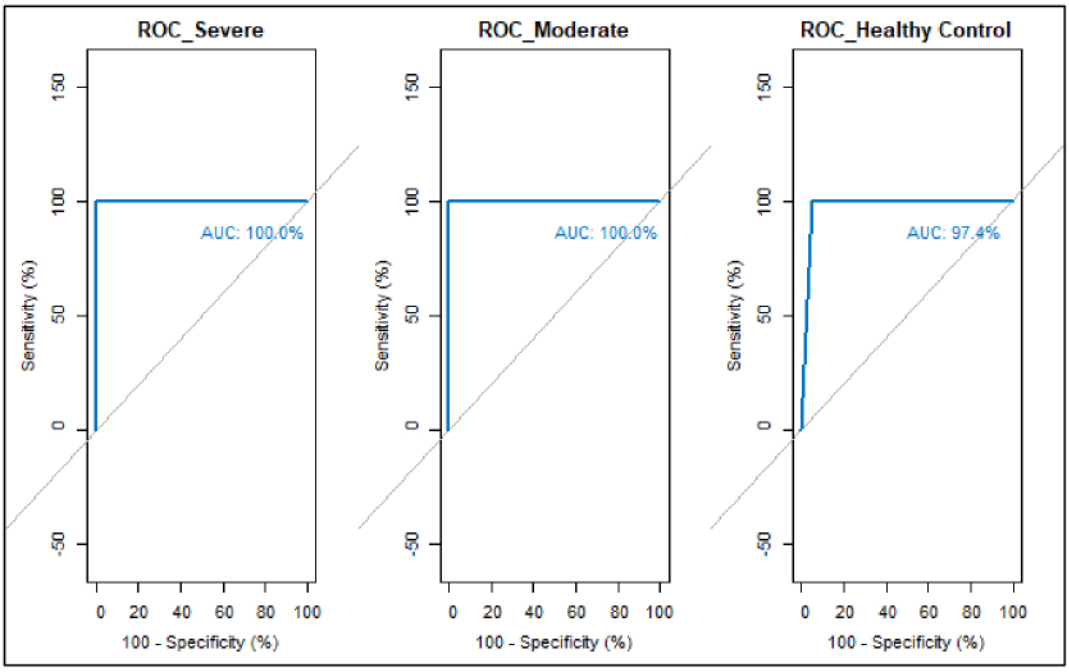
Tabel 5.26 Hasil *Performance Metrics* Pada Model

Performance Metrics	Nilai
Accuracy	98.25%
Precision	98.33%
Recall	98.25%
Specificity	99.12%
F1-*Score*	98.29%

Berdasarkan hasil dari analisis klasifikasi menggunakan metode *support vector machine* terhadap data ekspresi gen *induced sputum* pada penderita penyakit asma, diperoleh nilai akurasi yang merupakan kedekatan nilai prediksi dengan nilai aktual sebesar 98.25%. Selanjutnya diperoleh nilai *precision* atau rasio kelas positif yang diprediksi dengan benar terhadap semua kelas positif yang diprediksi dengan nilai *precision* sebesar 98.33%. Lalu diperoleh nilai *recall* yang merupakan seberapa besar model dapat memprediksi sampel yang benar-benar positif atau besaran true positif pada model, dengan nilai *recall* sebesar 98.25%. Selanjutnya terdapat nilai *specificity* atau spesifisitas yang mengukur proporsi *true* negatif yang diidentifikasi dengan benar oleh model, dimana diperoleh nilai *specificity* pada

model ini sebesar 99.12%. Lalu untuk metrik selanjutnya atau F1-*Score* diperoleh nilai F1-*Score* sebesar 98.29%, dimana F1-*Score* sendiri merupakan penggabungan dari *precision* dan *specificity* untuk melihat seberapa besar model dapat memprediksi data yang digunakan dengan benar

Selain dari beberapa metrik yang telah dijabarkan diatas, diperoleh satu metrik lainnya yaitu AUC atau *Area Under Control*, yang digunakan untuk pengukuran kualitas *classifier* probabilistik. Pada *multiclass* klasifikasi, AUC dapat digunakan dengan menggunakan metode "*One vs Rest*", dan seperti namanya, adalah salah satu metode untuk mengevaluasi model *multiclass* dengan membandingkan setiap kelas dengan kelas lainnya pada saat yang bersamaan. Dalam skenario ini, satu kelas akan diambil dan menganggapnya sebagai kelas "positif", sementara yang lainnya (sisanya) dianggap sebagai kelas "negatif", dan mengulangnya sebanyak 3 kali sesuai dengan jumlah kelas atau kategori yang digunakan dalam data. Selanjutnya akan ditampilkan hasil dari *plot* untuk masing-masing kelas atau kategori untuk melihat nilai AUC. Berikut merupakan *plot* tersebut



Gambar 5.5 AUC pada *Multiclass Classification*

Berdasarkan plot tersebut dapat diketahui pada *plot* pertama atau untuk kelas *severe*, *severe* dikategorikan sebagai kelas positif dan diplotkan terhadap kelas *moderate* dan *healthy control* yang dikategorikan sebagai kelas negatif, sehingga diperoleh nilai AUC nya sebesar 100%, lalu pada kelas *moderate*, diplotkan kurva untuk kelas moderate yang dianggap sebagai kelas positif, terhadap *severe* dan *healthy control* yang dikategorikan sebagai kelas negatif, dan didapatkan nilai AUC untuk kelas *moderate* ialah sebesar 100%, dan yang terakhir ialah pemlotan kurva untuk kelas *healthy control* yang dikategorikan sebagai kelas positif terhadap *severe* dan *moderate* yang dikategorikan sebagai kelas negatif, sehingga diperoleh nilai AUC nya sebesar 97.4%. Berikut merupakan nilai AUC untuk masing-masing kelas pada pengklasifikasian menggunakan metode *support vector machine* terhadap data ekspresi gen *induced sputum* pada pasien yang menderita penyakit asma.

Tabel 5.27 Nilai AUC

Kelas	AUC
Severe	100%
Moderate	100%
Healthy Control	97.4%

Berdasarkan tabel 5.11 tersebut, didapatkan nilai AUC untuk *multiclass classfication* dengan menggunakan metode "*One* vs *Rest*", dimana untuk masing-masing kelas nya, diperoleh nilai AUC untuk kelas *Severe* sebesar 67.4%, kelas *moderate* sebesar 98.7%, dan untuk kelas *healthy control* sebesar 90%. Setelah diperoleh nilai AUC untuk masing-masing kelas tersebut, selanjutnya akan dilakukan perhitungan untuk mendapatkan nilai AUC keseluruhan terhadap model yang digunakan dari metode *support vector machine*, dengan menggunakan perhitungan sebagai berikut.

$$\text{AUC} = \frac{AUC\ severe + AUC\ moderate + AUC\ healthy\ control}{\text{Banyak Kelas}}$$

$$= \frac{100+100+97.4}{3} \times 100\% = 99.13\%$$

Diperoleh nilai AUC *overall* atau nilai AUC untuk keseluruhan kelas dalam model yang telah diprediksi ialah sebesar 99.13%, dimana nilai tersebut jika

mengacu pada ketetapan kriteria penilaian AUC (tabel 3.2), dapat disimpulkan bahwa klasifikasi dengan menggunakan metode *support vector machine* merupakan *Excellent* Classifier, dimana nilai tersebut merupakan gambaran seberapa baik klasikasi yang dilakukan dapat memprediksi setiap kelas yang terdapat pada model yang digunakan atau pada data ekspresi gen *induced sputum* pada penyakit pernafasan atau asma.

Setelah itu ingin dicari tau beberapa variabel atau gen yang merupakan *gene importance* pada data ekspresi gen induksi sputum pada penyakit asma, dimana diketahui pada tahap *filtering*, diperoleh sebanyak 3330 *probe* sebagai variabel untuk penelitian dari model yang diperoleh, dan diambil lima variabel dengan nilai *weight* atau bobot terbesar dari 3330 variabel yang diidentifikasi.

Tabel 5.28 *Weight Probe Id*

Probe Id	Weight (w)
X207900_PM_at	0.01696484
X228492_PM_at	0.01649665
X240227_PM_at	0.01611175
X241730_PM_at	0.01608269
X242809_PM_at	0.01379929

Pada tabel diatas diketahui bobot untuk masing-masing *probe*, dan diambil 5 *probe* dengan bobot tertinggi, dimana *probe* dengan bobot tertinggi yaitu, *probe* X207900_PM_at dengan *symbol* CCL17 merupakan *probe* dengan nilai bobot paling besar, hal tersebut berarti bahwa *probe* dengan gen *symbol* CCL17 merupakan gen dengan pengaruh paling besar dalam pembentukan model dan juga merupakan salah satu gen yang memiliki keterkaitan paling tinggi pada pasien dengan penderita penyakit asma.

Untuk penjelasan selanjutnya mengenai nama *symbol* dan keterkaitannya dengan asma untuk masing-masing *probe* pada tabel 5.28 diatas, akan dijelaskan pada tabel dibawah ini, dengan penjelasan untuk masing-masing *symbol* dari *probe* yang paling berpengaruh yang mengacu pada penjelasan dan keterangan dari *website* NCBI.

Tabel 5.29 Keterangan *Probe* ID dari Hasil Analisis SVM Kernel Linear

No	Probe Id	Symbol	Keterangan
1	X207900_PM_at	CCL17	Gen ini mengkode C-C motif chemokine ligand 17, Gen antimikroba ini adalah salah satu dari beberapa gen sitokin Cys-Cys (CC) yang dikelompokkan pada lengan q kromosom 16. Sitokin adalah keluarga protein yang disekresikan yang terlibat dalam proses imunoregulasi dan inflamasi. Sitokin CC adalah protein yang dicirikan oleh dua sistein yang berdekatan. Sitokin yang dikodekan oleh gen ini menunjukkan aktivitas kemotaktik untuk limfosit T, tetapi tidak untuk monosit atau granulosit. Produk dari gen ini berikatan dengan reseptor kemokin CCR4 dan CCR8. Kemokin ini memainkan peran penting dalam perkembangan sel T dalam timus serta dalam peredaran dan aktivasi sel T dewasa.
2	X228492_PM_at	USP9Y	Gen ini adalah anggota keluarga peptidase C19. Gen ini mengkode protein yang mirip dengan protease spesifik ubiquitin, yang membelah bagian ubiquitin dari prekursor yang menyatu dengan ubiquitin dan protein yang terubiquitinasi.
3	X240227_PM_at	MIER1	Gen ini mengkode protein yang pertama kali diidentifikasi dalam Xenopus laevis melalui perannya dalam respon awal induksi mesoderm (MIER). Protein yang dikodekan berfungsi sebagai pengatur transkripsi.

No	Probe Id	Symbol	Keterangan
			Varian transkrip yang disambung secara alternatif menyandikan beberapa isoform, beberapa di antaranya tidak memiliki sinyal lokalisasi nuklir terminal-C.
4	X241730_PM_at	MYNN	Gen ini mengkode anggota keluarga protein yang mengandung domain BTB/POZ dan zinc finger yang terlibat dalam kontrol ekspresi gen. Penyambungan alternatif menghasilkan beberapa varian transkrip dan pseudogene telah diidentifikasi pada kromosom 14
5	X242809_PM_at	IL1RL1	Protein yang dikodekan oleh gen ini adalah anggota keluarga reseptor interleukin 1. Studi tentang gen serupa pada tikus menunjukkan bahwa reseptor ini dapat diinduksi oleh rangsangan proinflamasi, dan mungkin terlibat dalam fungsi sel T pembantu. Gen ini, reseptor interleukin 1, tipe I (IL1R1), reseptor interleukin 1, tipe II (IL1R2) dan reseptor interleukin 1 seperti 2 (IL1RL2) membentuk kluster gen reseptor sitokin di wilayah yang dipetakan ke kromosom 2q12. Penyambungan alternatif dari gen ini menghasilkan beberapa varian transkrip.

Probe dengan gen simbol CCL17 merupakan salah satu gen yang biasanya berperan dalam menyebabkan peradangan pada saluran udara yang biasa terjadi pada penderita asma. Gen CCL17 menyebabkan terjadinya pelemahan pada sel yang melindungi dan memberikan kekebalan pada saluran pernapasan, sehingga hal tersebut merupakan hal yang mendasari terjadinya peradangan akut pada saluran

udara yang menyebabkan timbulnya gejala batuk-batuk bahkan sampai sesak nafas yang biasa terjadi pada penderita penyakit asma

Selain dari CCL17, terdapat *probe* MIER1 yang juga memiliki keterkaitan pada asma, dimana keberadaan gen MIER1 seringkali merupakan penyebab ternjadinya disfungsi mitokondria, dimana disfungsi mitokondria memiliki hubungan langsung dengan asma, karena jika terjadi disfungsi mitokondria pada sel tubuh kita, hal tersebut akan menyebabkan suatu respon yang tidak baik pada saluran pernapasan, sehingga hal tersebut dapat menyebabkan terjadinya inflamasi pada saluran napas, alergi hingga asma.

Probe dengan simbol IL1RL1 memiliki hubungan yang signifikan antara dengan kerentanan asma. Variasi genetik atau polimorfisme tertentu pada gen IL1RL1 telah ditemukan lebih umum pada individu dengan asma dibandingkan dengan mereka yang tidak menderita asma atau pada gangguan pernapasan ringan. Varian genetik ini dapat memengaruhi ekspresi atau fungsi reseptor IL-33, yang menyebabkan respons imun yang tidak teratur dan peningkatan risiko asma. Gen IL1RL1 dan jalur pensinyalan IL-33 telah terlibat dalam menghasilkan peradangan saluran napas, hiperresponsif saluran napas, dan respons alergi yang biasa terjadi pada penderita asma. Aktivasi pensinyalan reseptor IL-33 merangsang produksi sitokin dan kemokin pro-inflamasi, sehingga menghambat sel kekebalan menuju saluran udara dan memperburuk gejala asma.

BAB VI
PENUTUP

6.1 Kesimpulan

1. Gambaran umum dari data ekspresi gen *induced sputum* pada pasien penderita penyakit asma, diketahui bahwa sampel untuk pasien perempuan lebih banyak dari laki-laki, dengan jumlah sampel untuk perempuan sebanyak 76 sampel, dan untuk laki-laki sebanyak 57 sampel, dan juga diketahui bahwa rata-rata usia sampel yang paling banyak menderita penyakit asma berada di rentang usia 40 sampai 60 tahun.
2. Hasil dari Klasifikasi menggunakan metode *support vector machine* didapatkan hasil bahwa kernel linier dan polynomial merupakan kernel yang paling dapat mengklasifikasikan data dengan baik dibandingkan kernel SVM yang lain, dengan nilai akurasi sebesar 78.57%, sedangkan untuk metode *naïve bayes classifier* diperoleh nilai akurasi dimana metode tersebut mampu mengklasifikasikan data secara baik adalah sebesar 71.43%, dan pada klasifikasi yang telah dilakukan penanganan *imbalanced* data dengan SMOTE, metode klasifikasi SVM dengan SMOTE mampu memberikan hasil akurasi sebesar 98.25%, dan untuk klasifikasi menggunakan metode *Naïve Bayes* dengan SMOTE mendapatkan nilai akurasi klasifikasi sebear 68.42
3. Berdasarkan hasil dari nilai akurasi untuk metode yang digunakan, baik yang pada data imbalanced ataupun pada data yang telah dilakukan penanganan imbalanced dengan SMOTE, diperoleh hasil bahwa metode yang paling mampu untuk mengklasifikasikan dataset yang digunakan, yaitu data ekspresi gen pada pasien penyakit asma yang telah dilakukan peinginduksian sputum atau dahak, adalah metode Support Vector Machine yang dilakukan penanganan imbalanced dengan SMOTE, yaitu dengan nilai akurasi sebesar 98.25%

4. Metode dengan nilai akurasi terbaik dari analisis klasifikasi yang digunakan adalah SVM yang menggunakan penanganan imbalanced dengan SMOTE, dimana nilai akurasi tersebut didapat dari confusion matrix data testting dengan jumlah sampel yang diuji sebanyak 57 sampel, dimana pada confusion matrix tersebut metode SVM mampu memprediksikan secara benar sesuai dengan data aktual nya untuk kategori severe sebanyak 18 sampel, untuk kategori moderate diprediksi secara benar sebanyak 19 sampel, dan untuk kategori healthy control mampu memprediksi secara benar sesuai dengan data aslinya sebanyak 19 sampel. Sehingga diperoleh prediksi benar sebanyak 56 dari 57 sampel, sehingga diperoleh nilai akurasinya sebesar 98.25%. selain dari nilai akurasi, terdapat beberapa metrik evaluasi lain pada klasifikasi menggunakan SVM, seperti nilai Precision sebesar 98.33%, Recall sebesar 98.25%, Specificity sebesar 99.12%, F1-Score sebesar 98.29%, dan juga diperoleh nilai AUC untuk keseluruhan data yaitu sebesar 99.13%, dimana nilai tersebut menunjukkan bahwa klasifikasi dengan menggunakan metode support vector machine termasuk dalam kategori Excellent Classifier

6.2 Saran

Adapun saran pada penelitian ini sebagai upaya perbaikan dan pengembangan atau penelitian lanjutan adalah sebagai berikut:

1. Penelitian selanjutnya dapat menggunakan beberapa metode klasifikasi lain untuk mendapatkan perbandingan hasil yang lebih banyak dalam melakukan klasifikasi data ekspresi gen induksi sputum pada pasien penderita penyakit asma.
2. Pada *probe* dengan 5 bobot tertinggi atau pada *gene importance* dapat dilakukan penelitian lebih lanjut lagi, bagaimana *gene-gene* tersebut dapat memberikan pengaruh yang besar terhadap pasien penderita penyakit asma.

DAFTAR PUSTAKA

Aini, S., Yulita, S., & Achmad, A. (2018). Seleksi Fitur Information Gain untuk Klasifikasi Penyakit Jantung Menggunakan Kombinasi Metode K-Nearest Neighbor dan Naïve Bayes. *JPTIIK UB*, 9.

Akbar, W., & WU, W.-P. M. (2019). *Machine Learning Classifiers for Asthma Disease Prediction: A Practical Illustration.*

Aprijani, D. A., & Elfaizi, M. A. (2004). *BIOINFORMATIKA: Perkembangan, Disiplin Ilmu dan Penerapannya di Indonesia.*

Aruna S., R. S. (2011). A novel SVM based CSSFFS feature selection algorithm for detecting breast cancer. *International Journal of Computing*, 31:14-20.

BBC. (2014, December 30). *AirAsia QZ8501: Does bad weather cause plane crashes?* Diambil kembali dari BBC: http://www.bbc.com/news/world-30631968

BioInformatics.org: The Open-Access Institute,. (2004). Diambil kembali dari BioInformatics.org: http://www.bioinformatics.org/

Bolstad, B. M. (2004). *Fundamentals of Data Mining in Genomics and Proteomics.* Boston: Springer.

Bramer, M. (2007). *Principles of Data Mining.* Springer.

Bustami. (2014). PENERAPAN ALGORITMA NAIVE BAYES UNTUK MENGKLASIFIKASI DATA NASABAH ASURANSI . *Jurnal Informatika*, Vol. 8, No. 1.

Chawla, N. (2002). Syntethic Minority Over-Sampling Technique. *Journal of Artificial*.

Chong-Silva, D. C., Nascimento, A., Cunha, R., Bitencourt, E., Botelho, L., Dantas, B. A., . . . Riedi, C. (2021). EVALUATION OF INDUCED SPUTUM CYTOLOGY IN ASTHMATIC CHILDREN. *AUTHOREA*.

Defiyanti, S., & Jajuli, M. (2015). *Integrasi Metode Klasifikasi Dan Clustering dalam Data Mining.*

Devi, A. V., & Venkatesulu, M. (2015). Gene Expression Data Classification using Support Vector Machine and Mutual Information-based Gene Selection. *Science Direct*, 13-21.

E. Pizzichini, R. L., & R. Djukanovic, P. S. (2002). Safety of sputum induction. *ERS Journal*, 20:9-18.

Fan, L., & Kim-Leng Poh a, &. P. (2009). A sequential feature extraction approach for naïve bayes classification of microarray data. *Expert Systems with Applications*, 9919-9923.

Friedman, N., Geiger, D., & Moises, G. (1997). Bayesian Network Classifiers. *Machine Learning*, 131-163. Diambil kembali dari ResearchGate.

García-Sánchez, A., & Marqués-García, F. (2016). Review of Methods to Study Gene Expression Regulation Applied to Asthma. Dalam N. Clifton, *Methods in molecular biology* (hal. 71-89).

Gentleman, R., Carey, V., Huber, W., & Hahne, F. (2019). *Package 'genefilter'.*

Gibney, E. R., & Nolan, C. M. (2010). Epigenetics and gene expression. *Heredity*.

GINA, G. I. (2012). *Global Strategy for Asthma Management and Prevention.* GINA.

Gonzalo Sanz, R., & Sánchez, A. (2018). Introduction to Microarrays Technology and Data Analysis. *Comprehensive Analytical Chemistry*.

Groth, E. E., Weber, M., Bahmer, T., Pedersen, F., Kristen, A., Bornigen, D., & Rabe, K. F. (2020). Exploration of the sputum methylome and omics deconvolution by quadratic programming in molecular profiling of asthma and COPD: the road to sputum omics 2.0. *Respiratory Research*.

Hair, J., Black, W., Babin, B., & Anderson, R. (2010). *Multivariate Data Analysis: A Global Perspective.* Pearson.

Hall, M. (2007). A decision tree-based attribute weighting filter for naive Bayes. *Science Direct*, 120-126.

Handayani, A., & Jamal, A. (2016). Evaluasi Algoritme Berbasis Pembelajaran Mesin untuk Klasifikasi Jenis Tumor Payudara. *JNTETI Vol. 6, No. 4*, 394-403.

Informatikalogi. (2017). Diambil kembali dari Algoritma Naive Bayes: https://informatikalogi.com/algoritma-naive-bayes/

IRRIB. (2017, September 19). *Bioinformatika Pada Teknologi Sekuensing Generasi Baru*. Diambil kembali dari irrib.org: https://iribb.org/bioinformatika-pada-teknologi-sekuensing-generasi-baru/

Johnson, R. A., & Bhattacharyya, G. K. (2010). *Statistics Principles & Methods.* USA: John Wiley & Sons.

Karabulut, E., Özel, S., & Ibrikci, T. (2012). Comparative study on the effect of feature selection on classification accuracy. *Procedia Technology*.

Khalimi, A. M. (2020). *Cara Menghitung Confusion Matrix 4 Kelas*. Diambil kembali dari Pengalaman-Edukasi: https://www.pengalaman-edukasi.com/2020/01/confusion-matrix-multi-class-menghitung.html

Koc-Günel, S., Schubert, R., Zielen, S., & Rosewich, M. (2018). Cell distribution and cytokine levels in induced sputum from healthy subjects and patients with asthma after using different nebulizer techniques. *BMC Pulmonary Medicine*.

Li, X. (2013). Comparison and Analysis between Holt Exponential Smoothing and Brown Exponential Smoothing Used for Freight Turnover Forecast. *Third International Conference on Intelligent System Design and Engineering Applications* (hal. 453-456). IEEE.

Metz, C. E. (2006). Receiver operating characteristic analysis: a tool for the quantitative evaluation of observer performance and imaging systems. *Journal of the American College of Radiology*, 13-22.

Mudrajat, K. (2013). *Metode Riset Untuk Bisnis & Ekonomi.* Jakarta: Erlangga.

Nor Azam, S. Z., Zakaria, N. H., Hassan, R., & Zulkifle, F. A. (2022). Classification of Psoriasis Microarray Data using Machine Learning. *IEEE*.

Novita, A., & Zabit, W. (2014). Hubungan Tingkat Pengetahuan Pasien Asma dengan Tingkat Kontrol Asma di Poliklinik Paru RSUD dr. Zainoel Abidin Banda Aceh. *Jurnal Kedokteran Syiah Kuala*.

Nugroho, A. S., Witarto, A. B., & Handoko, D. (2003). *Teori dan Aplikasinya dalam Bioinformatika.* IlmuKomputer.Com.

Perdew, G. H., Vanden Heuvel, J. P., & Peters, J. M. (2007). *Regulation of Gene Expression.* Totowa NJ: Humana Press.

Prasetyowati, E., & Ramadhani, N. (2018). SISTEM EVALUASI DAN KLASIFIKASI KINERJA AKADEMIK MAHASISWA UNIVERSITAS MADURA MENGGUNAKAN NAIVE BAYES DENGAN DIRICHLET SMOOTHING. *Jurnal Ilmiah Teknologi Informasi*, Vol. 16, No. 2, hlm: 192-202.

Prasojoe, R. R., & Setyorini. (2021). Uji Konsep Paralel SVM dengan Dekomposisi SMO Pada Data Set Cancer Microarray. *eProceedings of Engineering*, 3591.

Purnama, V. M., Astuti, W., & Adiwijaya. (2021). Analisis Perbandingan Klasifikasi Microarray menggunakan Naïve Bayes dan Support Vector Machine (SVM) untuk Deteksi Kanker dengan Feature Extraction PCA. *e-Proceeding of Engineering*, 9974.

Ramdaniah, R. A. (2019). Performance evaluation of naive Bayes and support vector machine in type 2 Diabetes Mellitus gene expression microarray data. *Journal of Physics: Conference Series*.

Raza, & Khalid. (2010). Application Of Data Mining In Bioinformatics. *Indian Journal of Computer Science and Engineering*, 114-118.

Rickard Sandberg, G. W.-I. (2001). Capturing whole-genome characteristics in short sequences using a naïve Bayesian classifier. *Genome research*, 1404-1409.

Rumus Statistik . (2013, Juli). Diambil kembali dari Rata-rata Hitung (Mean): https://www.rumusstatistik.com/2013/07/rata-rata-mean-atau-rataan.html

Rumus Statistik. (2013, Juli). Diambil kembali dari Varian dan Standar Deviasi (Simpangan Baku): https://www.rumusstatistik.com/2013/07/varian-dan-standar-deviasi-simpangan.html

Santosa, B. (2007). *Teknik pemanfaatan data untuk keperluan bisnis.* Yogyakarta: Graha Ilmu.

Santoso, I., Mariyah, S., Yuniarto, B., Pramana, S., & Nooraeni, R. (2018). *Data Mining dengan R.* Bogor: IN MEDIA.

Serin, A. (2011). Biclustering Analysis for Large Scale Data.

Suharman, R. A., & Hartono, H. (2022). Klasifikasi Kematangan Manggis Berdasarkan Fitur Warna dan Tekstur Menggunakan Algoritma Naive Bayes. *Pythagoras : Jurnal Pendidikan Matematika*.

SUKMAWATI, N. M. (2015). *BIOINFORMATIKA*. Denpasar: LABORATORIUM BIOKIMIA FAKULTAS PETERNAKAN UNIVERSITAS UDAYANA.

Surjanto, E., & Niwan, T. M. (2009). Induksi Sputum pada Asma. *Jurnal Respirologi Indonesia*.

Tramontano, A. (2018). *Introduction to Bioinformatics.* CRC Press.

Victor, A., & Langkah, S. (2017). *BIOINFORMATIKA*.

Walpole, R. E. (2011). *Probability & Statistics for Engineers & Scientists 9th Ed.* USA: Pearson.

Wasiati, H., & Wijayanti, D. (2014). Sistem Pendukung Keputusan Penentuan Kelayakan Calon Tenaga Kerja Indonesia Menggunakan Metode Naive Bayes (Studi Kasus: Di P.T. Karyatama Mitra Sejati Yogyakarta). *Indonesian Journal on Networking and Security* , Vol. 3, No. 2.

Wesolowska-Andersen, A., Everman, J., & Davidson, R. (2017). Dual RNA-seq reveals viral infections in asthmatic children without respiratory illness which are associated with changes in the airway transcriptome. *Genome Biol*.

Yang, P., Yang, J., Zhou, B., & Zomaya, A. (2010). A Review of Ensemble Methods in Bioinformatics. *Current Bioinformatics*.

Zhao, W., Lai, X., Liu, D., Zhang, Z., Ma, P., Wang, Q., . . . Pan, Y. (2020). Applications of Support Vector Machine in Genomic Prediction in Pig and Maize Populations. *Frontiers in Genetics*.

LAMPIRAN

Lampiran 1 *Script* Rstudio

```
# Packages-packages
library(affy)
library(GEOquery)
library(Biobase)
library(affyPLM)
library(u133x3pcdf)
library(u133x3p.db)
library(genefilter)
library(AnnotationDbi)
if (!require("BiocManager", quietly = TRUE))
  install.packages("BiocManager")
BiocManager::install(version = "3.16")
BiocManager::install("biomaRt")
BiocManager::install("hgu133plus2.db")

## Input Data
GSE76262<- list.celfiles("F:/TA/GSE76262_RAW",
full.names=T)
GSE76262.AFFY = ReadAffy(filenames=GSE76262)
GSE76262.AFFY
class(GSE76262.AFFY)
### get pheno data ###
GSET_GSE76262 <- getGEO(GEO="GSE76262",GSEMatrix =TRUE)
GSET_GSE76262
str(GSET_GSE76262)
data.GSE76262 <- exprs(GSET_GSE76262[[1]])
class(data.GSE76262)
dim(data.GSE76262)
data.GSE76262[1:5,1:4]
pheno_GSE76262 <- pData(phenoData(GSET_GSE76262[[1]]))
str(pheno_GSE76262)
```

```
varLabels(phenoData(GSET_GSE76262[[1]]))
dim(pheno_GSE76262)
View(pheno_GSE76262)
View(GSET_GSE76262)
setwd('F://TA')
#save pheno in csv file
write.csv(pheno_GSE76262, file
="F://TA/data_pheno/pheno_GSE76262.csv")
#### analisis deskriptif ####
##BOXPLOT
library(dplyr)
dataaffy.1 = log2(exprs(GSE76262.AFFY))
dev.new(width=4+dim(GSET_GSE76262)[[2]]/5, height=6)
par(mar=c(2+round(max(nchar(sampleNames(GSET_GSE76262)))/2),4,2,1)
)
title <- paste ("GSE76262", '/','GPL13158', " selected samples",
sep ='')
boxplot(dataaffy.1, boxwex=0.7, notch=T, main=title, outline=F,
las=2)

#Melihat Pheno
table(pheno_GSE76262$`gender:ch1`)
table(pheno_GSE76262$`age:ch1`)

#Barplot Gender
View(pheno_GSE76262)
des <-table(pheno_GSE76262$`gender:ch1`)
des <- as.data.frame(des)
barplot(des$Freq, main='Gender',col=c('yellow','blue'),
        xlab= 'Gender', ylab='Pasien', ylim = c(0,80),
        names.arg= c("Female","Male"))
des<-count(pheno_GSE76262, "pheno_GSE76262$age")
colnames(pheno_GSE76262)
?barplot
```

```
library(plyr)
des2<-count(pheno_GSE76262, "characteristics_ch1.1")
des2$characteristics_ch1.1 = as.character(gsub(
  "age:", "", des2$characteristics_ch1.1))
percent <- round(des2$freq/sum(des2$freq)*100)
des2 <- as.data.frame(des2)
lbls2 <- paste(des2$characteristics_ch1.1, percent,
               '%', sep=' ')
pie(des2$freq,main ='Age',label= lbls2, col=c(
  "red","orange","yellow","blue","green"),border="brown",
       cex=0.5)
barplot(des2$freq,main ='Age',label= lbls2)
```

```
#### pre processing data ####
library(affyPLM)
eset.dChip=threestep(GSE76262.AFFY,
                     background.method = "RMA.2",
                     normalize.method="quantile",
                     summary.method="median.polish")
Ekspres.GSE76262 <- exprs(eset.dChip)
dim(eset.dChip)
#BOXPLOT Setelah Pre-processing
#set parameter and draw the plot
dev.new(width=4+dim(GSET_GSE76262)[[2]]/5, height=6)
par(mar=c(2+round(max(nchar(sampleNames(GSET_GSE76262))
)/2),4,2,1))
title <- paste ("GSE76262", '/','   GPL13158',
                " selected samples", sep ='')
boxplot(Ekspres.GSE76262, boxwex=0.7, notch=T, main=title,
        outline=F, las=2)
##### filtering #####
library(hgu133plus2.db)
```

```
filterdata <- nsfilter(eset.dChip, require.entrez =T,
                       var.func = IQR,
                       remove.dupEntrez = T,
                       var.cutoff = 0.5,
                       feature.exclude = "^AFFX")
log <-filterdata$filter.log
eset <- filterdata$eset
class(eset)
featureNames(eset) <- make.names(featureNames(eset))
#Melihat hasil filter (eset)
dim(eset)
head(eset)
class(eset)
eset
```

```
# Membuat data microarray menjadi matrix/data frame
databaru <- exprs(eset)
dim(databaru)
class(databaru)
head(databaru)
View(databaru)

#Kelas Vektor ada 3(Normal, Intolerance Glucose dan Diabetes
Melitus)
#Kelas Vektor ada 3(severe, moderate, healthy control)
datacl <- c(0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,
            0,0,0,0,0,0,0,0,0,0,0,0,1,1,1,1,1,1,1,
            1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,1,2,2,
            2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2,2)
length(datacl)
class(datacl)
```

```
datafac <- factor(datacl,levels=0:2,
                  labels= c("sev", "mod", "hlct"))
datafac

#### Filtering feature selection with multtest ####
library(multtest)
datattest <- mt.teststat(databaru,datacl,test="f")
class(datattest)
length(datattest)
qqnorm(datattest)
qqline(datattest)
length(datattest)



#Adjusting p-value (untuk melihat p-value yg sesuai dan tidak
sesuai)
rawp = 2 * (1 - pnorm(abs(datattest)))
length(rawp)
prosedur = c("Bonferroni", "Holm", "Hochberg", "BH", "BY")
adjusted = mt.rawp2adjp(rawp, prosedur)
data <- adjusted$adjp[,]
data1 <- data[order(adjusted$index), ]
head(data1)
dim(data1)
#mengambil kolom rawp
ffs <- data1[,1]
class(ffs)
length(ffs)
ffs[1 : 10]
#Adjusting rawp
datarawp <- data.frame(databaru, ffs)
row.names(datarawp) <- row.names(databaru)
class(datarawp)
head(datarawp)
```

```
dim(datarawp)

library(dplyr)
datarawpfilterfinal <- subset(datarawp, ffs < 0.00000001)
rownames(datarawpfilterfinal)
class(datarawpfilterfinal)
dim(datarawpfilterfinal)
head(datarawpfilterfinal)

## mendefinisikan data baru setelah filter
datadef <- datarawpfilterfinal[,1:139]
head(datadef)
dim(datadesf)
colnames(datadef)
data_siap   = as.data.frame (t((datadef)))
head(data_siap)
dim(data_siap)
dataY = as.factor(datacl)
dataY
data_use = as.data.frame(cbind(data_siap,dataY))
dim(data_use)
summary(data_use)

#Usepackages(analisis)
library(e1071)
library(pROC)
library(devtools)
library(caret)
sev<-dplyr::filter(data_use, data_use$dataY==0)
mod<-dplyr::filter(data_use, data_use$dataY==1)
hlct<-dplyr::filter(data_use, data_use$dataY==2)
view(data_us)
#penentuan ukuran data traning dan testing
set.seed(123)
```

```
rasio=8/10

n<-round(nrow(sev)*rasio)
sampel_sev<-sample(1:nrow(sev),n)

m<-round(nrow(mod)*rasio)
sampel_mod<-sample(1:nrow(mod),m)

x<-round(nrow(hlct)*rasio)
sampel_hlct<-sample(1:nrow(hlct),x)

#data training dan testing
#training
training_sev=sev[sampel_sev,]
dim(training_sev)
training_mod=mod[sampel_mod,]
dim(training_mod)
training_hlct=hlct[sampel_hlct,]
dim(training_hlct)

#testing
testing_sev=sev[-sampel_sev,]
dim(testing_sev)
testing_mod=mod[-sampel_mod,]
dim(testing_mod)
testing_hlct=hlct[-sampel_hlct,]
dim(testing_hlct)
data_training=rbind(training_sev,training_mod,training_hlct)
data_testting=rbind(testing_sev,testing_mod,testing_hlct)
dim(data_training)
dim(data_testting)
View(data_training)

setwd("F://TA/Data_Skripsi")
```

```
write.csv(data_training,
          file ="F://TA/Data_Skripsi/GSE_76262/train80gen.csv" )
write.csv(data_testting,
          file ="F://TA/Data_Skripsi/GSE_76262/test80gen.csv" )

#### analisis klasifikasi menggunakan metode SVM ####
### 1. svm kernel linier
# 1.1.tuning (best parameter)
set.seed(12345)
tuning_lin <- tune(svm, dataY~. ,
                   data = data_training, kernel="linear",
                   types = "C-clasification", ranges= list(
                     cost = c(0.1,0.01, 0.001,1 , 10 , 100)))
summary(tuning_lin)
```

```
# 1.2. model svm kernel linier
# model_lin <- svm(dataY~. , data_training, kernel = "linear")
model_lin <- svm(dataY~. , data_training, kernel = "linear",
                 cost=0.1,scale=T,types = "C-clasification",
                 decision.value=T)
pred_train_lin <- predict(model_lin, data_training)
table(pred_train_lin, data_training$dataY)
pred_test_lin<-predict(model_lin, data_testting, type ="response")
table(pred_test_lin, data_testting$dataY)
mean(pred_test_lin==data_testting$dataY)
confusionMatrix(pred_test_lin, data_testting$dataY)
library(pROC)
par(mfrow = c(1,3))
test_roc = multiclass.roc(data_testting$dataY ~
as.numeric(pred_test_lin),
                          levels = c("1","2"),
                          plot = TRUE,
                          percent = TRUE,
                          print.auc = TRUE,
```

```
                        print.auc.cex = par("0.9"),
                        col = "#0073C2FF",print.auc.x = 50,
                        print.auc.y = 70, legacy.axes = T,
                        main = "ROC_Healthy Control")

### 2. svm kernel polynomial
# 2.1. tuning (best parameter)
set.seed(12345)
tuning_pol <- tune(svm, dataY~. , data = data_training,
                   kernel="polynomial",types = "C-clasification",
                   ranges= list( cost =
c(10,100,200,300,400,500)))
summary(tuning_pol)
```

```
# 2.2. model svm kernel polynomial
model_pol <- svm(dataY~., data_training, kernel="polynomial",
                 degree=3, cost=10,
                 types="C-clasification", decision.value=T)
pred_train_pol <- predict(model_pol, data_training)
table(pred_train_lin, data_training$dataY)
pred_test_pol <- predict(model_pol, data_testting)
table(pred_test_pol, data_testting$dataY)
mean(pred_test_pol==data_testting$dataY)
confusionMatrix(pred_test_pol, data_testting$dataY)

### 3. svm kernel sigmoid
# 3.1 tuning (best parameter)
set.seed(12345)
tuning_sig <- tune(svm, dataY~. , data = data_training,
                   kernel="sigmoid",
                   types = "C-clasification",
                   ranges= list(cost = c(
                     0.1,1,10,100,200,300),
                     gamma=c(0.1,1,2,3,4,5)))
```

```
summary(tuning_sig)
# 3.2 model svm kernel sigmoid
model_sig <- svm(dataY~., data_training,
                 kernel="sigmoid", cost=10,gamma=3,
                 types="C-clasification", decision.value=T)
pred_train_sig <- predict(model_sig, data_training)
table(pred_train_sig, data_training$dataY)
pred_test_sig <- predict(model_sig, data_testting)
table(pred_test_sig, data_testting$dataY)
mean(pred_test_sig==data_testting$dataY)
confusionMatrix(pred_test_sig, data_testting$dataY)
```

```
### 4. svm kernel RBF
# 4.1. tuning (best parameter)
set.seed(12345)
tuning_RBF <- tune(svm, dataY~. , data = data_training,
                   kernel="radial",types = "C-clasification",
                   ranges= list(
                     cost = c(0.1,0.01, 0.001,1 , 10 , 100),
                     gamma=c(1,2,3,4,5)))
summary(tuning_RBF)
model_rbf <- svm(dataY~., data_training,
                 kernel="radial", cost=0.1, gamma=1,
                 types="C-clasification", decision.value=T)
pred_train_rad <- predict(model_rbf, data_training)
table(pred_train_rad, data_training$dataY)
pred_test_rad <- predict(model_rbf, data_testting)
table(pred_test_rad, data_testting$dataY)
mean(pred_test_rad==data_testting$dataY)
confusionMatrix(pred_test_rad, data_testting$dataY)

### NAIVE BAYES ###
```

```
#Membuat model prediksi Naive Bayes
library(naivebayes)
nb <- naive_bayes(dataY ~ ., data = data_training)
head(data_training)
dim(data_training)
data_training[,3330]
colnames(data_training)[3330]
#Melihat model yang telah dibuat
nb

#Visualisasi Model
par(mfrow=c(2,3))
plot(nb)
```

```
#Melakukan prediksi dengan data testing
pred_nb <- predict(nb, as.data.frame(data_testting))

# validation
#Membuat Confussion Matrix Naive Bayes
confnb <- table(data_testting$dataY, pred_nb)
confnb

#confusion matrix lengkap
library(caret)
confusionMatrix(pred_nb, data_testting$dataY)

# akurasi
accnb <- mean(pred_nb==data_testting$dataY)
accnb
```

Lampiran 2 Data yang sudah siap digunakan untuk dianalisis

X212607_PM_at	X222922_PM_at	X243376_PM_at	...	...	DataY
4.808810	5.494441	8.793325	...	...	0
6.475510	4.877196	8.039033	...	...	0
5.668862	5.585216	7.564811	...	...	0
6.247186	7.078701	8.416042	...	...	0
4.673748	5.089810	9.096420	...	...	0
4.587676	6.504486	9.224420	...	...	0
6.661937	4.010921	9.135385	...	...	0
4.279839	5.679496	8.377728	...	...	0
⋮	⋮	⋮	...	...	⋮
⋮	⋮	⋮	...	...	⋮
5.120743	6.672175	7.606985	...	...	1
6.310432	6.236063	6.904571	...	...	1
6.769211	6.835820	7.960409	...	...	1
6.175237	6.534622	8.336535	...	...	1
6.075158	5.586960	8.312419	...	...	1
⋮	⋮	⋮	...	...	⋮
⋮	⋮	⋮	...	...	⋮
5.923911	6.326017	7.411746	...	...	2
6.169812	6.725742	7.817180	...	...	2

Lampiran 3 Hasil Klasifikasi Metode SVM Kernel Linear, Polynomial, RBF, dan Sigmoid

- Kernel Linier

```
> confusionMatrix(pred_test_lin, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 18  3  0
         1  1  2  2
         2  0  0  2

Overall Statistics

               Accuracy : 0.7857
                 95% CI : (0.5905, 0.917)
    No Information Rate : 0.6786
    P-Value [Acc > NIR] : 0.1556

                  Kappa : 0.5227

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.9474  0.40000  0.50000
Specificity            0.6667  0.86957  1.00000
Pos Pred Value         0.8571  0.40000  1.00000
Neg Pred Value         0.8571  0.86957  0.92308
Prevalence             0.6786  0.17857  0.14286
Detection Rate         0.6429  0.07143  0.07143
Detection Prevalence   0.7500  0.17857  0.07143
Balanced Accuracy      0.8070  0.63478  0.75000
```

- Kernel Polynomial

```
> confusionMatrix(pred_test_pol, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 19  3  1
         1  0  2  2
         2  0  0  1

Overall Statistics

               Accuracy : 0.7857
                 95% CI : (0.5905, 0.917)
    No Information Rate : 0.6786
    P-Value [Acc > NIR] : 0.1556

                  Kappa : 0.4799

 Mcnemar's Test P-Value : 0.1116

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            1.0000  0.40000  0.25000
Specificity            0.5556  0.91304  1.00000
Pos Pred Value         0.8261  0.50000  1.00000
Neg Pred Value         1.0000  0.87500  0.88889
Prevalence             0.6786  0.17857  0.14286
Detection Rate         0.6786  0.07143  0.03571
Detection Prevalence   0.8214  0.14286  0.03571
Balanced Accuracy      0.7778  0.65652  0.62500
```

- Kernel Sigmoid

```
> confusionMatrix(pred_test_sig, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 16  5  4
         1  2  0  0
         2  1  0  0

Overall Statistics

               Accuracy : 0.5714
                 95% CI : (0.3718, 0.7554)
    No Information Rate : 0.6786
    P-Value [Acc > NIR] : 0.919

                  Kappa : -0.139

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.8421  0.00000  0.00000
Specificity            0.0000  0.91304  0.95833
Pos Pred Value         0.6400  0.00000  0.00000
Neg Pred Value         0.0000  0.80769  0.85185
Prevalence             0.6786  0.17857  0.14286
Detection Rate         0.5714  0.00000  0.00000
Detection Prevalence   0.8929  0.07143  0.03571
Balanced Accuracy      0.4211  0.45652  0.47917
```

- Kernel RBF

```
> confusionMatrix(pred_test_rad, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 19  5  4
         1  0  0  0
         2  0  0  0

Overall Statistics

               Accuracy : 0.6786
                 95% CI : (0.4765, 0.8412)
    No Information Rate : 0.6786
    P-Value [Acc > NIR] : 0.5891

                  Kappa : 0

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            1.0000   0.0000   0.0000
Specificity            0.0000   1.0000   1.0000
Pos Pred Value         0.6786      NaN      NaN
Neg Pred Value            NaN   0.8214   0.8571
Prevalence             0.6786   0.1786   0.1429
Detection Rate         0.6786   0.0000   0.0000
Detection Prevalence   1.0000   0.0000   0.0000
Balanced Accuracy      0.5000   0.5000   0.5000
```

Lampiran 4 Hasil Klasifikasi menggunakan *Naïve Bayes*

```
> confusionMatrix(pred_nb, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 16  2  0
         1  1  1  1
         2  2  2  3

Overall Statistics

               Accuracy : 0.7143
                 95% CI : (0.5133, 0.8678)
    No Information Rate : 0.6786
    P-Value [Acc > NIR] : 0.4293

                  Kappa : 0.4386

 Mcnemar's Test P-Value : 0.4459

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.8421  0.20000   0.7500
Specificity            0.7778  0.91304   0.8333
Pos Pred Value         0.8889  0.33333   0.4286
Neg Pred Value         0.7000  0.84000   0.9524
Prevalence             0.6786  0.17857   0.1429
Detection Rate         0.5714  0.03571   0.1071
Detection Prevalence   0.6429  0.10714   0.2500
Balanced Accuracy      0.8099  0.55652   0.7917
```

Lampiran 6 Klasifikasi dengan menangani data *imbalanced* menggunakan SMOTE

```
> confusionMatrix(pred_test_lin, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 18  0  0
         1  1 19  0
         2  0  0 19

Overall Statistics

               Accuracy : 0.9825
                 95% CI : (0.9061, 0.9996)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.9737

 Mcnemar's Test P-Value : NA

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.9474   1.0000   1.0000
Specificity            1.0000   0.9737   1.0000
Pos Pred Value         1.0000   0.9500   1.0000
Neg Pred Value         0.9744   1.0000   1.0000
Prevalence             0.3333   0.3333   0.3333
Detection Rate         0.3158   0.3333   0.3333
Detection Prevalence   0.3158   0.3509   0.3333
Balanced Accuracy      0.9737   0.9868   1.0000
```

```
> confusionMatrix(pred_nb, data_testting$dataY)
Confusion Matrix and Statistics

          Reference
Prediction  0  1  2
         0 16  1  2
         1  1  8  2
         2  2 10 15

Overall Statistics

               Accuracy : 0.6842
                 95% CI : (0.5476, 0.8009)
    No Information Rate : 0.3333
    P-Value [Acc > NIR] : 6.644e-08

                  Kappa : 0.5263

 Mcnemar's Test P-Value : 0.149

Statistics by Class:

                     Class: 0 Class: 1 Class: 2
Sensitivity            0.8421   0.4211   0.7895
Specificity            0.9211   0.9211   0.6842
Pos Pred Value         0.8421   0.7273   0.5556
Neg Pred Value         0.9211   0.7609   0.8667
Prevalence             0.3333   0.3333   0.3333
Detection Rate         0.2807   0.1404   0.2632
Detection Prevalence   0.3333   0.1930   0.4737
Balanced Accuracy      0.8816   0.6711   0.7368
```