IDENTIFIKASI KANDIDAT BIOMARKER PADA PENYAKIT DIABETES MELLITUS TIPE 2 MENGGUNAKAN *RECURSIVE FEATURE EXTRACTION*

SKRIPSI

OLEH:

NUR NILAMYANI

H13114514

PROGRAM STUDI ILMU KOMPUTER

DEPARTEMEN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

DESEMBER 2018

IDENTIFIKASI KANDIDAT BIOMARKER PADA PENYAKIT DIABETES MELLITUS TIPE 2 MENGGUNAKAN *RECURSIVE FEATURE EXTRACTION*

SKRIPSI

Diajukan sebagai salah satu syarat untuk memperoleh gelar Sarjana Komputer pada Program Studi Ilmu Komputer Departemen Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin Makassar

NUR NILAMYANI
H13114514

PROGRAM STUDI ILMU KOMPUTER DEPARTEMEN MATEMATIKA

FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM

UNIVERSITAS HASANUDDIN

DESEMBER 2018

LEMBAR PERYATAAN KEOTENTIKAN

Saya yang bertanda tangan di bawah ini menyatakan dengan sungguh-sungguh bahwa skripsi ang saya buat dengan judul:

IDENTIFIKASI KANDIDAT BIOMARKER PADA PENYAKIT DIABETES MELLITUS TIPE 2 MENGGUNAKAN *RECURSIVE FEATURE EXTRACTION*

adalah benar hasil karya saya sendiri bukan hasil plagiat dan belum pernah dipublikasikan dalam bentuk apapun

Makassar, 19 Desember 2018

Nur Nilamyani
NIM. H 131 14 514

IDENTIFIKASI KANDIDAT BIOMARKER PADA PENYAKIT DIABETES MELLITUS TIPE 2 MENGGUNAKAN *RECURSIVE FEATURE EXTRACTION*

Disetujui Oleh:

Pembimbing Utama	Pembimbing Pertama
Dr. Eng. Armin Lawi, M.Eng.	**Sri Astuti Thamrin, S.Si, M.Stat.PhD**
NIP. 197204231995121001	**NIP. 19740713199903200**

Pada Tanggal : 19 Desember 2018

HALAMAN PENGESAHAN

Skripsi ini diajukan oleh :

Nama	: Nur Nilamyani
NIM	: H13114514
Program Studi	: Ilmu Komputer
Judul Skripsi	: Identifikasi Kandidat Biomarker Pada Penyakit Diabetes Mellitus Tipe 2 Menggunakan *Recursive Feature Extraction*

Telah berhasil dipertahankan dihadapan dewan penguji dan diterima sebagai bagian persyaratan yang diperlukan untuk memperoleh gelar Sarjana Komputer pada Program Studi Ilmu Komputer Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Hasanuddin.

DEWAN PENGUJI

		Tanda Tangan
1. Ketua	: Dr. Eng. Armin Lawi, M.Eng.	(.....)
2. Sekretaris	: Sri Astuti Thamrin, S.Si, M.Stat.PhD	(.....)
3. Anggota	: Dr. Loeky Haryanto, MS., M.Sc	(.....)
4. Anggota	: Dra.Nasrah Sirajang, M.Si	(.....)

Ditetapkan di : Makassar

Tanggal : 19 Desember 2018

iv

KATA PENGANTAR

Segala puji penulis panjatkan kehadirat Allah SWT, yang senantiasa melimpahkan rahmat dan karunia-Nya. Shalawat dan salam senantiasa penulis kirimkan kepada Baginda Rasulullah SAW, yang telah mengajarkan kebenaran dan membimbing umat – umatnya ke arah yang benar. Rasa syukur yang tak terkira atas segala nikmat yang telah diberikan terutama nikmat kesehatan, kesempatan dan kemudahan yang dikaruiniakan kepada penulis dalam menyelesaikan tugas akhir yang berjudul **Identifikasi Kandidat Biomarker Pada Penyakit Diabetes Mellitus Tipe 2 Menggunakan *Recursive Feature Extraction.***

Penyusunan tugas akhir ini tentunya tidak lepas dari bantuan berbagai pihak baik moril maupun materil. Oleh karena itu, penulis menyampaikan ucapan terima kasih yang tulus dan penghargaan yang tak terhingga kepada kedua orang tua saya tercinta bapak **Agussalim Lambong** dan ibu **A. Suryanti** atas segala dukungan, doa, restu, nasehat dan motivasi yang tak henti-hentinya mereka berikan kepada penulis untuk menggapai cita-cita sehingga penulis dapat menyelesaikan pendidikan di perguruan tinggi. Juga kepada adik - adik penulis, **Aka, Kiki, Ivan , Yayan** dan **Ame'** serta untuk seluruh keluarga besar, terima kasih atas doa dan motivasinya.

Penghargaan yang tulus dan ucapan terima kasih dengan penuh keikhlasan juga penulis ucapkan kepada :

1. Ibu **Rektor Universitas Hasanuddin** beserta jajarannya, Bapak **Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam beserta jajarannya**, dan seluruh pihak birokrasi atas pengetahuan dan kemudahan-kemudahan yang diberikan, baik dalam bidang akademik maupun bidang kemahasiswaan.
2. Bapak Prof. **Dr. Amir Kamal Amir, M.Sc**., selaku Ketua Jurusan matika, dan Bapak **Dr. Amran, S.Si., M.Si.**, selaku Sekretaris Jurusan, Bapak **Dr. Diaraya, M.Ak.,** selaku Kepala Program Studi Ilmu puter yang telah memberikan banyak bantuan selama penulis menjalani

pendidikan. Terima kasih juga untuk segenap jajaran Pegawai Akademik Jurusan Matematika atas bantuaannya dalam pengurusan akademik selama ini.

3. Bapak **Dr. Eng. Armin Lawi, M.Eng**, selaku pembimbing utama, atas segala ilmu, nasehat, dan kesabaran dalam membimbing penulis serta meluangkan waktu di sela-sela rutinitas yang begitu padat hingga skripsi ini dirampungkan, dan Ibu **Sri Astuti Thamrin, S.Si,M.Stat.PhD**, selaku pembimbing pertama, untuk segala ilmu, nasehat, dan kesabaran dalam membimbing dan mengarahkan penulis, serta bersedia meluangkan waktunya untuk mendampingi penulis sejak awal penyusunan hingga akhir perampungan skripsi ini.
4. Bapak **Dr. Loeky Haryanto, MS. M.Sc** dan ibu **Dra.Nasrah Sirajang, M.Si** selaku tim penguji untuk segala ilmu, nasehat, saran dan motivasi yang diberikan kepada penulis mulai dari perkuliahan hingga penyusunan skripsi ini.
5. **Kak Ich** sebagai partner *sharing everything* yang sudah mendukung dan mau meminjamkan komputer untuk pengolahan data dan juga membantu selama proses penyelesaian skripsi ini
6. Saudara-saudara **Keluarga Cemara** (**Nurul, Ola, Ayu, Yaumil dan Fuad**) yang telah menemani penulis selama perkuliahan, yang telah meluangkan waktu dan berbagi suka-duka dan kebersamaan selama menuntut ilmu. Serta **Qubers Family** (**Abang, Bellong, Eppe', Calomeng, Mpeng**) makhluk-makhluk astral yang selalu mendukung dan memberikan masukan sejak zaman baheula meskipun sekarang sudah jarang ketemu.
7. Teman-teman seperjuangan **Ilmu Komputer 2014 (Fuad, Jo, Luki, Dilla, Danti, Firda, Yayu, Nuhi, Aspar, Syam, Ica, Nanda, Darul, Nadya, Khalil, dll)** yang membantu dan memberi support penulis dalam penyusunan skripsi ini.

Hilal yang pertamakali mengenalkan bidang Bionformatika di saat ngungan mencari judul tugas akhir. **Kak Supri**, **kak Edy**, **kak Yudha**,

dan **kak Octa** yang sudah memberikan masukan selama proses perkuliahan dan juga kebersamaannya di lab.

9. Adik-Adik **Ilmu Komputer 2015, 2016, 2017,** dan **2018**.
10. Rekan-rekan **KKN UNHAS Gelombang 96 Jepang** yang telah menjadi keluarga baru selama KKN dan menjadikan KKN sebagai momen yang membahagiakan.
11. Semua pihak yang telah banyak berpartisipasi, baik secara langsung maupun tidak langsung dalam penyusunan skripsi ini yang tak sempat penulis sebutkan satu per satu.

Semoga segala bantuan yang dengan tulus ditujukan kepada penulis mendapatkan balasan dari Allah SWT. Mudah-mudahan tulisan ini memberikan manfaat kepada semua pihak yang membutuhkan dan terutama untuk penulis.

Makassar, 19 Desember 2018

Nur Nilamyani

PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR UNTUK KEPENTINGAN AKADEMIK

Sebagai civitas akademik Universitas Hasanuddin saya yang bertanda tangan di bawah ini:

Nama : Nur Nilamyani
NIM : H 131 14 514
Program Studi : Ilmu Komputer
Departemen : Matematika
Fakultas : Matematika dan Ilmu Pengetahuan Alam
Jenis Karya : Skripsi

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Hasanuddin **Hak Prediktor Royalti Non-eksklusif (*Non-exclusive Royalty- Free Right*)** atas tugas akhir saya yang berjudu:

"Identifikasi Kandidat Biomarker Pada Penyakit Diabetes Mellitus Tipe 2 Menggunakan *Recursive Feature Extraction*"

Beserta perangkat yang ada (jika diperlukan). Terkait dengan hal di atas, maka pihak universitas berhak menyimpan, mengalih-media/format-kan, mengelola dalam bentuk pangkalan data (*database*), merawat, dan memublikasikan tugas akhir saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

Dibuat di Makassar pada tanggal, 19 Desember 2018
Yang menyatakan,

myani

ABSTRAK

Ekspresi gen berbasis microarray dapat digunakan untuk mengidentifikasi gen, yang ekspresinya berubah sebagai respons terhadap patogen atau organisme lain dengan membandingkan ekspresi gen pada sel atau jaringan yang terinfeksi dengan yang tidak terinfeksi. Diabetes Melitus tipe 2 adalah kelainan metabolisme yang ditandai dengan kenaikan gula darah akibat penurunan sekresi insulin oleh sel beta pankreas atau gangguan fungsi insulin (resistensi insulin). Jumlah kejadian diabetes mellitus di Indonesia mencapai 10 juta dan 53% dari pasien tidak menyadari bahwa mereka terinfeksi dan 90% kasus diabetes dari seluruh dunia adalah diabetes tipe 2. Oleh karena itu, dalam penelitian ini, kami mengidentifikasi kemungkinan biomarker Diabetes Mellitus tipe 2 dengan menggunakan ekstraksi fitur rekursif. Dengan menerapkan ekstraksi fitur rekursif ke data, diperoleh 3 jenis gen sebagai kemungkinan biomarker untuk Diabetes Mellitus tipe 2.

***Kata Kunci**: Biomarker, Diabetes Mellitus Tipe 2, DNA Microarray, Ekstraksi Fitur, Fitur Ranking*

ABSTRACT

Microarray-based gene expression profiling can be used to identify genes, whose expressions are changed in response to pathogens or other organisms by comparing gene expression in infected to uninfected cells or tissues. Type 2 Diabetes Mellitus is a metabolic disorder that is marked by the rise in blood sugar due to a decrease in insulin secretion by pancreatic beta cells and insulin function or disorder (insulin resistance). The number of incidence of diabetes mellitus in Indonesia reached 10 million and 53% from the patients do not realize that they are infected and 90% case of diabetes from the whole world is type 2 of diabetes. Therefore, in this paper, we identify a probable biomarker of type 2 Diabetes by using the recursive feature extraction. By applying recursive feature extraction to data, we get 3 kinds of genes as a probable biomarker for type 2 Diabetes Mellitus.

***Keywords**: Biomarker, Type 2 Diabetes Melitus, DNA Microarray, Feature Extraction, Feature Ranking.*

DAFTAR ISI

DAFTAR GAMBAR

DAFTAR TABEL

BAB I

PENDAHULUAN

1.1. LATAR BELAKANG

Diabetes Mellitus (DM) tipe 2 adalah penyakit gangguan metabolik yang di tandai oleh kenaikan gula darah akibat penurunan sekresi insulin oleh sel beta pankreas atau ganguan fungsi insulin (resistensi insulin) sehingga tubuh tidak mampu merespon insulin secara normal. Diabetes tipe ini tidak menunjukkan gejala yang dapat dilihat secara langsung sehingga sulit untuk mendeteksi penyakit tersebut. Jumlah penderita DM tipe 2 terus meningkat. Menurut hasil penelitian dari International Diabetes Federation (IDF) tahun 2015, ada 415 juta orang di seluruh dunia menderita diabetes dimana proporsi kejadian DM tipe 2 adalah 95% dari populasi dunia yang menderita diabetes mellitus dan sekitar 10 jutanya terdapat di Indonesia. Lebih dari 60 persen pengidap diabetes tidak sadar kalau terkena diabetes sehingga sering kali penderita DM tipe 2 terdiagnosis setelah terjadi komplikasi. Terlambatnya penanganan terhadap pasien DM tipe 2 dapat berakibat fatal [1].

Hingga saat ini belum ada penanganan medis yang diketahui dapat menyembuhkan DM tipe 2 secara permanen. Tindakan pengobatan pada pasien hanya berfungsi untuk menjaga agar kadar glukosa darah berada pada level senormal mungkin dan mengontrol gejala yang muncul dikemudian hari bukan untuk menyembuhkan penyakit tersebut secara permanen [2]. Oleh sebab itu, penelitian mengenai penanganan yang tepat untuk penderita DM tipe 2 terus dilakukan salah satunya adalah dengan mengidentifikasi biomarker dari penyakit DM tipe 2. Biomarker dapat membantu pendeteksian dini terhadap suatu penyakit sehingga dapat lakukan penanganan yang tepat terhadap penyakit tersebut. Identifikasi marker dari penyakit DM tipe 2 terus berkembang salah satunya adalah gan menggunakan teknologi DNA microarray.

Teknologi DNA microarray digunakan untuk menentukan tingkat ekspresi ribuan gen yang dilakukan dalam sekali percobaan, dan secara simultan memantau proses biologi yang sedang berlangsung [3]. Identifikasi biomarker menggunakan ekspresi gen dapat memberikan hasil yang sangat penting karena dapat membantu pengembangan pengobatan personal (*personalized medicine*) dan juga membantu para peneliti untuk menemukan penanganan yang tepat terhadap penyakit serius, termasuk DM tipe 2.

Beberapa penelitian yang menggunakan data microarray juga telah dilakukan untuk menemukan biomarker di beberapa jenis penyakit kompleks seperti kanker dan DM tipe 2 seperti yang dilakukan oleh dilakukan oleh Zhang dkk [4] untuk mengindentifikasi biomarker DM tipe 2 menggunakan *Discriminative Area Of Functional Activity.* Penelitian lainnya dilakukan oleh Chakraborty dkk [5] yang melakukan penelitian terhadap biomarker dari penyakit kanker yang menggunakan data microarray dengan menggunakan metode *Feature Selection* dan *Semi Supervised Learning*. Chen dkk [6] juga melakukan penelitian yang serupa yaitu mengidentifikasi biomarker dari kanker tetapi menggunakan *Metode Network-Constrained Support Vector Machine*. Penelitian lainnya untuk menemukan biomarker dari DM tipe 2 menggunakan algoritma PreDx DRS [7] .

Kekurangan dari data microarray sendiri adalah jumlah dimensi yang besar dikarenakan jumlah sampel yang lebih sedikit dibanding jumlah gen, sehingga dibutuhkan metode seleksi fitur untuk memilih gen informatif. Dalam penelitian ini untuk proses seleksi fitur yang akan digunakan adalah *Recursive Feature Extraction*. Berdasarkan penelitian yang dilakukan oleh Mishra dkk [8] metode ini menghasilkan nilai akurasi yang mencapai 100% untuk pendeteksian kandidat biomarker pada data kanker payudara.

1.2. RUMUSAN MASALAH

Berdasarkan pada latar belakang yang telah di uraikan sebelumnya, maka rumusan masalah yang akan di bahas pada penelitian ini adalah

1) Bagaimana cara mengolah data microarray untuk memprediksi kandidat biomarker pada penyakit DM tipe 2?
2) Bagaimana cara mengatasi masalah dimensional pada data microarray?
3) Bagaimana performansi metode seleksi fitur yang digunakan untuk menganalisis data microarray untuk penyakit DM tipe 2?

1.3. TUJUAN

Dengan memperhatikan latar belakang dan rumusan masalah yang telah dikemukakan sebelumnya maka tujuan dari penelitian ini adalah

1) Menghasilkan prediksi biomarker yang tepat untuk penyakit DM tipe 2.
2) Mengatasi masalah dimensional pada data microarray dengan melakukan seleksi fitur.
3) Memperoleh performansi metode seleksi fitur untuk data microarray DM tipe 2.

1.4. MANFAAT

Hasil dari penelitian ini diharapkan dapat berguna untuk hal-hal berikut:

1) Pemanfaatan ilmu komputer dalam bidang biologi dan medis dapat dimanfaatkan oleh peneliti untuk mengurangi biaya penelitian konvensional
2) Biomarker yang diperoleh diharapkan dapat digunakan untuk mengembangkan pengobatan personal (*personalized medicine*) untuk penyakit DM tipe 2.

1.5. BATASAN MASALAH

1) Penelitian ini hanya akan membahas tentang pengidentifikasian biomarker dari penyakit DM tipe 2 dari data GSE18732.
2) Tingkat keakuratan hasil yang diperoleh hanya berlaku untuk dataset yang digunakan dalam penelitian ini.

1.6. ORGANISASI SKRIPSI

Sistematika penulisan skripsi ini adalah sebagai berikut :

BAB I : Pendahuluan

Bab ini membahas mengenai latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan manfaat penulisan, serta sistematika penulisan.

BAB II : Tinjauan Pustaka

Bab ini membahas mengenai landasan teori dan konsep dasar yang mendasari pokok permasalahan dalam tulisan ini.

BAB III : Metodologi Penelitian

Bab ini berisi waktu dan tempat penelitian, tahapan penelitian, rancangan sistem, sumber data, instrumen penelitian dan jadwal penelitian

BAB IV : Hasil dan Pembahasan

Bab ini akan membahas tentang hasil penelitian dan pembahasannya.

BAB V : Kesimpulan dan Saran

Berisikan tentang kesimpulan dari penelitian dan saran yang berguna untuk penelitian lebih lanjut dari skripsi ini.

BAB II
TINJAUAN PUSTAKA

2.1. LANDASAN TEORI

2.1.1. Biomarker

Istilah "biomarker" dari "penanda biologis", mengacu pada subkategori tanda-tanda medis yang luas yaitu indikasi objektif keadaan medis yang diamati dari luar pasien yang dapat diukur secara akurat dan reproduktif. Tanda-tanda medis kontras dengan gejala medis, yang terbatas pada indikasi kesehatan atau penyakit yang dirasakan oleh pasien itu sendiri. Biomarker atau biologikal markers merupakan molekul penanda yang khas bagi sel, yang dapat digunakan untuk mendiagnosa suatu penyakit dan terapi target molekul penyebab penyakit tertentu [9]. Pendapat lain mengatakan biomarker berfungsinya sebagai sistem peringatan dini terhadap gangguan yang dapat berupa potensi timbulnya suatu penyakit akibat tekanan yang dialami oleh organisme [10]. Pada tahun 1998, National Institute of Health Biomarker Definitions Working Group mendefinisikan biomarker sebagai karakteristik yang diukur dan dievaluasi secara objektif sebagai indikator proses biologis normal, proses patogenik, atau tanggapan farmakologis terhadap intervensi terapeutik. Sebuah usaha patungan untuk keamanan kimia, Program Internasional untuk Keselamatan Kimia, yang dipimpin oleh Organisasi Kesehatan Dunia (WHO) dan dalam berkoordinasi dengan Perserikatan Bangsa-Bangsa dan Organisasi Perburuhan Internasional, telah mendefinisikan biomarker sebagai substansi, struktur, atau proses apa pun dapat diukur dalam tubuh atau produknya dan mempengaruhi atau memprediksi kejadian hasil atau penyakit. Definisi yang lebih luas memperhitungkan bukan hanya kejadian dan hasil dari penyakit, tetapi juga dampak perawatan, intervensi, dan bahkan paparan lingkungan yang tidak diinginkan, seperti bahan kimia atau nutrisi. Dalam laporan mereka tentang validitas biomarker dalam penilaian risiko [illegible]an, WHO telah menyatakan bahwa definisi sejati biomarker mencakup [illegible]emua pengukuran yang mencerminkan interaksi antara sistem biologis [illegible]nsi bahaya, yang mungkin bersifat kimiawi, fisik, atau biologis. Respons

yang diukur mungkin fungsional dan fisiologis, biokimia pada tingkat sel, atau interaksi molekuler [11]. Kelebihan diagnosa penyakit berdasarkan biomarker yaitu diagnosa dapat dilakukan sejak dini dan hasil yang diperoleh lebih akurat [9].

2.1.2. Diabetes Mellitus Tipe 2

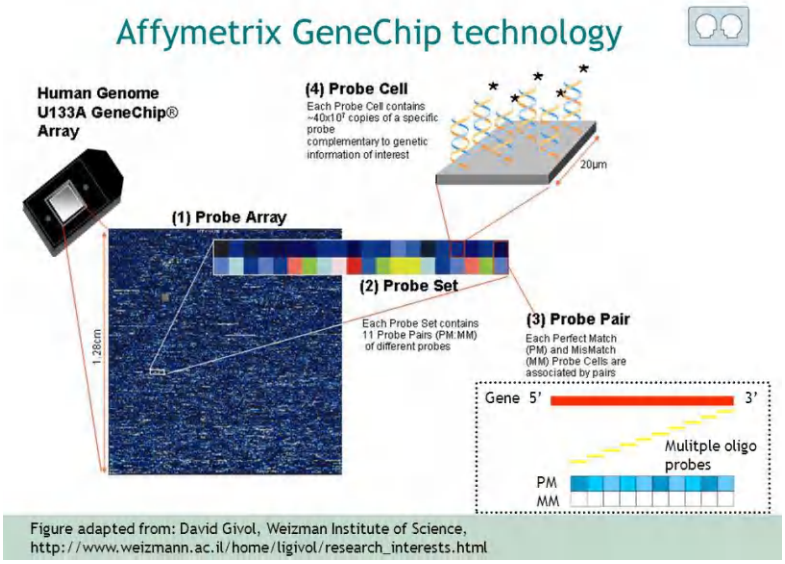
Diabetes Melitus (DM) tipe 2 adalah penyakit yang ditandai dengan terjadinya hiperglikemia dan gangguan metabolisme karbohidrat, lemak, dan protein yang dihubungkan dengan kekurangan secara absolut kerja insulin atau sekresi insulin. Menurut klasifikasi etiologi American Diabetes Association DM dibagi dalam beberapa jenis yaitu: DM tipe 1 atau Insulin Dependent DM/IDDM, DM tipe 2 atau Insulin Non-dependent DM/NIDDM, dan DM Gestasional. Diantara semua jenis DM, proporsi kejadian DM tipe 2 adalah 95% dari populasi dunia yang menderita DM [12].

Pada penderita DM tipe 2 ini terjadi hiperinsulinemia tetapi insulin tidak bisa membawa glukosa masuk ke dalam jaringan karena terjadi resistensi insulin yang merupakan turunnya kemampuan insulin untuk merangsang pengambilan glukosa oleh jaringan perifer dan untuk menghambat produksi glukosa oleh hati. Oleh karena terjadinya resistensi insulin (reseptor insulin sudah tidak aktif karena dianggap kadarnya masih tinggi dalam darah) akan mengakibatkan defisiensi relatif insulin. Hal tersebut dapat mengakibatkan berkurangnya sekresi insulin pada adanya glukosa bersama bahan sekresi insulin lain sehingga sel beta pankreas akan mengalami desensitisasi terhadap adanya glukosa. DM tipe ini terjadi perlahan-lahan karena itu gejalanya asimtomatik. Adanya resistensi yang terjadi perlahan-lahan akan mengakibatkan sensitivitas reseptor akan glukosa berkurang. DM tipe ini sering terdiagnosis setelah terjadi komplikasi [13].

2.1.3. DNA Micro Array

DNA microarray adalah chip dengan probe DNA mikroskopis tertanam di permukaan. Microarray dapat memeriksa ribuan gen dalam waktu yang sama serta u mengidetifikasi gen yang terlihat pada sel yang berbeda dan mencari antara masing-masing gen [14].

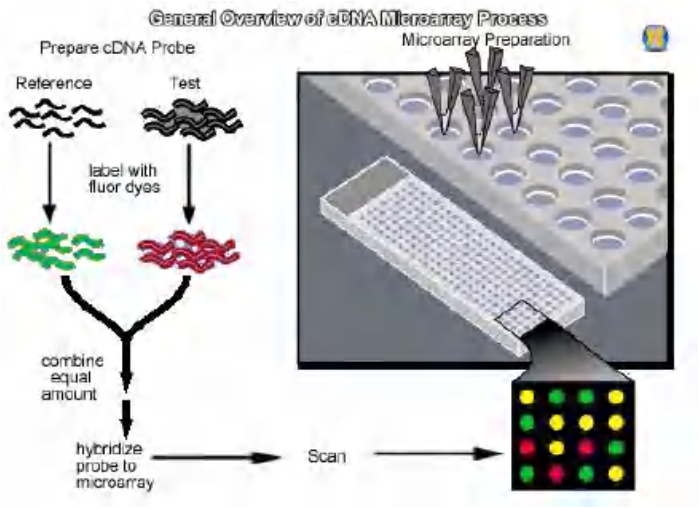
Microarray DNA oligonukleotida lebih lanjut dapat menjadi dua subkelompok: *long oligonucleotide arrays*, yang probenya terdiri dari 60-mer atau 50-mer sekuens DNA (contoh *Illumina Beadarray*), dan *short oligonucleotide arrays* yang menggunakan 25-mer (misal *Affymetrix GeneChip*) atau 30-mer dari desain urutan probe. Penelitian ini berkonsentrasi hanya pada fokus pada *Affymetrix* teknologi saja. Gambar 1 merupakan ilustrasi dari kepingan gen *Affymetrix* (*Affymetrix GeneChip*). Kepingan gen tersebut menggunakan teknologi seperti keping silikon komputer. Material silikon *affymetrix* dilindungi dengan menutup dan menerapkan proses *photolithographic* untuk mengendalikan sintesis oligonukleotida pada permukaan kaca/plastik. Perancangan probe menggunakan 25-mer gen spesifik oligonukleotida, secara lebih khusus probe set dibentuk dengan 11 sampai 20 pasang probe berbeda yang digunakan untuk mencocokan gen-gen berbeda. Desain pasangan probe terbagi menjadi 2 jenis yaitu pasangan probe yang tidak sesuai atau biasa disebut *mismatch* (MM) dan pasangan probe yang sesuai atau biasa disebut *perfect match* (PM) probe. Probe MM digunakan untuk mengendalikan ikatan-ikatan non-spesifik selama hibridisasi [14].



Gambar 1. Affymetrix GeneChip

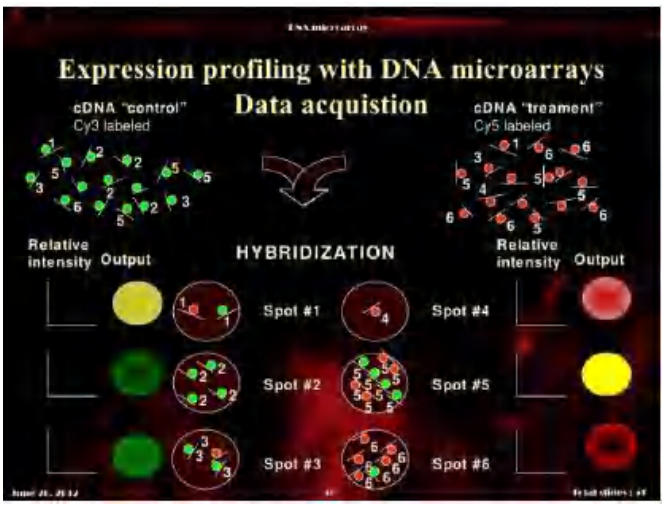
eknologi ini membantu para peneliti dalam mempelajari berbagai terutama kanker. Oleh karena itu, teknologi microarray dapat membantu n diagnosis, memonitor dan memprediksi suatu penyakit. Metode ini

menggunakan alat berupa *slide* yang terbuat dari kaca dan terdiri dari ribuan bahkan puluhan ribu blok [15].



Gambar 2. Cara Kerja Teknologi DNA Microarray

Sebenarnya prinsip kerja dari microarray seperti yang terlihat pada ilustrasi Gambar 2 adalah mengukur jumlah hibridisasi mRNA pada cDNA dalam chip tersebut. Pada umumnya analisis dengan menggunakan microarray menggunakan dua sampel yang berbeda, misalnya sel kulit normal dengan sel kulit berpenyakit. Kedua sampel tersebut diisolasi mRNA-nya dan kemudian diletakkan dalam keping microarray. Kemudian keping tersebut diberi penanda radioaktif untuk menghasilkan warna *fluorosens* setelah dilakukan *scanner* yang terhubung dengan komputer. Kemudian komputer akan menganalisis kedua sampel tersebut berdasarkan pola warna yang ada [15].



Gambar 3. Hasil Interpretasi Teknologi DNA Microarray

Hasil interpretasi dari teknologi DNA microarray ditampilkan pada Gambar 3, yaitu jika ekspresi gen tertentu lebih tinggi maka akan tampak merah. Sebaliknya, jika ekspresi dalam sampel percobaan lebih rendah, maka akan tampak hijau. Akhirnya, jika ada dua ekspresi yang sama dalam sebuah sampel, maka akan muncul kuning. Sebuah titik hitam menunjukkan bahwa tidak ada cDNA yang terikat pada DNA pada gen yang terletak di tempat tersebut. Hal ini menunjukkan bahwa gen tidak aktif (semua gen dalam percobaan aktif) [15,18].

Data yang dihasilkan dari microarray merupakan jenis data yang dipakai dalam bioinformatika**.** Karakteristik microarray data adalah jumlah data sedikit dan jumlah fitur yang sangat banyak. Data ini berisi informasi gen karena itu jumlah fiturnya sangat banyak. Meskipun jenis data tersebut sulit untuk diolah tetapi, hasil yang diperoleh akan sangat berguna misalnya penemuan obat-obatan baru (*drug discovery*) dan penentuan jenis pengobatan untuk penanganan pasien.

2.1.4. Reduksi Dimensi

Kelemahan utama dari penggunaan data microarray adalah masalah dimensionalitas. Permasalahan dimensionalaitas pada data microarray sangat berpengaruh pada waktu komputasi dan tingkat akurasi pada proses klasifikasi rmasalahan dimensional pada data microarray dapat menimbulkan dimensi (*curse of dimensionality*) yang sangat mempengaruhi tingkat alam proses klasifikasi. Hal tersebut dikarenakan besarnya jumlah fitur ak sebanding dengan jumlah sampel yang tersedia. Oleh karena itu

dibutuhkan proses reduksi dimensi pada data microarray agar dapat menghemat waktu komputasi dan juga meningkatkan tingkat akurasi. Salah satu teknik reduksi dimensi adalah seleksi fitur.

Pemilihan gen informatif dalam seleski fitur merupakan bagian penting dalam analisis data microarray. Dalam situasi yang melibatkan ribuan fitur, kesuksesan dari proses seleksi fitur memiliki beberapa manfaat. Pertama, reduksi dimensi dilakukan untuk mengurangi waktu komputasional. Kedua, meningkatkan keakurasian klasifikasi. Ekstraksi dari fitur - fitur atau karakteristik yang dihasilkan dari seleksi fitur sangat membantu proses identifikasi dan memonitor target penyakit.

2.1.5. Preprocessing

Preprocessing merupakan tahap untuk penyesuaian serta konversi data *affybatch* ke dalam bentuk *expression set*. Tidak hanya melakukan konversi data *affybatch*, dalam tahap preprocessing dilakukan proses *background correction*, *normalization* dan *summarization*. Istilah *background correction* mengacu pada penyesuaian berbagai macam metode, serta yang harus dilakukan meliput:

1) Memperbaiki *background noise* dan efek pengolahan
2) Menyesuaikan hibridisasi pengikat DNA non-spesifik pada array
3) Menyesuaikan estimasi ekspresi sehingga bersekala tepat atau berhubungan linear

Background correction secara umum mengacu pada pengertian pertama. Tahap setelah *background correction* yaitu *normalization*. *Normalization* adalah proses penghapusan akhiran non-biologis yang tidak diinginkan diantara keping dalam eksperimen microarray. Tahap setelah hal tersebut ialah tahap dilakukan *summarization*, yaitu untuk mengukur ekspresi gen. [17,19].

2.1.6. Penyaringan (*Filtering*)

Filtering data microarray adalah proses pemilihan subset dari probe yang tersedia untuk pengecualian atau penyertaan dalam analisis. Fitur *filtering* dapat kkan beberapa varian kecil atau ketimpangan data secara konsisten di ampel, hal ini dapat berguna untuk analisis selanjutnya [20].

2.1.7. Fitur Ranking

Terdapat perbedaan yang cukup besar antara jumlah sampel dan fitur (gen) pada data microarray. Oleh karena itu, penggunaan pendekatan seleksi fitur untuk untuk perbedaan fitur sebanyak itu terkadang tidak memberikan hasil yang cukup baik maka dari itu sebelum proses seleksi fitur, terlebih dahulu akan dilakukan proses fitur ranking dengan cara memberikan skor pada masing-masing fitur yang bertujuan untuk mengeliminasi fitur yang tidak relevan menggunakan beberapa metode fitur ranking, yaitu:

A. Chi-Square

Uji Chi-Square merupakan uji statistik non-parametrik yang paling banyak digunakan dalam penelitian bidang kesehatan masyarakat, karena uji ini memiliki kemampuan membandingkan dua kelompok atau lebih pada data-data yang telah dikategorisasikan. Meski demikian, uji chi-square dapat pula dipakai pada pengujian satu kelompok dan berskala interval/rasio [21].

Distribusi Chi-square (χ^2) adalah distribusi probabilitas teoritis yang asimetrik dan kontinyu. Nilai sebuah χ^2 selalu positif antara 0 sampai dengan ∞ (tak hingga) atau $0 \leq \chi^2 \leq \infty$, tidak seperti distribusi normal atau distribusi t yang dapat bernilai negatif. Nilai statistik χ^2 dihitung dengan rumus sebagai berikut [22] :

$$\chi^2 = \sum_{i=1}^{b} \sum_{j=1}^{m} \frac{(o_{ij} - e_{ij})^2}{e_{ij}}, \quad (1)$$

dimana b = baris, c = kolom, o_{ij} = banyaknya frekuensi yang diobservasi dan e_{ij}= banyaknya frekuensi yang diharapkan. Adapun langkah – langkah pengujian Chi –Square yaitu:

1. Hipotesis

   H_0 : tidak ada hubungan yang signifikan antara fitur data set

   H_α : ada hubungan yang signifikan

2. Menentukan taraf nyata yaiu α = 0,05

   Menghitung o_{ij} dan e_{ij}. Untuk menghitung frekuensi harapan e_{ij}, akan dibuat tabel kontingensi. Berdasarkan Tabel 1 tersebut $A_1, A_2, \ldots, A_m$ adalah fitur-fitur yang ada pada satu fitur.

4. Mencari nilai frekwensi yang diharapkan (e_{ij})

$$e_{ij} = \frac{(n_i)(n_j)}{n}, \quad (2)$$

dimana i = 1,2,…,b

j = 1,2,…,m

5. Menghitung nilai χ^2 berdistribusi *chi-square* dengan derajat kebebasan yaitu $(b-1)(m-1)$
6. Menarik kesimpulan dengan kriteria keputusan :

Tolak H_0 jika $\chi^2 \geq \chi^2_{tabel(\alpha,v)}$.

Tabel 1. Bentuk Umum Tabel Kontingensi

	A_1	A_2	...	A_m	Jumlah
Label kelas (L_1) = 1	o_{11}	o_{12}	...	o_{1m}	$n_{1.}$
Label kelas (L_2)= -1	o_{21}	o_{22}	...	o_{2m}	$n_{2.}$
Jumlah	$n_{.1}$	$n_{.2}$	...	$n_{.m}$	N

Uji chi-squared digunakan untuk menentukan hubungan antara kedua variabel. Hasil yang diperoleh merupakan bobot dari atribut diskrit berdasarkan chi-square tes. Sedangkan untuk melihat seberapa kuat hubungan dari kedua variabel dari hasil uji chi-square digunakan Creamer's V. Hasil yang nantinya diperoleh dari fungsi chi.squared dari *package FSelector* merupakan nilai dari *Creamer's V* dimana ukuran kekuatan korelasi antara varibelnya berada pada rentang 0 – 1.

B. *Information Gain*

Dalam *machine learning*, *Information Gain* dapat digunakan untuk meranking fitur-fitur dalam data. Biasanya, fitur dengan nilai *Information Gain* yang tinggi harus diberikat peringkat yang lebih tinggi dibandingkan fitur lain karena memiliki pengaruh yang besar dalam mengklasifikasi data [23]. Nilai *Information Gain* diperoleh dari nilai entropi sebelum pemisahan dikurangi dengan nilai entropi setelah pemisahan. *Information Gain* adalah unan entropi yang diharapkan yang merupakan akibat dari partisi el-sampel berdasarkan atribut yang diberikan. Dalam teori informasi, pi secara umum merupakan nilai yang mengukur tingkat kemurnian

(*impurity)* dalam sebuah kelompok sampel. Entropi yang akan digunakan adalah *Shannon Entropy* [24]

$$H(S) = -\sum_{i=1}^{m} p_i log_2(p_i) \,, \tag{3}$$

dimana $H(S)$ adalah entropi dan p_i adalah probabilitas kelas i dalam kelompok data S. Kemudian untuk *Information Gain* didefinisikan dengan

$$G(S,A) = H(S) - \sum_{i\,\epsilon\,A} \frac{|S_i|}{|S|} H(S_i) \,, \tag{4}$$

dimana $H(S)$ adalah entropi dari dataset yang ditetapkan dan $H(S_{i)}$ adalah entropi dari subset i yang dihasilkan dengan mempartisi S berdasarkan fitur A.

C. Hutan Acak Utama (*Random Forest Importance*)

Algoritma *random forest* merupakan salah satu metode *machine learning* yang dapat menyelesaikan masalah klasifikasi atau regresi. Prinsip utama dari random forest adalah mengkombinasikan beberapa pohon keputusan (*decision tree*) yang dibangun menggunakan beberapa sampel *bootstrap* yang diambil dari sampel utama L dan memilih secara acak setiap node yang merupakan subset dari variabel penjelas X. Algoritma ini dapat menyediakan variabel dependen pada sejumlah kelas pohon yang dibentuk. Skema ini pertama kali dicetuskan oleh Leo Breiman pada tahun 2000 untuk membangun prediktor dengan sekumpulan pohon keputusan (*decision tree*) yang berkembang secara acak pada subruang data [25].

Pada umumnya algoritma *random forest* menggunakan metode *Classification And Regression Tree* (CART) untuk membentuk aturan prediktor untuk membangun *decision tree* akan tetapi dalam penelitian ini untuk proses fitur ranking digunakan algoritma OneR *classifier* untuk membentuk aturan prediktor untuk membangun pohon keputusan. OneR *clissifier* adalah algoritma klasifikasi yang menghasilkan aturan yang sama untuk setiap prediktor dalam data, kemudian memilih aturan dengan menggunakan ...iktor yang memiliki total eror terkecil sebagai satu aturan. Untuk ...bangun aturan pada prediktor akan dikonstruksikan tabel frekuensi untuk ... prediktor untuk melawan target.

...un algoritma dari OneR *classifier* adalah sebagai berikut

Untuk setiap atribut A:

Untuk setiap nilai V dari atribut, buat sebuah aturan:

1. Hitung seberapa sering setiap kelas muncul
2. Temukan kelas dengan frekuensi terbanyak, c
3. Buat aturan “jika A=V maka C=c”

Kalkulasikan total eror dari aturan untuk setiap atribut

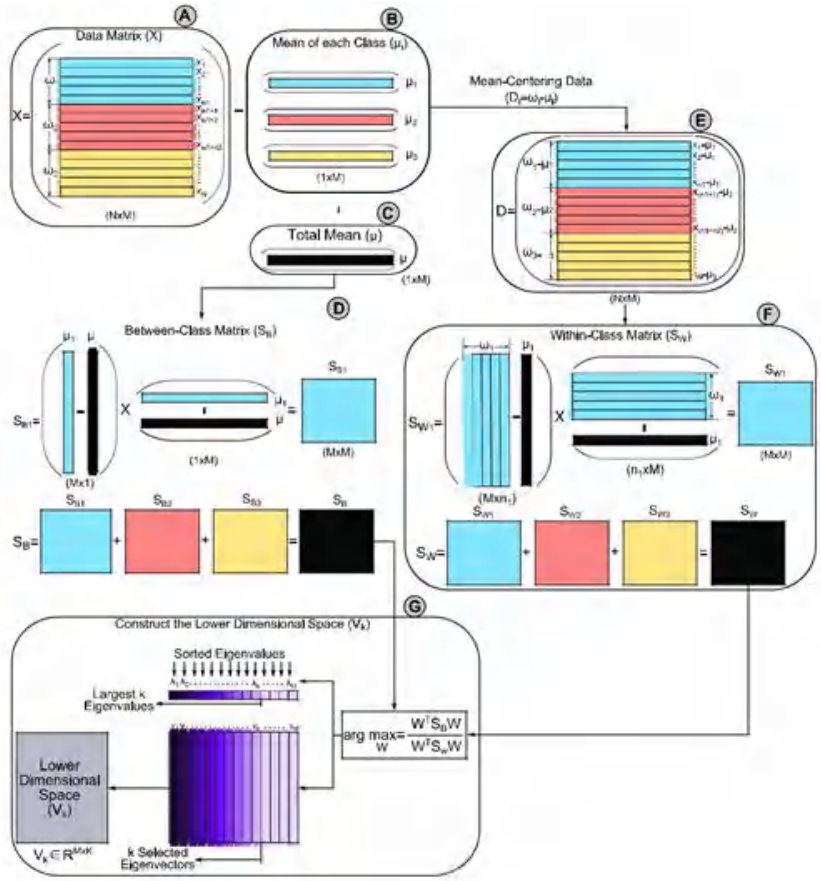
Pilih prediktor dengan total eror terkecil

2.1.8. Seleksi Fitur

Pendekatan seleksi fitur dilakukan untuk memilih gen paling berpengaruh. Dalam penelitian ini akan digunakan beberapa jenis metode seleksi fitur yang bertujuan untuk memperoleh hasil terbaik, yaitu:

A. Analisis Diskriminan Linier

Analisis Diskriminan Linear (LDA) adalah sebuah metode untuk proses reduksi dimensionalitas yang digunakan dalam ilmu statistika, pengenalan pola, pembelajaran mesin, dan bioinformatika untuk mencari kombinasi linear fitur yang menjadi ciri atau yang memisahkan dua atau beberapa objek. LDA juga berfungsi meminimumkan jarak di dalam kelas objek yang sama. LDA dikembangkan untuk mentransformasi fitur ke ruang dimensi yang lebih rendah dengan memaksimalkan rasio varians antar kelas ke varians dalam kelas dengan menjamin pemisah kelas maksimum [27].



Gambar 4. Visualisasi Cara Kerja LDA

Gambar 4 menjelaskan 8 langkah (A, B, C, D, E, F, G) dari proses yang dibutuhkan dalam LDA proses utama terdapat pada langkah D, E, dan F. Berikut adalah penjelesan dari proses utama tersebut.

1. Langkah pertama adalah mengkalkulasikan pemisah antar kelas berbeda yang disebut *between-class variance* (S_B) atau *between-class matrix. Between-class variance* dari kelas ke-i (S_{Bi}) merepresentasikan jarak antara rata-rata kelas ke-i (μ_i) dan total rata-rata (μ). LDA memaksimalkan jarak pemisah antar kelas. Untuk menjelaskan proses kalkulasi dari S_B diasumsikan sebagai berikut. Misalkan diberikan matriks $X = \{x_1, x_2, \dots, x_N\}$, dimana x_i adalah sampel ke-i dan N adalah total sampel. Setiap sampel direpresentasikan dengan M fitur M). Atau dengan kata lain setiap sampel direpresentasikan dalam dimensi umsikan data matriks di partisi ke dalam 3 kelas $X = [w_1, w_2, w_3]$. Setiap memiliki 5 sampel $(n_1 = n_2 = n_3 = 5)$, dimana n_i merepresentasikan

nomor sampel dari kelas ke-i. Total jumlah sampel N dikalkulasikan dengan $\sum_{i=1}^{3} n_i$. Untuk kalkulasi S_B pemisah jarak antar kelas berbeda yang dinotasikan dengan $(m_i - m)$ dapat dihitung dengan:

$$(m_i - m)^2 = (W^T \mu_i - W^T \mu)^2 \qquad (5)$$
$$= W^T (\mu_i - \mu)(\mu_i - \mu)^T W \quad ,$$

dimana $m_i = W^T \mu_i$ adalah proyeksi rata-rata kelas ke-i, $m = W^T \mu$ adalah proyeksi total rata-rata dari seluruh kelas, W adalah transformasi matriks LDA, $\mu_i(1 \times M)$ merepresentasikan rata-rata kelas ke-i dapat dihitung dengan persamaan (6) dan $\mu(1 \times M)$ adalah total rata-rata seluruh kelas yang dapat dihitung dengan persamaan (7).

$$\mu_j = \frac{1}{n_j} \sum_{x_i \in w_j} x_i \qquad (6)$$

$$\mu = \frac{1}{N} \sum_{i=1}^{N} x_i \sum_{i=1}^{c} \frac{n_i}{N} \mu_i \quad , \qquad (7)$$

dimana c adalah total kelas yang dimiliki dalam contoh ini c=3.

$(\mu_i - \mu)(\mu_i - \mu)^T$ pada persamaan (5) adalah representasi pemisah jarak antara rata-rata kelas ke-i (μ_i) dan total rata-rata (μ), atau secara sederhananya adalah representasi dari *between-class variance* dari kelas ke-$i(S_{Bi})$. Jika disubtitusikan ke persamaan (5) menjadi:

$$(m_i - m)^2 = W^T S_{Bi} W \qquad (8)$$

Untuk menghitung total *between-class variance* $(S_B = \sum_{i=1}^{c} n_i S_{Bi})$ pada Gambar 1 (Step D) dapat dilihat bagaimana *between-class matrix* dari kelas pertama (S_{Bi}) dihitung dan bagaimana total *between-class matrix* (S_B) dihitung dengan menambahkan seluruh *between-class matrix* dari seluruh kelas.

2. Mengkalkulasikan jarak antara rata-rata dengan sampel untuk setiap kelas, yang disebut *within class variance* (S_W) atau *within-class matriks. Within-class variance* dari kelas ke-i (S_{Wi}) merepresentasikan perbedaan antara rata-rata dan sampel dari kelas tersebut. LDA mencari ruang dimensi terendah yang ...kan untuk meminimalisir perbedaan antarak rata-rata yang ...eksikan (m_i) dan sampel yang diproyeksikan dari setiap kelas $(W^T x_i)$. ...*-class variance* untuk setiap kelas dapat dihitung dengan,

$$S_{Wj} = {d_j}^T \sum_{i=1}^{n} (x_{ij} - \mu_j)(x_{ij} - \mu_j)^T, \quad (9)$$

dimana x_{ij} merepresentasikan sampel ke-i dalam kelas ke-j seperti pada gambar 1 (Step E dan F), dan d_j adalah pusat data dari kelas ke-j. Step (F) pada Gambar 1 mengilustrasikan bagaimana *within-class variance* pada kelas pertama (S_{W_1}) pada contoh dikalkulasikan. Total *withn-class variance* merepresentasikan jumlah dari seluruh *within-class* matriks dari semua kelas, dan dapat dikalkulasikan dengan persamaan.

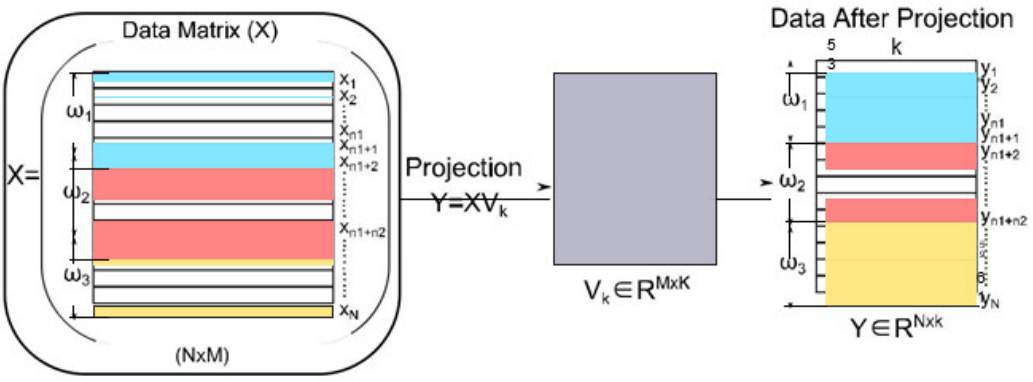
$$\begin{aligned} S_w &= \sum_{i-1}^{3} S_{W_i} \\ &= \sum_{x_i \in \omega_1} (x_i - \mu_1)(x_i - \mu_1)^T \\ &+ \sum_{x_i \in \omega_2} (x_i - \mu_2)(x_i - \mu_2)^T \\ &+ \sum_{x_i \in \omega_3} (x_i - \mu_3)(x_i - \mu_3)^T \end{aligned} \quad (10)$$

3. Mengkonstruksikan ruang dimensi rendah dengan memaksimalkan *between-class variance* (S_B) dan meminimalisirkan *whitin class variance* (S_W). Transformasi matriks (W) menggunakan LDA dapat dihitung dengan persamaan (11) kemudian diformulasikan ulang pada persamaan (12)

$$arg\ max_w \frac{W^T S_B W}{W^T S_W W} \quad (11)$$

$$S_W W = \lambda S_B W, \quad (12)$$

dimana λ adalah nilai eigen pada transformasi matriks (W). Solusi dari permasalahan ini dapat diperoleh dengan menghitung nilai eigen $(\lambda = \{\lambda_1, \lambda_2, \dots, \lambda_M\})$ dan vector eigen $(V = \{v_1, v_2, \dots, v_M\})$ dari $W = S_W^{-1} S_B$.



Gambar 5. Gambaran Hasil Akhir Metode LDA

Gambar 5 adalah ilustrasi dari hasil akhir proses LDA dimana data yang awalnya berdimensi besar direduksi ke dalam bentuk dimensi yang lebih kecil.

B. Hutan Acak (*Random Forest*)

Algoritma *Random Forest* (RF) merupakan salah satu metode *machine learning* yang dapat menyelesaikan masalah klasifikasi atau regresi. Prinsip utama dari RF adalah mengkombinasikan beberapa pohon keputusan yang dibagun menggunakan beberapa sampel *bootstrap* yang diambil dari sampel utama L dan memilih secara acak setiap node yang merupakan subset dari variabel penjelas X. Algoritma ini dapat menyediakan variabel dependen pada sejumlah kelas tree yang dibentuk. Skema ini pertama kali dicetuskan oleh Leo Breiman pada tahun 2000 untuk membangun prediktor dengan sekumpulan pohon keputusan yang berkembang secara acak pada subruang data [28].

Untuk proses seleksi fitur akan digunakan algoritma CART untuk membentuk aturan prediktor untuk membangun pohon keputusan (*decision tree*). CART adalah salah satu metode atau algoritma dari teknik pohon keputusan. CART adalah suatu metode statistik nonparametrik yang dapat menggambarkan hubungan antara variabel respon (variabel dependen) dengan satu atau lebih variabel prediktor (variabel independen). Pembentukan pohon klasifikasi terdiri atas 3 tahap yang memerlukan sampel L sebagai model pembelajaran. Tahap adalah pemilihan pemilah. Setiap pemilahan hanya bergantung pada nilai asal dari satu variabel independen. Tahap kedua adalah penentuan simpul Tahap ketiga adalah penandaan label tiap simpul terminal berdasar mlah anggota kelas terbanyak. Proses pembentukan pohon klasifikasi

berhenti saat terdapat hanya satu pengamatan dalam tiap simpul anak [29]. Berikut ini adalah simulasi contoh algoritma RF:

Misalkan pada Tabel 2 kita memiliki data pasien, dari data tersebut akan dibuat pohon keputusan untuk menarik sebuah kesimpulan apakah pasien berpotensi menderita penyakit jantung atau tidak dengan menggunakan 4 varibel uji yaitu sakit dada, tekanan darah, arteri terjepit dan berat badan .

Tabel 2. Contoh Dataset Utama Pembentukan *Random Forest*

Sakit Dada	Tekanan Darah	Arteri Terjepit	Berat Badan	Penyakit Jantung
Tidak	Tidak	Tidak	190	Tidak
Ya	Ya	Ya	180	Ya
Ya	Ya	Tidak	150	Tidak
Ya	Tidak	Ya	220	Ya
⋮	⋮	⋮	⋮	⋮

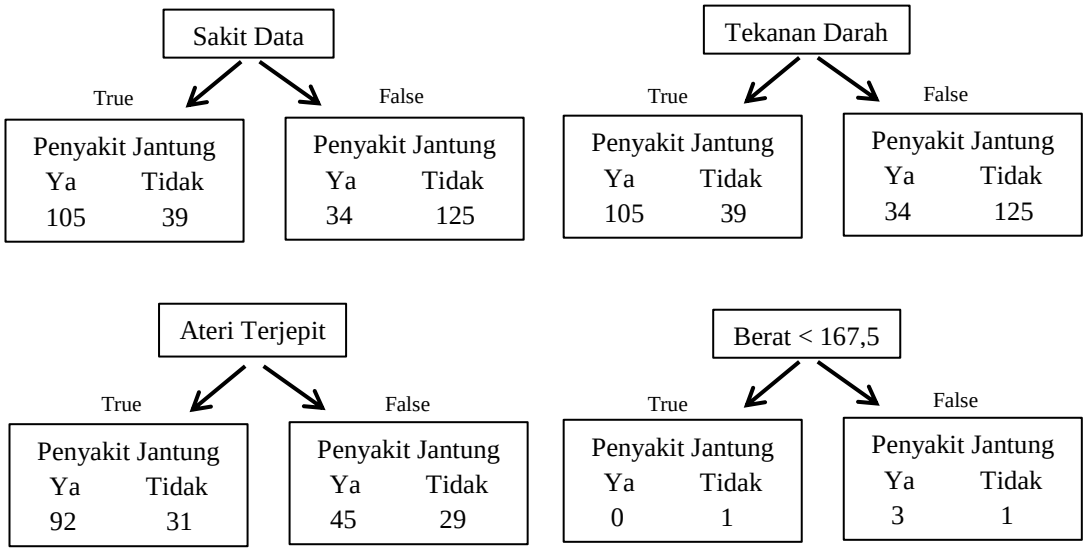
Untuk membetuk pohon keputusan dari contoh data di atas berikut ini merupakan langkah-langkah pembuatannya.

1) Membuat *Bootstrapped* dataset dari dataset utama. Tabel 3 merupakan contoh *Bootstrapped* dataset yang dapat dibentuk berdasarkan pengambilan data secara acak dari dataset utama.

Tabel 3. Contoh *Bootstrapped* Dataset dari Dataset Utama

Sakit Dada	Tekanan Darah	Arteri Terjepit	Berat Badan	Penyakit Jantung
Tidak	Tidak	Tidak	190	Tidak
Ya	Ya	Ya	180	Ya
Ya	Tidak	Ya	220	Ya
Ya	Tidak	Ya	220	Ya
⋮	⋮	⋮	⋮	⋮

Dari contoh *Bootstrapped* dataset di atas dapat dilihat bahwa data pada urutan ke-3 dan ke-4 sama. Hal tersebut bisa saja terjadi mengingat bahwa pengambilan data dilakukan secara acak dan pengambilan data yang sama diperbolehkan. Misalkan Gambar 6 adalah salah satu pohon keputusan yang ntuk dari *Bootstrapped* dataset utama:



Gambar 6. Contoh Pohon dari *Bootstrapped* dataset

2) Pohon keputusan dibentuk berdasarkan *Bootstrapped* data set, hanya saja untuk setiap pembuatan *cutoff* akan digunakan subset variabel (kolom yang diambil secara acak). Untuk contoh ini hanya akan mengambil 2 variabel untuk setiap pembentukan *cutoff*. Misalkan untuk pembentukan sisi kiri pohon dipilih variabel sakit dada dan tekanan darah untuk menentukan variabel mana yang dapat menjadi pemisah terbaik maka dilakukan perhitungan *Gini Impurity* terhadap 2 variabel tersebut.

   Maka *Gini Impurity* dari variabel sakit dada adalah:

$$GI_{True} = 1 - (Probabilitas\ \text{"Ya"})^2 - (Probabilitas\ \text{"Tidak"})^2$$

$$GI_{True} = 1 - \left(\frac{105}{105+59}\right)^2 - \left(\frac{39}{105+39}\right)^2$$

$$GI_{True} = 0{,}395$$

$$GI_{True} = 1 - (Probabilitas\ \text{"Ya"})^2 - (Probabilitas\ \text{"Tidak"})^2$$

$$GI_{True} = 1 - \left(\frac{34}{34+125}\right)^2 - \left(\frac{125}{34+125}\right)^2$$

$$GI_{True} = 0{,}336$$

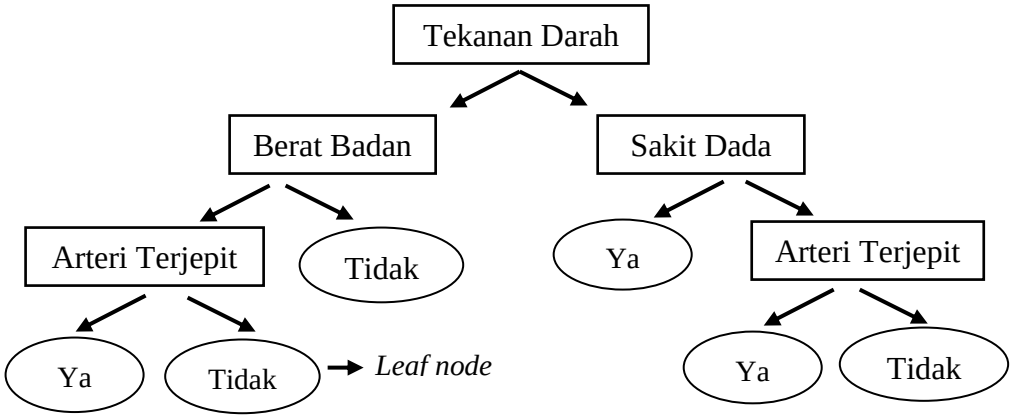
$$GI_{Total} = 1 - (Probabilitas\ \text{"Ya"})^2 - (Probabilitas\ \text{"Tidak"})^2$$

$$_{ul} = 1 - \left(\frac{144}{144+159}\right)^2 - \left(\frac{159}{144+159}\right)^2$$

$$_{ul} = 0{,}364$$

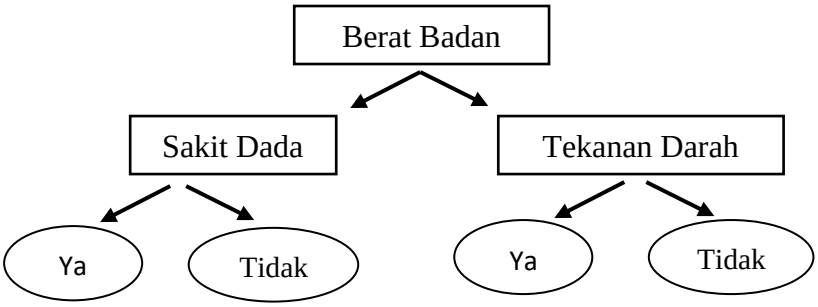
Lakukan perhitungan *Gini Impurity* terhadap variabel tekanan darah dengan cara yang sama, maka diperoleh $GI_{Total} = 0{,}360$. Berdasarkan hasil perhitungan nilai *Gini Impurity* total dari variabel tekanan darah lebih kecil dibanding variabel sakit dada maka variabel yang akan dijadikan sebagai *root node* adalah variabel dengan nilai *Gini Impurity* total terkecil yaitu tekanan darah.

3) Untuk *root node* berikutnya kembali dilakukan pemilihan 2 variabel dari 3 variabel yang tersisa secara acak misalkan dipilih variabel arteri terjepit dan berat badan. Lakukan perhitungan *Gini Impurity* terhadap masing-masig variabel dan pilih variabel dengan nilai *Gini Impurity* total terkecil sebagai *root node*. Misalkan nilai *Gini Impurity* total dari variabel berat badan lebih kecil maka variabel tersebut menjadi *rood node* selanjutnya. Ulangi langkah tersebut hingga semua *cutoff* dari variabel terbentuk node akhir dari pohon keputusan dimanakan *leaf node*. Node akhir tersebut merupakan keputusan akhir yang nantinya akan digunakan sebagai kesimpulan akhir dari proses RF.
4) Untuk pembentukan sisi kanan dari pohon keputusan ulangi langkah 2-3. Gambar 7 adalah contoh gambaran akhir pohon keputusan yang terbentuk

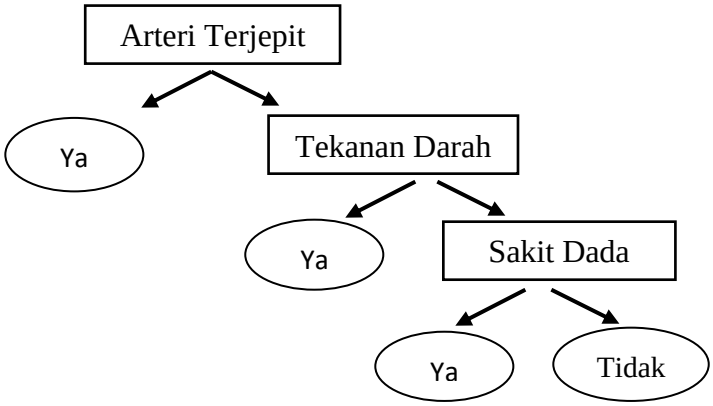


Gambar 7. Contoh Model 1 Hasil Akhir Pohon Keputusan

5) Ulangi langkah 1-4 hingga didapatkan beberapa model pohon keputusan. Proses *bootstrapped* idealnya dilakukan sebanyak 100 kali. Gambar 8 dan bar 9 adalah beberapa gambaran model lain contoh pohon keputusan dapat terbentuk.



Gambar 8. Contoh Model 2 Pohon Keputusan



Gambar 9. Contoh Lain Model 3 Pohon Keputusan

6) Untuk mengambil keputusan dari model RF yang telah dibentuk akan diujikan data baru terhadap model tersebut. Misalkan terdapat sebuah data contoh data uji pada Tabel 4.

Tabel 4. Contoh Data Uji Model RF

Sakit Dada	Tekanan Darah	Arteri Terjepit	Berat Badan	Penyakit Jantung
Ya	Tidak	Tidak	168	…?

Dari contoh di atas ingin diketahui apakan pasien dengan data tersebut dapat dikategorikan sebagai pasien berpenyakit jantung atau tidak maka uji data tersebut ke dalam model-model pohon keputusan yang telah dibuat sebelumnya. Asumsikan hasil yang diperoleh adalah sebagai berikut:

- Model 1 = Ya
- Model 2 = Ya
- Model 3 = Tidak

Berdasarkan hasil yang diperoleh prediksi untuk kelas Ya= 2 dan Tidak= 1 maka dapat disimpulkan bahwa pasien tersebut menderita penyakit jantung.

Selanjutnya dikombinasikan dengan algoritma *Recursive Feature Elimination* (RFE) adalah sebuah metode yang digunakan di analiss data microarray untuk proses seleksi fitur yang dapat digunakan untuk menghapus data yang tidak berkorelasi dengan kelas, yang mana hal tersebut dapat meningkatkan efisiensi dan akurasi

Algoritma *Recursive Feature Elimination*

1. Melatih model dari data training menggunakan seluruh prediktor
2. Mengkalkulasi performansi model
3. Mengkalkulasi variabel penting atau ranking
4. **For** Untuk setiap subset size $S_i, i = 1,2, \dots, S$ **do**

   Simpan S_i sebagai variabel penting

   Melatih model dari data training menggunakan S_i sebagai prediktor

   Kalkulasikan performansi model
5. **End**
6. Kalkulasikan performansi S_i
7. Tentukan jumlah predictor yang sesuai
8. Gunakan model yang sesuai untuk mengoptimalkan S_i

C. Support Vector Machine-Recursive Feature Elimination

Tujuan utama dari *Support Vector Machine-Recursive Feature Elimination* (SVM-RFE) adalah untuk menghitung bobot dari seluruh fitur dan menggolongkan fitur berdasarkan bobot vektor sebagai dasar klasifikasi SVM. SVM-RFE adalah proses iterasi yang menggunakan aturan *backward elimination*, yang mana proses dimulai dengan seluruh variabel fitur dan menghapus satu variabel fitur pada waktu yang sama. Setiap step koefisien dari bobot vektor **w** dari SVM linier digunakan untuk menghitung skor ranking fitur. Fitur ke-i dengan ranking skor terkecil $c_i = (w_i)^2$ dieliminasi ...na w_i merepresentasikan komponen yang saling berkoresponding pada ... vektor **w** dan $c_i = (w_i)^2$ sebagai kriteria ranking yang sesuai untuk

menghapus fitur yang tidak memberikan pengaruh yang besar terhadap objek apabila dihilangkan [22].

Algoritma eliminasi rekursif pada SVM-RFE pertama kali diusukan oleh [30], yaitu sebagai berikut:

1. Diberikan $\{x_i, y_i\}, i = 1,2, \dots, n$ adalah himpunan pasangan data sebanyak n, dengan $\boldsymbol{x}_i \in \mathbb{R}^m$ dan $y_i \in \{+1, -1\}$ adalah target.
2. Pelatihan klasifikasi dengan menggunakan SVM untuk mendapatkan vektor $\boldsymbol{w}$ yang merepresentasikan parameter bobot
$$\boldsymbol{w} = \sum_{i=1}^{n} a_i, y_i, \boldsymbol{x}_i \quad (13)$$
3. Hitung vektor $\boldsymbol{w}$ yang merepresentasikan parameter bobot
4. Hitung skor ranking untuk semua fitur $c_j = (w_j)^2$ dengan $w_j, j = 1,2,.., m$ adalah elemen pada vektor $\boldsymbol{w}$
5. Pencarian fitur dengan skor ranking terendah yaitu $f = \arg\min(c_j)$
6. Mengeliminasi fitur dengan $c = f$ (fitur yang dieliminasi bisa lebih dari satu fitur).

2.1.9. Proses Evaluasi Hasil

Evaluasi model bertujuan untuk mengetahui keakuratan model fungsi klasifikator dalam memprediksi data baru yang bukan termasuk dalam data pelatihan. *K – Fold Cross Validation* digunakan untuk menghitung akurasi model fungsi klasifikator terhadap data baru. Dalam penelitian ini, yang digunakan 10 – *Fold Cross Validation*. Untuk menyajikan hasil *K – Cross Validation,* digunakan *confusion matrix*. Tabel 5 merupakan skema dari *confusion matrix.*

Tabel 5. Skema *Confusion Matrix*

		Target Prediksi	
		-1	1
Target Sebenarnya	-1	*True Negative* (TN)	*False Positive* (FP)
	1	*FalseNegative* (FN)	*True Positive* (TP)

- *True Posstive* (TP), yaitu jumlah data dari kelas 1 yang benar dan diklasifikan sebagai kelas 1.
- *True Negative* (TN), yaitu jumlah data dari kelas -1 yang benar diklasifikasikan sebagai kelas -1.
- *False Positive* (FP), yaitu jumlah data dari kelas -1 yang salah diklasifikasikan sebagai kelas 1.
- *False Negative* (FN) yaitu jumlah data dari kelas 1 yang salah diklasifikasikan sebagai kelas -1

Berdasarkan skema *confusion matrix* pada Tabel 4 satuan kinerja model yaitu sebagai berikut:

1. *Succes rate* atau tingkat akurasi kesuksesan adalah proporsi jumlah prediksi yang benar. Dapat dihitung dengan:

$$Succes\ Rate = \frac{TP+TN}{TP+TN+FP+FN} \quad (14)$$

2. *True Positif Rate* (*TPR)* atau *Sensitivity* adalah membandingkan proporsi TP terhadap tupel yang positif. Dapat dihitung dengan:

$$TPR = \frac{TP}{TP+FN}\ x\ 100\% \quad (15)$$

3. *False Positif Rate (FPR)* atau *Specifivity* adalah membandingkan proporsi FP terhadap tupel yang negatif. Dapat dihitung dengan:

$$FPR = \frac{FP}{FP+TN}\ x\ 100\% \quad (16)$$

ROC Curve (Receiver Operating Charateristic) adalah alat visual yang berguna untuk membandingkan dua atau lebih model klasifikasi. *ROC Curve* adalah grafik dua dimensi dengan FP sebagai garis horizontal dan TP sebagai garis vertikal. Nilai *Area Under Curve* (AUC) merupakan salah satu ukuran yang berhubungan paling populer dengan kurva ROC. AUC adalah ukuran gabungan sensitivitas dan spesifisitas yang merupakan ukuran keseluruhan kinerja tes diagnostik yang diinterpretasikan sebagai rata-rata nilai sensitivitas untuk semua kemungkinan nilai spesifisitas [31]. Nilai AUC terentang antara 0 dan 1 karena sumbu x dan y ...yai nilai mulai dari 0 sampai 1. Jika nilai AUC mendekati 1, kinerja ...an tes diagnostik semakin baik dan tes dengan nilai AUC = 1 berarti ...ngat akurat [32]. Rumus AUC sebagai berikut [33]:

$$AUC = \frac{\sum_{i=1}^{n^+} \sum_{j=1}^{n^-} 1_{f(x_i^+) > f(x_j^-)}}{n^+ n^-} \quad , \qquad (17)$$

dimana :

f() = nilai suatu fungsi

x^+dan x^- = sampel positif dan negatif

n^+dan n^- = jumlah sampel positif dan negatif.

Tingkat akurasi nilai AUC dalam klasifikasi dibagi menjadi lima kelompok dinyatakan dalam Tabel 6 berikut :

Tabel 6. Tingkat Akurasi AUC

Interval Nilai AUC	**Tingkat Akurasi**
0.90 – 1.00	Sangat bagus
0.80 – 0.90	Bagus
0.70 – 0.80	Cukup bagus
0.70 – 0.80	Cukup bagus
0.50 – 0.60	Sangat buruk

2.2. KERANGKA KONSEPTUAL

Meningkatnya jumlah penderita DM tipe 2 setiap tahunnya tetapi tidak diiringi dengan penanganan medis yang tepat untuk menyembuhkan penyakit tersebut secara permanen. Oleh sebab itu, penelitian mengenai penanganan yang tepat untuk penderita DM tipe 2 terus dilakukan salah satunya adalah dengan mengidentifikasi biomarker dari penyakit DM tipe 2 dengan memanfaatkan ekspresi gen yang dihasilkan dari teknologi microarray

Penggunaan ekspresi gen dari teknologi DNA microarray digunakan untuk menentukan tingkat ekspresi ribuan gen yang dilakukan dalam sekali percobaan, dan secara simultan memantau proses biologi yang sedang berlangsung. Teknologi ini membantu para peneliti dalam mempelajari cara penanganan yang tepat terhadap penyakit, termasuk DM tipe 2. Akan tetapi data microarray memiliki dimensi yang sangat besar sehingga dibutuhkan metode yang tepat untuk mereduksi sekaligus mengklasifikasi data tersebut.

Solusi yang ditawarkan adalah menemukan metode seleksi fitur yang memiliki tingkat akurasi tinggi untuk menghasilkan prediksi kandidat biomarker.

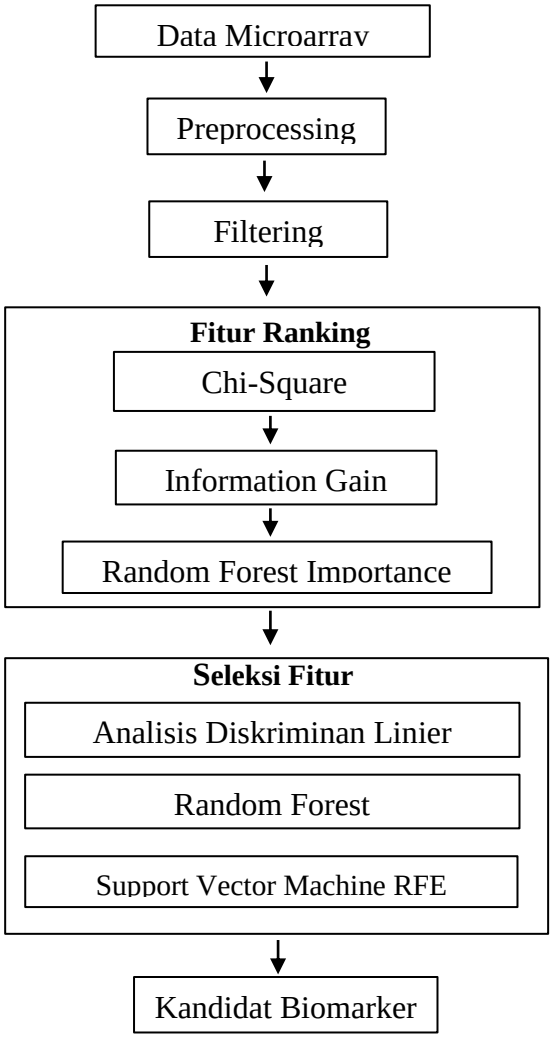
Dalam penelitian ini akan digunakan beberapa metode seleksi fitur untuk memilih gen informatif dari data micro array untuk penyakit DM tipe 2

Dengan adanya penelitian ini diharapkan dapat ditemukan metode seleksi fitur serta yang tepat untuk menganalisis data microarray sehingga dapat dihasilkan prediksi biomarker dengan akurasi yang tinggi.

BAB III
METODE PENELITIAN

3.1. TAHAPAN PENELITIAN

Tahapan penelitian ini di awali dengan tahap preprocessing data. Selanjutnya dilakukan penyaringan (*Filtering*) kemudian hasil dari proses tersebut diranking menggunakan 3 metode yaitu chi-square, *information gain, random forest importance.* Setelah diranking, akan dilakukan seleksi fitur terhadap gen yang di peroleh dari proses sebelumnya menggunakan 3 metode seleksi fitur yaitu analisis diskriminan linier, *random forest* dan SVM-RFE. Gambar 10 adalah gambaran untuk tahapan dari penelitian ini.



Gambar 10. Tahapan Penelitian

3.2. WAKTU DAN TEMPAT

Penelitian ini dilaksanakan dari bulan September 2018 sampai dengan bulan Desember 2018. Lokasi penelitian dilakukan di Laboratorium Rekayasa Perangkat Lunak Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Hasanuddin.

3.3. SUMBER DATA

Data microarray yang digunakan adalah GSE18732 diperoleh dari https://www.ncbi.nlm.nih.gov/geo/query/acc.cgi?acc=GSE18732. Data yang digunakan berisikan 118 sample yang merupakan data ekspresi gen dari pasien penderita penyakit DM tipe 2, pasien normal, dan pasien *glucose intolerant.*

3.4. INSTRUMEN PENELITIAN

Instrumen yang digunakan dalam penelitian adalah komputer dengan Processor 64 Bit Intel i7, dengan RAM sebesar 16 Gigabyte menggunakan sistem operasi Windows 10. Pengolahan data dilakukan menggunakan software R 3.5.0

3.5. VARIABEL PENELITIAN

Variabel – variabel yang digunakan dalam penelitian ini adalah sebagai berikut:

a. Variabel bebas, yaitu data berupa gen ekspresi
b. Variabel terikat, yaitu data kelas antara berpenyakit dan tidak berpenyakit

BAB IV
HASIL DAN PEMBAHASAN

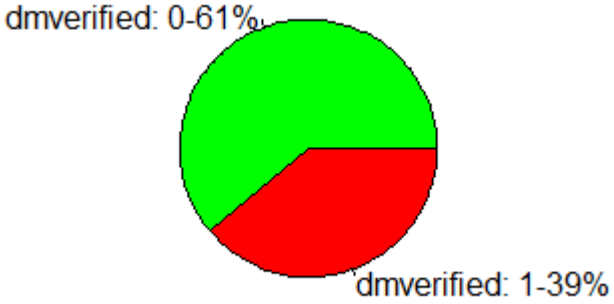
4.1. DESKRIPSI DATA

Tabel 7 adalah gambaran dari data GSE18732 yang digunakan pada penelitian ini. Data tersebut merupakan data ekspresi gen yang dimbil dari ekspresi mRNA pada otot tulang dari pasien normal, pasien DM tipe 2 dan pasien *glucose intolerant* dengan jumlah total pasien sebanyak 118 orang dan berasal dari Inggris.

Tabel 7. Pheno Data GSE18732

	title	⋯	characteristics_ch1.5	⋯	dmverified:ch1	whr:ch1
GSM465274	muscle_glucoseIntolerant_15493	⋯	dmverified: 0	⋯	0	0.96
GSM465275	muscle_normal_15494	⋯	dmverified: 0	⋯	0	0.85
GSM465276	muscle_glucoseIntolerant_15496	⋯	dmverified: 0	⋯	0	1.15
GSM465277	muscle_normal_15497	⋯	dmverified: 0	⋯	0	1.05
GSM465278	muscle_diabetic_15498	⋯	dmverified: 1	⋯	1	1.05
GSM465279	muscle_normal_15501	⋯	dmverified: 0	⋯	0	0.97
GSM465280	muscle_diabetic_15504	⋯	dmverified: 1	⋯	1	1.05
GSM465281	muscle_diabetic_15510	⋯	dmverified: 1	⋯	1	0.99
GSM465282	muscle_diabetic_15516	⋯	dmverified: 1	⋯	1	1.09
GSM465283	muscle_glucoseIntolerant_15522	⋯	dmverified: 0	⋯	0	1
GSM465284	muscle_normal_15534	⋯	dmverified: 0	⋯	0	1.07
GSM465285	muscle_glucoseIntolerant_15537	⋯	dmverified: 0	⋯	0	0.98
GSM465286	muscle_diabetic_15549	⋯	dmverified: 1	⋯	1	0.96
⋮	⋮	⋯	⋮	⋯	⋮	⋮
GSM465340	muscle_glucoseIntolerant_16616	⋯	dmverified: 1	⋯	1	0.87
⋮	⋮	⋯	⋮	⋯	⋮	⋮
GSM465391	muscle_normal_17197	⋯	dmverified: 0	⋯	0	0.86

Gambar 11 adalah persentasi pasien secara keseluruhan dimana jumlah pasien yang menderita DM tipe 2 sebanyak 39% dan pasien sehat sebanyak 61%.



Gambar 11. Persentasi Pasien

4.2. PENGOLAHAN DATA

Pengolahan data akan digunakan menggunakan software R, untuk memperoleh data yang akan digunakan digunakan package GEOquery dan untuk membaca data *Affymetrix* atau data ekspresi gen dibutuhkan *package affy*. Seluruh package yang digunakan pada penelitian ini terdapat pada Lampiran 1.

4.2.1. Preprocessing

Tahap ini memerlukan fungsi *threestep()* dari *package affyPLM*. Fungsi *threestep()* selain melakukan konversi data *affybatch,* dalam fungsi tersebut juga dilakukan proses *background correction, normalization* dan *summarization*. Metode *background correction* dalam penelitian ini menggunakan *Robust Multi-array Average* (RMA). Metode RMA bertujuan untuk mengetahui distribusi intesitas probe [31]. Metode normalisasi RMA adalah algoritma yang digunakan untuk membuat matriks ekspresi dari data *Affymetrix*. Dalam data ekspresi gen terdapat pasang probe yang terdiri atas pasangan (PM) dan (MM). Probe PM adalah probe yang didesain untuk mengikat pasangan basa gen terget secara tepat sedangkan probe MM adalah probe yang didesain untuk mengikat pasangan basa yang bukan merupakan pasangan basa aslinya atau dengan kata lain tidak mengikat pasangan basanya secara spesifik sehingga sering terlihat sebagai *background-corrected*. RMA membentuk matriks ekspresi gen dimulai dengan menghitung intensitas *background-corrected* PM untuk setiap PM dalam sel. Setelah itu tahap melakukan transformasi log2 untuk setiap intensitas *background-corrected* PM yang diperoleh. Tahapan setelah *background correction* adalah melakukan *normalization* dengan metode quantile untuk menyetarakan data. Pada proses normalisasi nilai tertinggi dari background-corrected PM dan transformasi log2 dari setiap gen ditentukan. Setiap nilai kemudian dirata-ratakan, dan nilai individu diganti dengan nilai rata tersebut. Proses tersebut diulang hingga diperoleh *background-corrected* dua tertinggi dan tiga tertinggi nilai transformasi log2 dari intensitas PM pada setiap gen. Tahap ilakkukan *background correction* dan *normalizatition* dilanjutkan ke *mmarization* dengan metode *median polish*. Seluruh proses dari tahap

preprocessing terdapat pada Lampiran 2. Tabel 8 adalah bentuk data sebelum dilakukan proses preprocessing.

Tabel 8. Data Sebelum Preprocessing

GSM465274.CEL.gz	169	12680	248	12555
GSM465275.CEL.gz	122	12237	220	12398
GSM465276.CEL.gz	176	15112	247	15514
GSM465277.CEL.gz	223	14424	264	16022
GSM465278.CEL.gz	164	9838	218	9845
GSM465279.CEL.gz	113	13287	237	13383
GSM465280.CEL.gz	114	8179	204	8397
GSM465281.CEL.gz	536	14343	702	15548
GSM465282.CEL.gz	109	10330	161	10345
GSM465283.CEL.gz	133	9521	189	9413
⋮	⋮	⋮	⋮	⋮

Setelah dilakukan proses preprocessing maka terjadi perubahan pada data awal. Data hasil proses preprocessing ditampilkan pada Tabel 9.

Tabel 9**.** Data Setelah Preprocessing

GSM465274.CEL.gz	7.96557	6.170039	4.828359	7.926923
GSM465275.CEL.gz	8.300493	6.313449	4.844165	8.273382
GSM465276.CEL.gz	8.141435	6.291263	4.873871	8.232847
GSM465277.CEL.gz	7.708152	5.867595	4.929714	8.337393
GSM465278.CEL.gz	8.123801	6.173816	4.733844	8.106295
GSM465279.CEL.gz	7.847043	6.136359	4.872117	8.20431
GSM465280.CEL.gz	7.864656	6.211747	4.900254	8.133455
GSM465281.CEL.gz	8.299243	6.419662	5.503506	7.948785
GSM465282.CEL.gz	8.562667	6.38846	4.77162	7.977001
GSM465283.CEL.gz	7.925344	6.28611	4.804959	8.106792
⋮	⋮	⋮	⋮	⋮

4.2.2. Penyaringan (*Filtering*)

Penyaringan data microarray adalah proses pemilihan subset dari probe yang tersedia untuk pengecualian atau penyertaan dalam analisis. Program R menggunakan tambahan *package genefilter* untuk melakukan penyaringan. Fungsi pertama yang digunakan pada tahap *filtering* pada penelitian ini ialah *nsFilter*, fungsi ini menyediakan suatu opsi serba ada untuk berbagai pilihan penyaringan *expression set*. Fitur *filtering* dapat menunjukkan beberapa varian kecil mpangan data secara konsisten di seluruh sampel, hal ini dapat berguna lisis selanjutnya.