

**PERBANDINGAN EMPIRIS FUNGSI *KERNEL* PADA METODE  
KLASIFIKASI *SUPPORT VECTOR MACHINE* DAN PENERAPANNYA  
PADA DATA PENDERITA PENYAKIT JANTUNG**

**TESIS**

Oleh

**Dira Dini Dian Kemala  
NPM. 2127031003**



**PROGRAM STUDI MAGISTER MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

## ABSTRAK

### PERBANDINGAN EMPIRIS FUNGSI *KERNEL* PADA METODE KLASIFIKASI *SUPPORT VECTOR MACHINE* DAN PENERAPANNYA PADA DATA PENDERITA PENYAKIT JANTUNG

Oleh

Dira Dini Dian Kemala

*Support Vector Machine* (SVM) adalah salah satu metode *machine learning* yang digunakan untuk mengklasifikasikan dengan membagi data menjadi dua kelas yang berbeda. Prinsip metode SVM adalah mencari fungsi pemisah / *hyperplane* yang terbaik. Apabila data tidak dapat dipisahkan secara linear maka data tersebut merupakan data nonlinear. Salah satu metode untuk mengatasi hal tersebut adalah fungsi *kernel*. Tujuan penelitian ini adalah mengetahui fungsi *kernel* terbaik yang menghasilkan akurasi tertinggi pada SVM melalui studi simulasi dan menerapkan metode SVM untuk klasifikasi penderita penyakit jantung. Hasil yang diperoleh, melalui studi simulasi bahwa fungsi *kernel* terbaik adalah fungsi *kernel Radial Basis Function* (RBF), proporsi *split* dataset yang terbaik adalah 80 *training* dan 20 *testing*, dan parameter terbaik adalah  $cost = 1$ ,  $degree = 1$ , dan  $gamma = 0,5$ . Selain itu, penerapan metode SVM pada data penyakit jantung diperoleh hasil bahwa fungsi *kernel* RBF dengan *split* dataset 80 diperoleh nilai akurasi = 1, artinya dalam mengklasifikasikan seseorang apakah termasuk penderita atau tidak penderita dapat diklasifikasikan secara tepat. Dengan menggunakan fungsi *kernel* dan proporsi *split* dataset tersebut diperoleh parameter  $w$  dan  $b$ , yakni sebagai berikut:  $w_{Age} = -9,43$ ,  $w_{Gender(1)} = 4,22$ ,  $w_{Gender(2)} = -4,22$ ,  $w_{Chest Pain Type(1)} = 6,73$ ,  $w_{Chest Pain Type(2)} = -0,92$ ,  $w_{Chest Pain Type(3)} = -5,44$ ,  $w_{Resting Blood Pressure} = -12,49$ ,  $w_{Cholesterol} = 11,264$ ,  $w_{Fasting Blood Sugar1} = -4,639$ ,  $w_{Resting ECG1} = -3,769$ ,  $w_{Resting ECG2} = -1,456$ ,  $w_{MaxHR} = 19,413$ ,  $w_{Exercise Angina1} = -9,583$ ,  $w_{Oldpeak} = -25,042$ ,  $w_{STslope1} = -13,259$ ,  $w_{STslope2} = -1,762$ ,  $b = 0,0734$ .

**Kata Kunci** : *Support Vector Machine*; *Kernel Radial Basis Function*; Klasifikasi; Penyakit Jantung

## ABSTRACT

Empirical Comparison of Kernel Function in Support Vector Machine Classification Method and its Application to Data of Heart Disease Patients

By

**Dira Dini Dian Kemala**

Support Vector Machine (SVM) is a machine learning method that is used to classify data by dividing data into two different classes. The principle of the SVM method is to find the best separator / hyperplane function. If the data cannot be separated linearly, then the data is nonlinear. One method to overcome this is by kernel function approach. This research aims to find the best kernel function that produces the highest accuracy in SVM through a simulation study and to apply the SVM method to classify people with heart disease. The results of simulation study showed that the best kernel function is the Radial Basis Function (RBF) kernel function, the best proportion of split datasets is 80 training and 20 testings, and the best parameters are cost = 1, degree = 1, and gamma = 0, 5. In addition, the application of the SVM method to heart disease data yields the result that the RBF kernel function with a split dataset of 80 obtains an accuracy value of = 1, meaning that in classifying a person, whether a person is a sufferer or not, a patient can be classified correctly. By using the kernel function and the split proportion of the dataset, the parameters  $w$  and  $b$  are obtained, namely as follows:

$w_{Age} = -9,43, w_{Gender(1)} = 4,22, w_{Gender(2)} = -4,22, w_{Chest\ Pain\ Type(1)} = 6,73,$   
 $w_{Chest\ Pain\ Type(2)} = -0,92, w_{Chest\ Pain\ Type(3)} = -5,44, w_{Resting\ Blood\ Pressure} = -12,$   
 $49, w_{Cholesterol} = 11,26, w_{Fasting\ Blood\ Sugar1} = -4,64, w_{RestingECG1} = -3,76,$   
 $w_{RestingECG2} = -1,456, w_{MaxHR} = 19,413, w_{ExerciseAngina1} = -9.583, w_{Oldpeak} = -25,0$   
 $42, w_{STslope1} = -13,259, w_{STslope2} = -1,762, b = 0,0734.$

**Keywords:** Support Vector Machine; Kernel Radial Basic Functions; Classification; Heart disease

**PERBANDINGAN EMPIRIS FUNGSI *KERNEL* PADA METODE  
KLASIFIKASI *SUPPORT VECTOR MACHINE* DAN PENERAPANNYA  
PADA DATA PENDERITA PENYAKIT JANTUNG**

Oleh

**Dira Dini Dian Kemala**

Tesis

**Sebagai Salah Satu Syarat untuk Mencapai Gelar  
MAGISTER MATEMATIKA**

**Pada  
Jurusan Matematika  
Fakultas Matematika dan Ilmu Pengetahuan Alam  
Universitas Lampung**



**PROGRAM STUDI MAGISTER MATEMATIKA  
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM  
UNIVERSITAS LAMPUNG  
BANDAR LAMPUNG  
2023**

Judul Tesis : **PERBANDINGAN EMPIRIS FUNGSI  
KERNEL PADA METODE KLASIFIKASI  
SUPPORT VECTOR MACHINE DAN  
PENERAPANNYA PADA DATA  
PENDERITA PENYAKIT JANTUNG**

Nama Mahasiswa : **Dira Dini Dian Kemala**

Nomor Pokok Mahasiswa : **2127031003**

Program Studi : **Magister Matematika**

Jurusan : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



**Dr. Khoirin Nisa, S.Si., M.Si.**  
NIP. 19740726 200003 2 001

**Prof. Drs. Mustofa Usman, M.A., Ph.D.**  
NIP. 19570101 198403 1 020

**2. Ketua Program Studi Magister Matematika**

**Prof. Dr. Asmiati, S.Si., M.Si.**  
NIP. 19760411 200012 2 001

## MENGESAHKAN

### 1. Tim Penguji

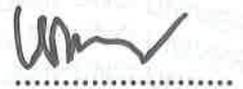
Ketua : **Dr. Khoirin Nisa, S.Si., M.Si.**



Sekretaris : **Prof. Drs. Mustofa Usman, M.A., Ph.D.**



Penguji Anggota : **1. Ir. Warsono, M.S., Ph.D.**



**2. Dr. Muslim Ansori, S.Si., M.Si.**



### 2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



**Dr. Eng. Heri Satria, S.Si., M.Si.**  
NIP 19711001 200501 1 002



### 3. Direktur Program Pascasarjana



**Prof. Dr. H. Murhadi, M.Si.**  
NIP 19640326 198902 1 001



Tanggal Lulus Ujian Tesis : **27 Juni 2023**

## PERNYATAAN

Saya yang bertanda tangan di bawah ini:

Nama : **Dira Dini Dian Kemala**  
Nomor Pokok Mahasiswa : **2127031003**  
Program Studi : **Magister Matematika**  
Jurusan : **Matematika**

Dengan ini menyatakan bahwa tesis saya yang berjudul “**Perbandingan Empiris Fungsi *Kernel* pada Metode Klasifikasi *Support Vector Machine* dan Penerapannya pada Data Penderita Penyakit Jantung**” ini adalah hasil pekerjaan saya sendiri, bukan hasil orang lain, dan semua hasil tulisan yang tertuang dalam tesis ini telah mengikuti kaidah penulisan karya ilmiah Universitas Lampung. Apabila kemudia hari terbukti bahwa tesis ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 27 Juni 2023

Penulis,



Dira Dini Dian Kemala  
NPM. 2127031003

## **RIWAYAT HIDUP**

Penulis dilahirkan pada tanggal 20 Agustus 1999 di Pringsewu, anak kedua dari dua bersaudara pasangan Bapak Zulbahri (Alm.) dan Ibu Armiyati, serta adik dari Nindo Mayang Sumba.

Penulis memulai pendidikan di TK IT Rabbani Bengkulu pada tahun 2004-2005, pendidikan tingkat dasar di SD Negeri 19 Bengkulu pada tahun 2005-2011.

Pada tahun 2011, penulis melanjutkan ke pendidikan tingkat menengah pertama di SMP Negeri 1 Bengkulu pada tahun 2011 sampai dengan tahun 2012, dan kemudian melanjutkan pendidikan tingkat menengah pertama di SMP Negeri 1 Bandar Lampung pada tahun 2012-2014. Kemudian penulis melanjutkan pendidikan tingkat menengah atas di SMA Al-Azhar 3 Bandar Lampung dan diselesaikan pada tahun 2017.

Pada tahun 2017 melalui jalur SBMPTN, penulis diterima dan terdaftar sebagai mahasiswa S1 Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung. Pada tahun 2021 penulis berkesempatan untuk melanjutkan pendidikan di program studi Magister Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung, selama menjadi mahasiswa Magister penulis cukup aktif mengikuti seminar internasional, webinar, dan kuliah umum.

## **KATA MUTIARA**

"Kemudian apabila kamu telah membulatkan tekad, maka bertawakallah kepada Allah. Sesungguhnya Allah menyukai orang-orang yang bertawakal kepada-Nya."

**(Q.S Ali Imran: 159)**

"Dan barang -siapa yang bertakwa kepada Allah, niscaya Allah menjadikan baginya kemudahan dalam urusannya."

**(Q.S At-Talaq: 4)**

"Allah tidak membebani seseorang melainkan sesuai dengan kesanggupannya."

**(QS. Al-Baqarah: 286)**

## **PERSEMBAHAN**

Dengan mengucapkan puji dan syukur kehadirat Allah SWT yang telah memberikan petunjuk dan kemudahan untuk menyelesaikan studiku, kupersembahkan karya kecilku ini untuk:

Papa (Alm.) dan Mama tercinta yang selalu mendidik, mendoakan, membantu, mendukung, dan hal lainnya yang tidak dapat diungkapkan dengan kata-kata.

Uniku tersayang.

Dosen pembimbing dan penguji yang sangat berjasa dan tidak lelah memberikan arahan serta masukan sehingga peulis dapat menyelesaikan tesisku.

Sahabat dan teman-temanku, Terimakasih atas kebersamaan, do'a dan semangat yang selalu kalian berikan kepadaku.

Universitas Lampung.

## SANWACANA

Puji syukur kepada Allah SWT. karena limpahan rahmat, hidayah, dan atas seizin-Nya tesis dengan judul “Perbandingan Empiris Fungsi *Kernel* pada Metode Klasifikasi *Support Vector Machine* dan Penerapannya pada Data Penderita Penyakit Jantung” dapat penulis selesaikan dengan baik. Sholawat beriring salam selalu penulis haturkan kepada Nabi Muhammad SAW., yaitu manusia biasa yang dengan kebiasaannya menjadi manusia luar biasa dan menjadi suri tauladan bagi setiap muslim.

Terselesaikannya penelitian ini tidaklah terlepas dari bantuan, kerjasama, dan dukungan berbagai pihak. Pada kesempatan kali ini penulis mengucapkan sangat berterimakasih kepada :

1. Dr. Khoirin Nisa, S.Si., M.Si., selaku Dosen Pembimbing 1 dan Dosen Pembimbing Akademik yang telah banyak memberikan bimbingan, kritik, saran, dan *support* dalam penyelesaian tesis ini sehingga penulis tidak hanya menyelesaikan tesis ini, tetapi penulis juga mendapatkan pengetahuan baru yang insyaAllah akan sangat berguna dikemudian hari.
2. Prof. Drs. Mustofa Usman, M.A., Ph.D., selaku Dosen Pembimbing 2 yang memberikan pengarahan dan saran dalam penyelesaian tesis.
3. Ir. Warsono, M.S., Ph.D., selaku Dosen Pembahas 1 yang telah memberikan kritik dan saran kepada penulis dalam penyelesaian tesis.
4. Dr. Muslim Ansori, S.Si., M.Si., selaku Dosen Pembahas 1 yang telah memberikan kritik dan saran kepada penulis dalam penyelesaian tesis.
5. Dr. Aang Nuryaman, S.Si., M.Si., selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam.

6. Prof. Dr. Asmiati, S.Si., M.Si. selaku Ketua Program Studi Magister Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam.
7. Dr. Eng. Suripto Dwi Yuwono, M.T. selaku Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
8. Dosen, staf dan karyawan Jurusan Matematika FMIPA Universitas Lampung yang telah memberikan ilmu dan bantuan kepada penulis.
9. Papa (Alm.), Mama, dan Uni tercinta, yang selalu mendo'akan penulis, memberikan dukungan yang tak terhingga serta menjadi penyemangat tersendiri bagi penulis.
10. Sahabat penulis, yaitu Nabila yang selalu membantu dan memotivasi penulis dalam belajar, dan menghibur penulis saat suka maupun duka.
11. Teman-teman Magister Matematika Angkatan 2021.

Penulis juga menyadari bahwa dalam penulisan tesis ini masih banyak terdapat kekurangan. Oleh karena itu, Penulis mengharapkan saran dan kritik yang membangun guna penelitian selanjutnya agar lebih baik.

Bandar Lampung, 27 Juni 2023  
Penulis,

Dira Dini Dian Kemala

## DAFTAR ISI

Halaman

<b>DAFTAR TABEL</b> .....	xiv
<b>DAFTAR GAMBAR</b> .....	xvi
<b>I. PENDAHULUAN</b>	
1.1 Latar Belakang Penelitian.....	1
1.2 Tujuan Penelitian.....	2
1.3 Manfaat Penelitian.....	3
<b>II. TINJAUAN PUSTAKA</b>	
2.1 <i>Machine Learning</i> .....	4
2.2 <i>Supervised learning</i> .....	5
2.3 <i>Support Vector Machine</i> .....	6
2.3.1 Definisi <i>Support Vector Machine</i> .....	6
2.3.2 Karakteristik <i>Support Vector Machine</i> .....	6
2.3.3 Kelebihan <i>Support Vector Machine</i> .....	6
2.3.4 <i>Hard Margin</i> .....	8
2.3.5 <i>Soft Margin</i> .....	13
2.3.6 <i>Kernel Support Vector Machine</i> .....	16
2.4 Pemilihan Parameter pada <i>Support Vector Machine</i> .....	19
2.5 Evaluasi Model.....	20
<b>III. METODOLOGI PENELITIAN</b>	
3.1 Waktu dan Tempat Penelitian.....	22
3.2 Data Penelitian.....	22
3.2.1 Data Simulasi.....	22
3.2.2 Data Penyakit Jantung.....	22
3.3 Metode Penelitian.....	23
3.3.1 Metode Penelitian Data Simulasi.....	23
3.3.2 Metode Penelitian Analisis Data Penyakit Jantung.....	25
<b>IV. HASIL DAN PEMBAHASAN</b>	
4.1 Nilai Akurasi Data Simulasi.....	28
4.1.1 Nilai Akurasi Simulasi Data dengan Fungsi <i>Kernel</i> Linear.....	28
4.1.2 Nilai Akurasi Simulasi Data dengan Fungsi <i>Kernel</i> Polinomial.....	30
4.1.3 Nilai Akurasi Simulasi Data dengan Fungsi <i>Kernel</i> RBF.....	33

4.2 Penerapan Fungsi <i>Kernel</i> pada Data Penyakit Jantung.....	35
4.3 Contoh Perhitungan dalam Pengaplikasian pada Fungsi <i>Kernel</i> Metode <i>Support Vector Machine</i> .....	38
4.3.1 Pengaplikasian pada Fungsi <i>Kernel</i> Linear.....	40
4.3.2 Pengaplikasian pada Fungsi <i>Kernel</i> Polinomial.....	43
4.3.3 Pengaplikasian pada Fungsi <i>Kernel</i> RBF.....	45
<b>V. KESIMPULAN DAN SARAN</b>	
5.1 Kesimpulan.....	49
5.2 Saran.....	50
<b>DAFTAR PUSTAKA.....</b>	<b>51</b>
<b>LAMPIRAN.....</b>	<b>56</b>

## DAFTAR TABEL

Tabel	Halaman
1. <i>Confusion Matrix</i> .....	20
2. Nilai Rata-Rata Akurasi <i>Kernel</i> Linear <i>Testing</i> Dataset.....	28
3. Nilai Rata-Rata Akurasi <i>Kernel</i> Polinomial <i>Testing</i> Dataset.....	30
4. Nilai Rata-Rata Akurasi <i>Kernel</i> RBF <i>Testing</i> Dataset.....	33
5. Nilai Akurasi Fungsi <i>Kernel</i> dengan Parameter Terbaik.....	35
6. <i>Confussion Matrix</i> Fungsi <i>Kernel</i> RBF 60 <i>Split Testing</i> Dataset.....	36
7. <i>Confussion Matrix</i> Fungsi <i>Kernel</i> RBF 70 <i>Split Testing</i> Dataset.....	36
8. <i>Confussion Matrix</i> Fungsi <i>Kernel</i> RBF 80 <i>Split Testing</i> Dataset.....	36
9. Nilai Akurasi Fungsi <i>Kernel</i> RBF pada Data Penyakit Jantung.....	37
10. Data Bangkitan Simulasi Variabel <i>Fasting Blood Sugar</i> dan Variabel <i>Resting ECG</i> .....	39

## DAFTAR GAMBAR

Gambar	Halaman
1. Model <i>Supervised Learning</i> .....	5
2. Beberapa Kemungkinan Batas Keputusan Dataset.....	9
3. Representasi <i>Hyperplanes</i> .....	10
4. <i>Hard Maximum Margin Separating Hyperplane</i> .....	11
5. <i>Soft Margin Support Vector Machine</i> .....	14
6. <i>Hyperplane</i> pada <i>Kernel Support Vector Machine</i> .....	16
7. <i>Kernel</i> Linear.....	18
8. <i>Kernel</i> Polinomial.....	18
9. Diagram Alir untuk Tahapan Simulasi.....	26
10. Diagram Alir untuk Tahapan Analisis Data Penyakit Jantung.....	27
11. Grafik Parameter <i>Cost</i> dengan Nilai Akurasi pada <i>Kernel</i> Linear <i>Testing</i> Dataset.....	29
12. Grafik <i>Split</i> data dengan Nilai Akurasi pada <i>Kernel</i> Linear <i>Testing</i> Dataset.....	29
13. Grafik Nilai Akurasi <i>Kernel</i> Polinomial dengan Parameter <i>Degree</i> 1.....	31
14. Grafik Nilai Akurasi <i>Kernel</i> Polinomial dengan Parameter <i>Cost</i> 1.....	32
15. Grafik Nilai Akurasi <i>Kernel</i> Polinomial dengan Parameter <i>Cost</i> 1 dan Parameter <i>Degree</i> 1.....	32

16. Grafik Nilai Akurasi <i>Kernel</i> RBF dengan Parameter <i>Gamma</i> 0,1.....	34
17. Grafik Nilai Akurasi <i>Kernel</i> RBF dengan Parameter <i>Cost</i> 0,1.....	34
18. Grafik Nilai Akurasi <i>Kernel</i> Polinomial dengan Parameter <i>Cost</i> 0,1 dan Parameter <i>Gamma</i> 0,1.....	35
19. <i>Scatterplot</i> Data Bangkitan Simulasi Variabel <i>Fasting Blood Sugar</i> dan Variabel <i>Resting ECG</i> .....	39
20. <i>Hyperplanes</i> Fungsi <i>Kernel</i> Linear.....	42

## I. PENDAHULUAN

### 1.1 Latar Belakang Penelitian

*Support Vector Machine* (SVM) adalah salah satu metode *machine learning* untuk menemukan fungsi pengklasifikasi yang dapat membagi data menjadi dua kelas yang berbeda (Vapnik, 1995). Prinsip SVM, yakni upaya dalam mencari *hyperplane* terbaik yang berfungsi sebagai pemisah antara dua *class* atau multi *class* pada *input space*. (Nugraha, *et. al.*, 2019). Metode SVM dengan fungsi *kernel* perlu memetakan data asli berdasarkan dimensinya menjadi dimensi yang berbeda, yakni berdimensi yang relatif lebih tinggi.

Menurut Lu, *et. al.* (2015), SVM memiliki banyak keunggulan jika dibandingkan dengan algoritma lain, yaitu ketika diterapkan dalam menyelesaikan masalah pengenalan pola sampel kecil, *nonlinier*, dan berdimensi tinggi. Selain itu, keunggulan SVM adalah tidak semua data *train* dilibatkan pada saat *training* (Prasetyo, 2014). Metode SVM menyimpan sebagian kecil data *train* pada saat prediksi, sedangkan metode ANN dan *nearest neighbor* menyimpan semua data *train*. Menurut Cervantes, *et. al.* (2020), SVM merupakan metode yang lebih unggul jika dibandingkan dengan metode *supervised learning* lainnya, sehingga SVM menjadi metode yang paling populer.

Sebelumnya beberapa penelitian mengkaji mengenai nilai akurasi terbaik berdasarkan pengujian besarnya persentase dataset dan perbandingan fungsi *kernel*, yakni Nachev & Teodosiev (2015) dan Widayani & Harliana (2021). Penelitian Nachev & Teodosiev (2015), diperoleh hasil bahwa nilai akurasi pengujian pada 20% data *training* memiliki nilai akurasi lebih tinggi jika dibandingkan dengan persentase data *training* lainnya. Sedangkan, pada penelitian

Widayani & Harliana (2021), diperoleh hasil bahwa 60% data *training* dan 40% data *testing* diperoleh nilai akurasi yang lebih tinggi. Dengan demikian, semakin besar data *training* yang digunakan tidak selalu meningkatkan nilai akurasi. Selain itu, pengujian perbandingan fungsi *kernel* pada kedua penelitian tersebut diperoleh hasil bahwa fungsi *kernel Radial Basis Function (RBF)* mempunyai nilai akurasi yang tinggi jika dibandingkan dengan fungsi *kernel* lainnya.

Berkembangnya teknologi dan ilmu pengetahuan pada saat ini metode SVM telah banyak digunakan pada berbagai bidang medis. Salah satu data yang dapat diterapkan dengan SVM adalah data penderita penyakit jantung. Beberapa faktor yang dipertimbangkan untuk mempengaruhi penyakit jantung, yaitu delapan faktor signifikan. Adapun faktor signifikan tersebut, yakni *age*, *chest pain type*, *resting blood pressure*, *cholesterol*, *fasting blood sugar*, *resting heart rate*, *maximum heart rate*, dan *exercise induced angina* (Nahar, *et.al.*, 2013). Salah satu peneliti yang menggunakan metode SVM dan data penderita penyakit jantung adalah Sandhya (2020). Penelitian tersebut membandingkan hasil dari algoritma SVM dengan algoritma *machine* lainnya. Diperoleh hasil bahwa algoritma SVM memberikan akurasi, spesifisitas dan sensitivitas yang lebih baik jika dibandingkan dengan algoritma *machine learning* lainnya.

Berdasarkan penjabaran di atas, maka penelitian ini akan dikaji tentang perbandingan fungsi *kernel* pada SVM melalui simulasi data dengan berbagai jumlah sampel yang berbeda dan menerapkannya pada klasifikasi penderita penyakit jantung.

## 1.2 Tujuan Penelitian

Adapun tujuan penelitian ini adalah:

1. Mengetahui fungsi *kernel* terbaik yang menghasilkan akurasi tertinggi pada SVM melalui studi simulasi.
2. Menerapkan metode SVM untuk klasifikasi penderita penyakit jantung.

### **1.3 Manfaat Penelitian**

Manfaat penelitian ini adalah diharapkan dapat menjadi rujukan/referensi bagi penelitian-penelitian sejenis dan menambah keilmuan bagi penelitian-penelitian di bidang klasifikasi data khususnya metode *support vector machine*.

## II. TINJAUAN PUSTAKA

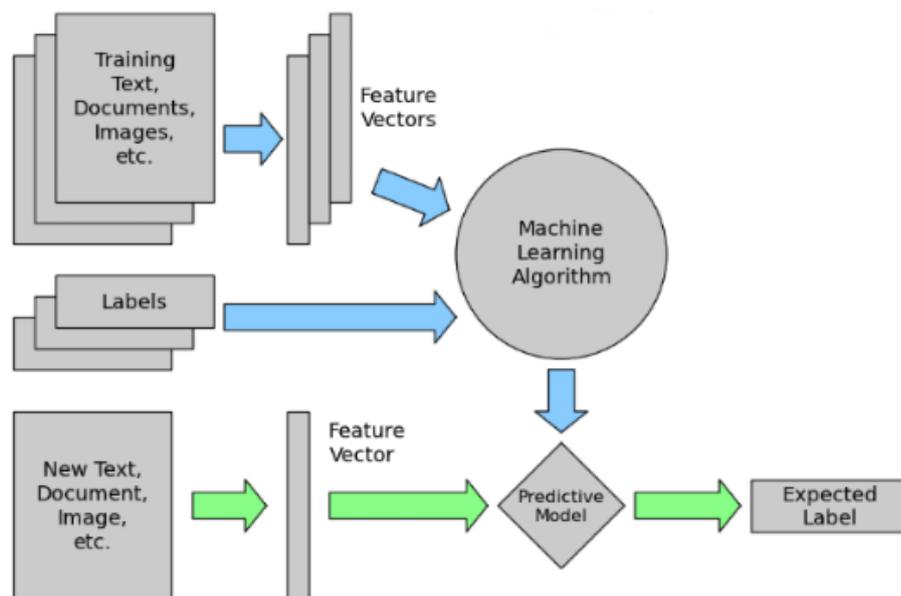
### 2.1 *Machine Learning*

*Machine learning* merupakan salah satu cabang gabungan antara pengaplikasian ilmu komputer dan algoritma matematika melalui *learning* yang bersumber dari data, sehingga diperoleh prediksi di waktu kedepan (Goldberg & Holland, 1988). Proses *learning* tersebut merupakan salah satu upaya dalam memperoleh kecerdasan dari dua tahap, yakni (*training*) dan pengujian (*testing*) (Huang, *et.al.*, 2006). Pada *training*, sampel data *training* diambil sebagai input fitur yang akan ditelaah oleh algoritma *learning* dan membuat model *learning* (Dhage & Raina, 2016). Sedangkan, pada *testing* model *learning* yang menggunakan mesin eksekusi untuk menjadikan prediksi untuk pengujian.

*Machine learning* dipisah menjadi tiga, yakni *supervised learning*, *unsupervised learning*, *reinforcement learning* (Somvanshi & Chavan, 2016). Menurut Kotsiantis (2007), jika contoh berlabel yang diketahui (sesuai output benar) maka *learning* disebut dengan *supervised*. Hal tersebut berbeda dengan *unsupervised learning*, yakni contoh tidak berlabel. Sedangkan, *reinforcement learning* mempunyai ide bahwa harus mengatasi tujuan tanpa adanya notifikasi dari komputer secara jelas jika tujuan tersebut telah tercapai (Das & Nene, 2017). *Supervised learning* bertujuan untuk memprediksi hasil berdasarkan *input*, sedangkan tujuan *unsupervised learning* adalah menjelaskan hubungan (*associations*), serta pola diantara data-data *input*.

## 2.2 Supervised Learning

*Supervised learning* dikelompokkan menjadi dua masalah, yakni klasifikasi dan regresi. Jika masalah klasifikasi maka variabel terikat berbentuk kategori, contohnya adalah merah dan biru atau penyakit dan tidak ada penyakit. Sedangkan, jika masalah regresi maka variabel terikat adalah nilai real, contohnya adalah dollar atau berat (Brownlee, 2016).



Gambar 1. Model *Supervised Learning* (Dhage & Raina, 2016).

Menurut Nasteski (2017), Gambar 1 algoritma membuat perbedaan antara data yang diamati  $X$ , yaitu data *training*, banyak kasus data terstruktur diberikan ke model selama proses *training*. Pada proses ini, algoritma *supervised learning* membangun model prediksi. Setelah *training*, model yang memprediksi paling banyak kemungkinan label untuk *set* sampel baru  $X$  di kumpulan *testing*. Langkah selanjutnya adalah diklasifikasikan tergantung pada sifat target  $y$ , yakni sebagai klasifikasi atau regresi.

## 2.3 Support Vector Machine

### 2.3.1 Definisi Support Vector Machine

*Support Vector Machine* (SVM) merupakan algoritma klasifikasi *supervised learning* yang dibuat oleh Vladimir Vapnik. Metode tersebut digunakan untuk memprediksi kelas berdasarkan proses *training* (Ernawati, 2019). Menurut Han, *et. al.* (2012), SVM dikembangkan oleh Vapnik, bersama Bernhard Boser dan Isabelle Guyon pada tahun 1992.

Dalam literatur SVM, variabel bebas disebut atribut, dan atribut berubah yang digunakan untuk menentukan *hyperplane* disebut fitur. Tugas dalam menentukan representasi yang paling sesuai adalah dikenal sebagai seleksi fitur. Satu *set* fitur yang menggambarkan satu masalah (misalnya, deretan nilai-nilai prediktor) disebut vektor. Vektor-vektor di dekat *hyperplane* adalah *support vector* (Suwardika, 2017).

### 2.3.2 Karakteristik Support Vector Machine

Menurut Prasetyo (2013), karakteristik SVM dapat diringkas sebagai berikut :

1. Model yang diperoleh selalu sama dan solusi dengan margin maksimal.
2. Proses *training* pada SVM tidak sebanyak ANN. Pada umumnya kinerja SVM yang lebih baik jika dibandingkan dengan ANN.
3. Pada kasus data yang tidak dapat dipisahkan secara linear maka dapat menggunakan fungsi *kernel* SVM.

### 2.3.3 Kelebihan Support Vector Machine

Adapun beberapa kelebihan dari metode SVM adalah sebagai berikut:

1. Generalisasi

Menurut Vapnik (1995), *generalization error* dipengaruhi oleh dua faktor, yaitu *error* terhadap *training set* dan satu faktor lagi yang dipengaruhi oleh dimensi VC (Vapnik Chervokinensis). Strategi *learning* pada *neural*

*network*, serta metode *machine learning* lainnya diutamakan untuk meminimalkan *error* pada *training set*. Strategi tersebut dikenal dengan *Empirical Risk Minimization* (ERM). Pada SVM, selain meminimalkan *error* pada *training set*, juga meminimalkan faktor kedua. Strategi ini disebut *Structural Risk Minimization* (SRM). Metode SVM dibuat dengan memilih *hyperplane* dengan margin terbesar. Berbagai analisis empiris menunjukkan bahwa strategi SRM pada SVM memberikan *generalization error* yang lebih kecil jika dibandingkan dengan strategi ERM pada *neural network* maupun metode yang lainnya. Selain itu, menurut Basari, *et. al.* (2013), SVM memiliki kemampuan generalisasi yang baik dalam kasus dengan sampel kecil.

## 2. *Curse of dimensionality*

*Curse of dimensionality* merupakan salah satu masalah yang dihadapi suatu metode *pattern recognition* dalam mengestimasi parameter dikarenakan jumlah sampel data yang relatif lebih sedikit dibandingkan dengan dimensional ruang vektor data tersebut. Semakin tingginya dimensi ruang vektor yang diolah maka diperlukan jumlah data dalam proses *learning*. Pada kenyataannya data yang diolah berjumlah terbatas, dan untuk mengumpulkan data yang lebih banyak, maka tidak sulit dilaksanakan karena adanya kendala biaya dan permasalahan teknis. Dengan demikian, jika metode “terpaksa” harus bekerja pada data yang berjumlah relatif sedikit jika dibandingkan dimensinya, maka proses estimasi parameter menjadi sangat sulit.

Vapnik membuktikan bahwa tingkat generalisasi itu tidak dipengaruhi oleh dimensi dari *input* vektor (Vapnik, 1999). Hal tersebut sebagai landasan bahwa SVM merupakan salah metode yang tepat digunakan untuk menyelesaikan permasalahan berdimensi tinggi, dan keterbatasan pada jumlah sampel data.

### 3. *Feasibility*

Metode SVM dapat diimplementasikan relatif mudah. Hal tersebut disebabkan oleh proses penentuan *support vector* dapat dirumuskan dalam *QP problem* (Nugroho, *et. al.*, 2003).

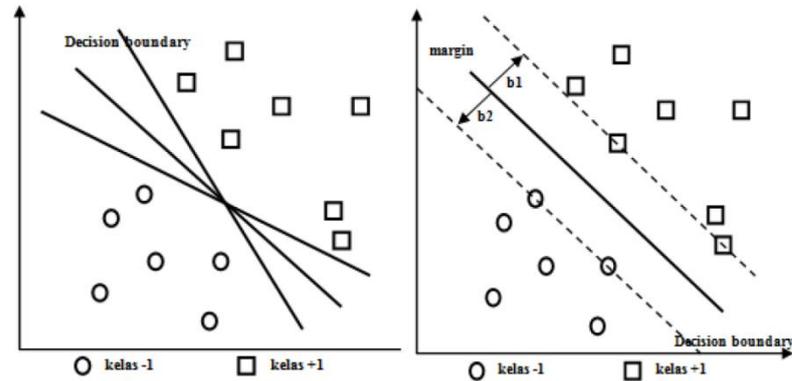
### 4. Landasan Teori

Sebagai metode yang didasarkan statistik, sehingga SVM mempunyai landasan teori yang dapat dianalisis dengan jelas.

#### 2.3.4 *Hard Margin*

Metode SVM dapat diterapkan pada hasil yang berkelanjutan, biner, kategorikal, logistik, dan multinomial dengan membentuk margin *hyperplane* (Guenther & Schonlau, 2016) (Aykanat, *et.al.*, 2017). Margin merupakan jarak antara *hyperplane* dengan *pattern* terdekat dari masing-masing *class*. *Pattern* terdekat disebut sebagai *support vector*. *Pattern* merupakan anggota dari dua buah kelas, yakni +1 dan -1 dan berbagi alternatif garis pemisah (*discrimination boundaries*) (Suwardika, 2017).

Menurut Gunn (1998), tujuan SVM adalah untuk mendapatkan *hyperplane* terbaik yang memisahkan kedua kelas. Mendapatkan fungsi pemisah (*classifier/hyperplane*) terbaik sama dengan memaksimalkan jarak antara *hyperplane* dengan *pattern* terdekat dari masing-masing kelas (margin). *Hyperplane* terbaik adalah *hyperplane* yang berada di tengah-tengah antara dua *set* objek dari dua kelas (Ritonga & Purwaningsih, 2018). Menurut Kowalczyk (2017), menentukan *hyperplane* terbaik untuk memberi jarak atau pemisah antara dua kelas yang telah ditentukan. Upaya menemukan lokasi *hyperplane* ini merupakan inti dari proses *training* SVM (Cristianini & Shawe-Taylor, 2000).



Gambar 2. Beberapa Kemungkinan Batas Keputusan Dataset (Prasetyo, 2014).

Berdasarkan Gambar 2 memperlihatkan beberapa data yang tergabung dalam dua buah kelas data, yaitu +1 dan -1. Data yang tergabung dalam kelas -1 dilambangkan dengan lingkaran, sedangkan data di kelas +1 disimbolkan dengan persegi (Rebentrost, *et. al.*, 2014).

Tiap data dinotasikan dengan  $x_i \in \mathcal{R}^D, i = 1, 2, \dots, N$ . Dimana  $N$  adalah banyaknya data. *Positive class* dinotasikan sebagai +1 dan *negative class* dinotasikan sebagai -1. Dengan demikian, setiap data dan label *class*-nya dinotasikan sebagai:  $y_i \in \{-1, +1\}$ . Diasumsikan bahwa kedua *class* tersebut dapat dipisahkan secara sempurna oleh *hyperplane* di *D-dimensional feature space* (Nugroho, 2007).

Persamaan *hyperplane* diberikan oleh dua parameter, bobot vektor  $\mathbf{w}$  nilai real dari dimensi yang sama dengan fitur *input* vektor  $\mathbf{x}$ , dan bias  $b$ , yakni:

$$f(\mathbf{x}) = \mathbf{x} \cdot \mathbf{w} + b = 0,$$

dimana:

$$\mathbf{x} \cdot \mathbf{w} = \sum_i x_i w_i,$$

dengan:

$$\sum_i x_i w_i \text{ yang berarti } x_1 w_1 + x_2 w_2 + \dots + x_D w_D.$$

$D$  adalah bilangan dari dimensi fitur vektor  $\mathbf{x}$ .

Konsep penting dalam mendefinisikan *classifier* yang bersifat linear, yaitu *dot product* atau *inner product* atau *scalar product* diantara dua vektor. Pada beberapa buku, *dot-product* dilambangkan sebagai  $\mathbf{x} \cdot \mathbf{w}$ . Kedua vektor harus memiliki dimensi yang sama. Jika berbeda maka *dot product* tidak dapat didefinisikan.

Label yang diprediksi untuk beberapa *input* fitur  $\mathbf{x}$  diberikan sebagai berikut:

$$\text{sign}(f(\mathbf{x})) = \text{sign}(\mathbf{x} \cdot \mathbf{w} + b) = y_i, \forall i.$$

dimana *sign* adalah operator matematika yang mengambil nilai berapapun sebagai *input* dan menghasilkan +1 jika *input*-nya adalah bilangan positif atau -1 jika *input*-nya adalah bilangan negatif (Burkov, 2019).

Dalam memisahkan kedua kelas dengan maka pemisah sejajar ditentukan oleh:

$$\mathbf{x} \cdot \mathbf{w} + b \geq +1 ; \text{ untuk } y_i = +1.$$

$$\mathbf{x} \cdot \mathbf{w} + b \leq -1 ; \text{ untuk } y_i = -1.$$

Berdasarkan di atas menyatakan bahwa setiap titik data harus berada di sisi margin yang tepat. Hal tersebut dapat dituliskan sebagai berikut:

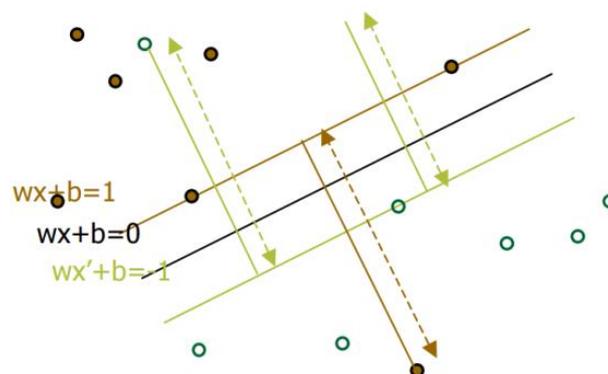
$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1; \forall i.$$

dimana:

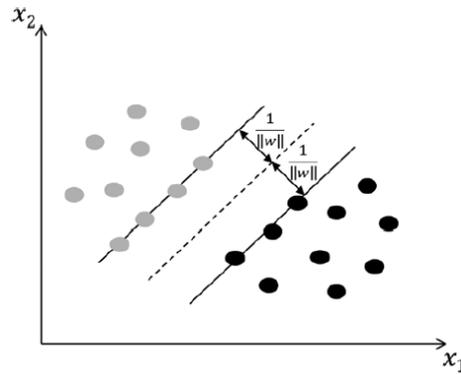
$\mathbf{x}_i$  : inisialisasi data ke-i berbentuk vektor.

$b$  : nilai bias.

$y_i$  : *class* data ke-i.



Gambar 3. Representasi *Hyperplanes* (Vapnik, et. al., 1997).



Gambar 4. *Hard Maximum Margin Separating Hyperplane* (Awad & Khanna, 2015).

Pada Gambar 4 optimal margin dapat dihitung dengan memaksimalkan jarak antara *hyperplane* dan *pattern terdekat*. Hal tersebut mengakibatkan memaksimalkan nilai:

$$\frac{1}{\|\mathbf{w}\|} \quad (1)$$

Dimana:

$\|\mathbf{w}\|$  : *norm* dari *weight vector*  $\mathbf{w}$ .

Berdasarkan Gambar 4 bidang pemisah dapat dirumuskan sebagai berikut (Awad & Khanna, 2015):

$$m = \frac{1}{\|\mathbf{w}\|} + \frac{1}{\|\mathbf{w}\|} = \frac{2}{\|\mathbf{w}\|}$$

dimana:

$m$  : jarak antara dua bidang.

Persamaan (1) ekuivalen dengan persamaan dibawah ini, yakni meminimumkan nilai:

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

(Mohamed, 2017),

dimana:

$$\|\mathbf{w}\|^2 = \mathbf{w}^T \cdot \mathbf{w} = w_1^2 + w_1^2 + \dots + w_p^2$$

Dengan kendala:

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 \geq 0; \forall i = 1, 2, \dots, N. \quad (3)$$

Optimisasi ini dapat diselesaikan dengan Lagrange *multiplier*, yakni sebagai berikut:

$$L = \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^N a_i [y_i (\mathbf{x}_i \cdot \mathbf{w} + b) - 1].$$

$$L = \frac{1}{2} \mathbf{w}^T \cdot \mathbf{w} - \sum_{i=1}^N a_i y_i \mathbf{x}_i \cdot \mathbf{w} - \sum_{i=1}^N a_i y_i b + \sum_{i=1}^N a_i \quad (4)$$

Dimana  $a$  adalah Lagrange *multiplier*, yang bernilai 0 atau positif, yakni  $a_i \geq 0$ . Menurut (Hamel, 2009), nilai optimal dari persamaan (4) dapat dihitung dengan meminimalkan  $L$  terhadap  $\mathbf{w}$  dan  $b$ , yakni:

$$\frac{\partial L}{\partial \mathbf{w}} = 0.$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^N a_i y_i \mathbf{x}_i = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^N a_i y_i \mathbf{x}_i. \quad (5)$$

$$\frac{\partial L}{\partial b} = 0.$$

$$\frac{\partial L}{\partial b} = -\sum_{i=1}^N a_i y_i \Rightarrow \sum_{i=1}^N a_i y_i = 0. \quad (6)$$

Karena Lagrange *multiplier*  $a_i$  tidak diketahui, maka  $\mathbf{w}$  dan  $b$  masih belum dapat dicari solusinya. Optimasi dapat dilakukan dengan memaksimalkan  $L$  terhadap  $a_i$  dengan substitusi persamaan (5) dan (6) ke dalam persamaan (4), yakni:

$$L = \frac{1}{2} \left( \sum_{i=1}^N a_i y_i \mathbf{x}_i \right) \cdot \left( \sum_{j=1}^N a_j y_j \mathbf{x}_j \right) - \sum_{i=1}^N a_i y_i \mathbf{x}_i \cdot \left( \sum_{j=1}^N a_j y_j \mathbf{x}_j \right) + \sum_{i=1}^N a_i.$$

$$L = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j. \quad (7)$$

Dengan syarat:

$$a_i \geq 0 \quad (i = 1, 2, \dots, N), \quad \sum_{i=1}^N a_i y_i = 0,$$

dimana:

$a_i a_j$  : Lagrange *multiplier* yang berkorespondensi dengan  $\mathbf{x}_i \cdot \mathbf{x}_j$ .

$\mathbf{x}_i \cdot \mathbf{x}_j$  : *dot product* dua data dalam data *training*.

$N$  : jumlah data yang menjadi *support vector*.

Menghitung nilai  $b$ , yakni untuk suatu pengamatan ke- $i$  sebagai berikut:

$$y_i (\mathbf{x} \cdot \mathbf{w} + b_i) = 1 \quad (8)$$

Kalikan kedua sisi dengan  $y_i$  pada persamaan (8), sehingga diperoleh:

$$y_i^2 (\mathbf{x} \cdot \mathbf{w} + b_i) = y_i,$$

dimana:

$y_i^2 = 1$  maka:

$$b_i = y_i - \mathbf{x} \cdot \mathbf{w}.$$

Kemudian, dapat dihitung  $b$  sebagai berikut:

$$b = \frac{1}{S} \sum_{i=1}^S (y_i - \mathbf{x} \cdot \mathbf{w}),$$

dengan:

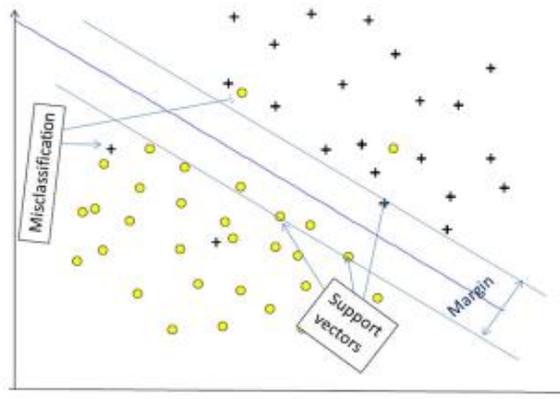
$S$  : banyaknya *support vectors*.

Solusi dari masalah optimisasi ini, diberikan oleh  $\mathbf{w}$  dan  $b$  disebut statistik model. Tujuan dari algoritma *learning* pada SVM adalah untuk melihat pengaruh dari dataset dan menemukan nilai optimal parameter  $\mathbf{w}$  dan  $b$ .

Masalah *hard margin* SVM adalah jika dataset tidak dapat dipisahkan secara linear maka itu menjadi tidak mungkin untuk mengklasifikasikan data, yaitu tidak ditemukan *hyperplane* pemisah dan adanya *noise* atau *outlier* sangat mempengaruhi *margin*. Penjelasan sebelumnya didasarkan oleh asumsi bahwa kedua belah *class* dapat terpisah secara sempurna oleh *hyperplane*. Akan tetapi, yang kerap terjadi adalah kedua buah *class* tersebut tidak dapat dipisahkan secara sempurna sehingga dapat menghasilkan permasalahan pada proses optimisasi dan tidak dapat diselesaikan. Metode SVM dapat mengatasi beberapa data yang tidak dapat diklasifikasikan secara benar (*misclassification error*), yakni dengan dilakukan margin lunak (*soft margin*).

### 2.3.5 Soft Margin

Cortes dan Vapnik memperkenalkan variabel *slack* ke SVM *hard margin* untuk memungkinkan beberapa kesalahan klasifikasi. Hal tersebut merupakan *soft margin* SVM (Vapnik, 1995).



Gambar 5. *Soft Margin Support Vector Machine.*

Pada Gambar 5 menggambarkan contoh *soft margin* SVM, yakni data dua dimensi yang tidak dapat dipisahkan. Hal tersebut disebabkan oleh data yang tidak sepenuhnya dapat dipisahkan maka SVM memungkinkan beberapa kesalahan klasifikasi (*misclassification*) untuk menemukan pemisah *hyperplane*.

Untuk merealisasikan *soft margin*, diperkenalkan variabel *slack* atau variabel kesalahan, yakni  $\xi_i$  ( $\xi_i \geq 0$ ) dimana  $i = 1, 2, \dots, N$  dengan satu variabel *slack* untuk setiap data *training* (Scholkopf & Smola, 2002). *Soft margin* memodifikasi pertidaksamaan (3) dengan menambahkan variabel *slack*, sehingga menjadi pertidaksamaan sebagai berikut (Nugroho, 2007) :

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) \geq 1 - \xi_i; \forall i = 1, 2, \dots, N. \quad (9)$$

Demikian juga persamaan (2), sehingga diperoleh persamaan di bawah ini:

$$\min \tau(\mathbf{w}) = \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i. \quad (10)$$

Dengan kendala:

$$y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i \geq 0; \forall i = 1, 2, \dots, N.$$

Peminimalisasian fungsi tujuan pada persamaan (10) berdasarkan syarat pada pertidaksamaan (9) akan menyediakan kemungkinan margin terbesar dalam kasus dimana suatu kesalahan klasifikasi diizinkan. Parameter *cost* / *C* dipilih berdasarkan model SVM pada proses validasi data *training*.

Optimisasi dapat diselesaikan dengan Lagrange *multipliers*, yakni:

$$L = \left[ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \right] - \sum_{i=1}^N a_i [y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i] - \sum_{i=1}^N \mu_i \xi_i.$$

$$L = \left[ \frac{1}{2} \|\mathbf{w}\|^2 + C \sum_{i=1}^N \xi_i \right] - \sum_{i=1}^N a_i y_i \mathbf{x}_i \cdot \mathbf{w} - \sum_{i=1}^N a_i y_i b + \sum_{i=1}^N a_i - \sum_{i=1}^N a_i \xi_i - \sum_{i=1}^N \mu_i \xi_i \quad (11)$$

Syarat pertidaksamaan dapat ditransformasi menjadi syarat yang berupa persamaan menggunakan kondisi Karush Kuhn Tucker (KKT) berikut:

$$\xi_i \geq 0, a_i \geq 0, \mu_i \geq 0.$$

$$a_i [y_i(\mathbf{x}_i \cdot \mathbf{w} + b) - 1 + \xi_i] = 0.$$

$$\mu_i \xi_i = 0.$$

Turunan pertama dari persamaan (11) terhadap  $\mathbf{w}$ ,  $b$ , dan  $\xi_i$  yang disamadengankan nol akan menghasilkan persamaan berikut:

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{w} - \sum_{i=1}^N a_i y_i \mathbf{x}_i = 0 \Rightarrow \mathbf{w} = \sum_{i=1}^N a_i y_i \mathbf{x}_i. \quad (12)$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^N a_i y_i = 0. \quad (13)$$

$$\frac{\partial L}{\partial \xi_i} = C - a_i - \mu_i = 0 \Rightarrow \mu_i = C - a_i. \quad (14)$$

Optimasi dilakukan dengan mensubstitusi persamaan (12), (13), (14) ke persamaan (11), yakni sebagai berikut:

$$L = \left[ \frac{1}{2} \left( \sum_{i=1}^N a_i y_i \mathbf{x}_i \right) \cdot \left( \sum_{j=1}^N a_j y_j \mathbf{x}_j \right) \right] - \sum_{i=1}^N a_i y_i \mathbf{x}_i \cdot \left( \sum_{j=1}^N a_j y_j \mathbf{x}_j \right) + \sum_{i=1}^N a_i.$$

$$L = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j. \quad (15)$$

Berdasarkan penjabaran halaman sebelumnya terlihat bahwa persamaan (15) dan persamaan (7) terlihat sama. Namun, syaratnya yang berbeda, yakni:

$$0 \leq a_i \leq C, \sum_{i=1}^N a_i y_i = 0.$$

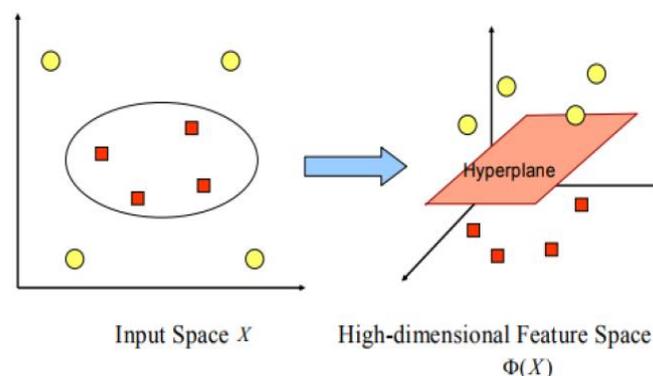
Permasalahan di atas dapat diselesaikan secara numerik menggunakan *quadratic programming* untuk mencari nilai  $a_i$ . Pengamatan  $\mathbf{x}_i$  untuk  $a_i > 0$ , yaitu suatu titik (*support vector*) yang berada di atas atau di dalam margin ketika *soft margin* digunakan (Scholkopf dan Smola, 2002).

Meminimalkan persamaan (10) akan memaksimalkan jarak margin antar kelas. Dengan adanya penambahan *slack* variabel  $\xi_i$  akan meminimalisir *misclassification error* (jumlah kesalahan pada klasifikasi). Parameter C berfungsi

untuk mengendalikan *tradeoff* atau kesepadanan antara *margin* dan *classification error* ( $\xi$ ). Menurut Nugroho (2007), jika semakin besar nilai  $C$  maka semakin besar penalti yang dikenakan untuk tiap *classification error*, dan juga jika nilai  $C$  besar maka dalam menemukan *hyperplane* dan *margin* sehingga terdapat beberapa *pattern* dalam *margin* yang mempunyai model sangat kompleks dengan hanya mempunyai *margin* kecil dan bisa juga *pattern* tidak mudah dipisahkan. Sedangkan, jika nilai  $C$  yang terlalu kecil maka kesalahan yang lebih tinggi pada data *training* dan juga sebaliknya, maka akan memiliki *margin* yang besar sehingga kemungkinan memiliki model yang lebih baik dan kuat.

### 2.3.6 Kernel Support Vector Machine

Pada awalnya SVM dirancang untuk menyelesaikan masalah untuk klasifikasi linear. Menurut Pratiwi & Setyawan (2021), kebanyakan kasus pada dunia nyata jarang terdapat kasus yang bersifat linear. Salah satu metode dalam menyelesaikan kasus tersebut adalah dengan fungsi *kernel*. Metode SVM dengan fungsi *kernel* bertujuan untuk menghubungkan *input data training* ke fitur dan identifikasi dimensi ruang yang lebih luas *hyperplane*-nya sebagai ruang pemisah (Indriani & Muslim, 2019). Fungsi *kernel* digunakan untuk mentransformasikan data ke ruang berdimensi yang lebih tinggi, dan menjadikan data nonlinear dapat terpisah secara linear (Herwijayanti, *et. al.*, 2018). Oleh sebab itu, SVM dapat digeneralisasikan pada kasus nonlinear (Härdle, *et. al.*, 2014).



Gambar 6. *Hyperplane* pada *Kernel Support Vector Machine* (Nugroho, 2003).

Pada Gambar 6 di sebelah kiri terlihat *input space*  $X$  berisi beberapa data (kelas kuning lingkaran dan merah kotak) berdimensi dua dan tidak bisa dipisahkan secara linear. Sedangkan, di sebelah kanan memperlihatkan sebuah fungsi  $\Phi$  yang dapat memetakan data *input* ke dalam ruang vektor baru dengan dimensi 3 (lebih tinggi) dengan garis pemisah *hyperplane* yang linear (Pratiwi & Setyawan, 2021).

Persamaan *hyperplane* diberikan oleh dua parameter dengan transformasi  $\Phi$ :

$$f(\Phi(\mathbf{x})) = \Phi(\mathbf{x}) \cdot \mathbf{w} + b = 0.$$

Klasifikasi vektor  $\mathbf{w}$  pada transformasi *space*, dirumuskan sebagai berikut:

$$\mathbf{w} = \sum_{i=1}^N a_i y_i \Phi \mathbf{x}_i.$$

Optimisasi diselesaikan dengan Lagrange *multipliers*, yakni:

$$L = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j). \quad (16)$$

Pada umumnya transformasi  $\Phi$  tidak diketahui dan sangat sulit untuk dipahami maka perhitungan *dot product* dapat digantikan dengan fungsi *kernel*  $K\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j)$  yang mendefinisikan secara implisit transformasi  $\Phi$ . Proses transformasi dikenal dengan istilah “*Kernel Trick*” (Scholkopf dan Smola, 2002). Oleh sebab itu, *support vector* akan lebih mudah diperoleh dengan *kernel trick* ini. Kita tidak perlu mengetahui fungsi nonlinear  $\Phi$  karena telah ditransformasi menjadi  $\Phi(\mathbf{x}_i)\Phi(\mathbf{x}_j)$ , sehingga kita hanya perlu mengetahui fungsi *kernel*-nya. (Nugroho, 2003).

*Kernel Trick*, dapat dirumuskan sebagai berikut ini:

$$K(\mathbf{x}_i, \mathbf{x}_j) = \Phi(\mathbf{x}_i) \cdot \Phi(\mathbf{x}_j). \quad (17)$$

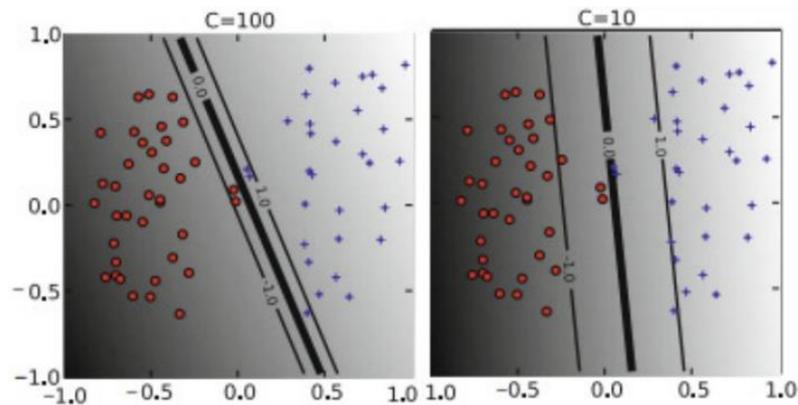
Optimisasi dapat dilakukan dengan *kernel trik* dilakukan dengan mensubstitusi persamaan (17) ke persamaan (16), sehingga diperoleh persamaan:

$$L = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j).$$

Berikut ini merupakan fungsi *kernel* yang digunakan untuk mengatasi masalah pada metode SVM, yaitu:

1. *Kernel Linear*

$$K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i \cdot \mathbf{x}_j.$$



Gambar 7. *Kernel Linear* (Ben-Hur & Weston, 2010).

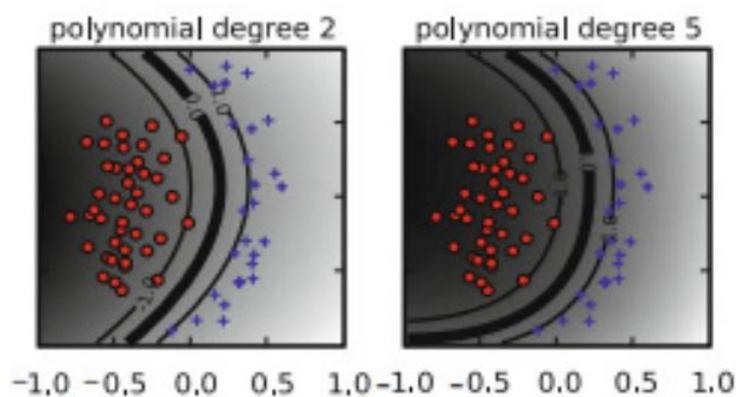
Pada Gambar 7 jika nilai *cost* yang rendah (10) maka nilai *margin error* yang rendah, memperlebar nilai *margin* dan mengabaikan *ignore point* yang dekat dengan *decision boundary*, dan juga sebaliknya jika nilai *cost* yang tinggi (100) maka *margin error* besar dan mempersempit *margin*.

## 2. *Kernel Polinomial*

$$K(\mathbf{x}_i, \mathbf{x}_j) = (\mathbf{x}_i \cdot \mathbf{x}_j + 1)^p.$$

Dimana:

$p$  : derajat polinomial.



Gambar 8. *Kernel Polinomial* (Ben-Hur & Weston, 2010).

Derajat atau *degree* pada *kernel* polinomial dapat mengendalikan fleksibilitas dari hasil klasifikasi, semakin tinggi nilai derajat pada *kernel* polinomial memungkinkan *decision boundary* yang lebih fleksibel (Ben-Hur & Weston,

2010). Menurut Kowalczyk (2017), menggunakan derajat polinomial yang terlalu tinggi dapat mengakibatkan *overfitting* atau model yang digunakan terlalu fokus pada data *training*, sehingga pengujian dengan data *testing* yang berbeda akan menyebabkan penurunan akurasi.

### 3. *Kernel Radial Basis Function* (RBF)

$$K(\mathbf{x}_i, \mathbf{x}_j) = \exp\left(-\gamma\|\mathbf{x}_i - \mathbf{x}_j\|^2\right), \gamma > 0.$$

*Kernel* RBF mempunyai dua parameter, yaitu *Gamma* dan *Cost*. Parameter *cost* sebagai parameter yang digunakan untuk menghindari misklasifikasi pada setiap sampel dalam *training* dataset. Parameter *Gamma* dapat menentukan seberapa jauh pengaruh dari satu sampel *training* dataset dengan nilai rendah berarti “jauh”, dan nilai tinggi berarti “dekat”. Jika nilai *gamma* rendah, maka titik yang berada jauh dari garis pemisah yang logis dalam perhitungan untuk garis pemisah. Sedangkan, ketika *gamma* tinggi berarti titik-titik berada di sekitar garis yang logis akan dipertimbangkan dalam perhitungan (Meyer & Wien, 2015).

## 2.4 Pemilihan Parameter pada *Support Vector Machine*

Menurut Maimon (2010), beberapa parameter dari algoritma SVM harus diperbaiki untuk mendapatkan tingkat kinerja yang tinggi, yaitu pemilihan kinerja fungsi *kernel* tergantung pada pilihan fungsi *kernel*, besaran parameter *kernel* dan penentuan parameter *cost*. Jika nilai penentuan parameter *cost* yang dipilih terlalu besar atau terlalu kecil, maka generalisasi SVM mungkin menurun. Prediksi menggunakan SVM sangat sensitif terhadap nilai parameter (Enri, 2018). Jika parameter *kernel* dan penentuan parameter yang tepat, maka kinerja SVM akan optimal. Oleh sebab itu, penggunaan parameter yang tepat diharapkan dapat meningkatkan akurasi SVM, serta merupakan salah satu langkah penting pada SVM sebagai metode klasifikasi.

Metode SVM akan beroperasi secara benar sepanjang fungsi *kernel*-nya tepat, meskipun tidak tahu pemetaan yang digunakan (Santosa, 2007). Menurut Hsu, *et al.* (2003), fungsi *kernel* yang direkomendasikan untuk diuji pertama kali adalah fungsi *kernel* RBF karena dapat memetakan hubungan tidak linear, RBF lebih *robust* terhadap *outlier* karena fungsi *kernel* RBF berada antara selang  $(-\infty, \infty)$ , sedangkan fungsi *kernel* yang lain memiliki rentang antara (-1 sampai dengan 1). Selain itu, fungsi *kernel* RBF juga efektif menghindari *overfitting* dengan memilih nilai yang tepat untuk parameter C dan  $\gamma$ .

## 2.5 Evaluasi Model

Evaluasi model dalam pemilihan metode pembagian dataset dan metode klasifikasi terbaik yang dilihat melalui ukuran kinerja klasifikasi. Ukuran kinerja tersebut dengan memperhatikan *confussion matrix*. *Confussion matrix* adalah alat yang digunakan untuk menganalisis seberapa baik atau seberapa akurat metode klasifikasi dapat mengenali objek pengamatan dari kelas yang berbeda (Maalouf & Siddiqi, 2014).

Sebagai contoh untuk *confusion matrix* dua kelas maka akan disebutkan menjadi kelas positif dan kelas negatif. *True* positif mengarah pada kelas positif yang diberi tanda dengan tepat oleh *classifier*, sementara *true* negatif adalah kelas negatif yang diberi tanda dengan tepat oleh *classifier*. Untuk *false* positif adalah kelas negatif yang diberi tanda dengan tidak tepat. Selanjutnya, *false* negatif adalah kelas positif yang diberi label dengan tidak tepat.

Tabel 1. *Confusion Matrix*

Kualitas Benar	Klasifikasi	
	Positif (+)	Negatif (-)
Positif Benar	<i>True</i> Positif (TP)	<i>False</i> Negatif (FN)
Negatif Benar	<i>False</i> Positif (FP)	<i>True</i> Negatif (TN)

Salah satu metode evaluasi, yakni akurasi. Akurasi adalah angka prediksi yang benar (atau salah) yang dibuat oleh model melalui kumpulan dari data. Pada umumnya akurasi dihitung dengan menggunakan uji independensi yang tidak selalu digunakan dalam *machine learning*. Berikut ini perhitungan dengan akurasi:

$$Accuracy = \frac{(TP+TN)}{TP+TN+FP+FN} \times 100\%.$$

### **III. METODOLOGI PENELITIAN**

#### **3.1 Waktu dan Tempat Penelitian**

Penelitian ini dilakukan pada semester genap tahun akademik 2022/2023 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Lampung.

#### **3.2 Data Penelitian**

##### **3.2.1 Data Simulasi**

Data simulasi yang digunakan sesuai dengan data penderita penyakit jantung pada Lampiran 4 (halaman ke 66). Pembangkitan data setiap variabel dapat dilihat secara detail pada Lampiran 2 (halaman 57 – 64).

##### **3.2.2 Data Penyakit Jantung**

Data real yang digunakan merupakan data sekunder yang diperoleh dari website <https://www.kaggle.com/datasets/firdaus9914/penyakit-jantung> . Data tersebut terdiri dari 12 variabel, yakni dengan 11 variabel independen/bebas dan 1 variabel dependen/terikat.

Berikut ini merupakan informasi dari masing-masing atribut:

1. *Age*: usia pasien [tahun].
2. *Gender*: jenis kelamin pasien [M : Pria, F : Wanita].
3. *Chest Pain Type*: tipe nyeri dada [TA : Angina Khas, ATA : Angina Atipikal, NAP: Nyeri Non-Anginal, ASY : Asimptomatik].

4. *Resting Blood Pressure*: mengistirahatkan tekanan darah [mm Hg].
5. *Cholesterol*: kolesterol [mm/dl].
6. *Fasting Blood Sugar* : gula darah puasa [2: jika FastingBS > 120 mg/dl, 1 : sebaliknya].
7. *Resting Electrocardiography (ECG)*: hasil elektrokardiogram istirahat [Normal : Normal, ST : memiliki kelainan gelombang ST-T (inversi gelombang T dan / atau elevasi ST atau depresi > 0,05 mV), LVH : menunjukkan kemungkinan atau pasti hipertrofi ventrikel kiri dengan kriteria Estes].
8. *Maximum Heart Rate*: detak jantung maksimum tercapai [Nilai numerik antara 60 dan 202].
9. *Exercise Induced Angina*: angina yang diinduksi latihan [Y : Ya, N : Tidak].
10. *Oldpeak*: oldpeak = ST [Nilai numerik diukur dalam depresi].
11. *ST Slope*: kemiringan segmen ST latihan puncak [Atas : *upsloping*, Datar : *flat*, Bawah : *downsloping*].
12. *Heart Disease*: kelas keluaran [1 : penyakit jantung, -1 : Normal].

### 3.3 Metode Penelitian

Penelitian ini menggunakan bantuan software Rstudio dengan versi R 4.2.2.

#### 3.3.1 Metode Penelitian Data Simulasi

Pada simulasi ini dibangkitkan data populasi sebanyak  $N = 4.000$  berdasarkan karakteristik data, yaitu terdiri dari 6 variabel kategorik dan 5 variabel kontinu. Adapun langkah-langkahnya adalah sebagai berikut:

- 1) Memisahkan data / mengelompokkan data berdasarkan variabel *heart disease*, yakni apakah termasuk penderita penyakit jantung atau tidak penderita penyakit jantung.
- 2) Melihat distribusi dari setiap variabel berdasarkan tipe data. Data bertipe kategorik, yakni variabel *gender*, *fasting blood sugar*, dan *exercise induced*

*angina* akan dibangkitkan data berdistribusi binomial, serta variabel *chest pain type*, *resting ECG*, dan *ST Slope* akan dibangkitkan data berdistribusi multinomial. Sedangkan, data bertipe kontinu, yakni variabel *age*, *resting blood pressure*, *cholesterol*, *maximum heart heart*, dan *oldpeak* akan dibangkitkan data berdasarkan apakah termasuk berdistribusi gamma / weibull / normal berdasarkan nilai *loglikelihood*, AIC, dan BIC. Uji distribusi semua variabel dapat dilihat pada Lampiran 2 (halaman 57 – 64).

- 3) Bangkitkan setiap variabel untuk setiap kelompok sebanyak  $N_1 = N_2 = 2.000$ .
- 4) Menggabungkan data penderita penyakit jantung atau tidak penderita penyakit jantung dan disimpan sebagai data populasi, sehingga  $N = N_1 + N_2 = 4.000$ .

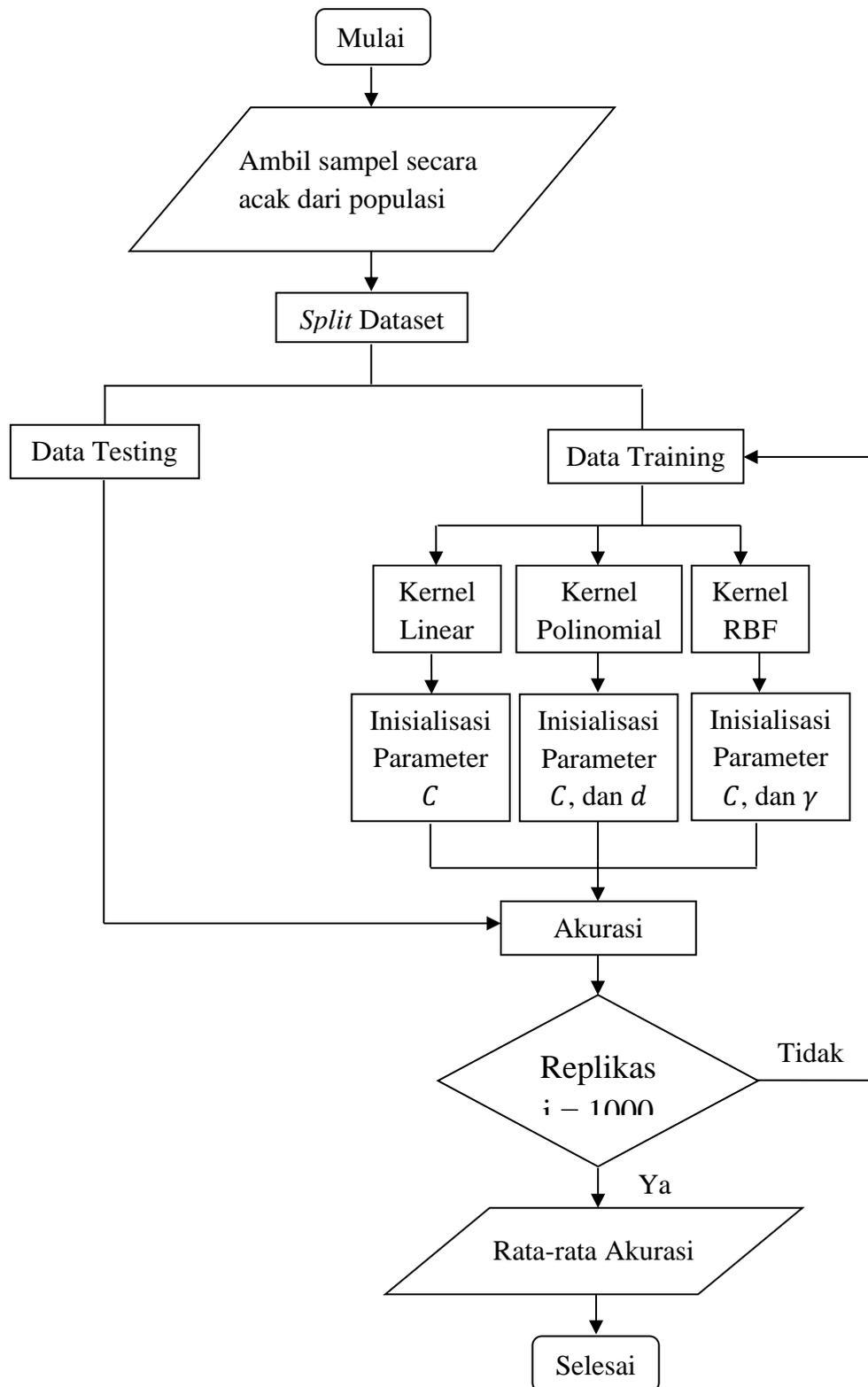
Tahapan simulasi adalah sebagai berikut:

1. Ambil sampel secara acak dari populasi sebanyak  $n = 400$ .
2. Membagi dataset dengan *split* dataset, yakni 60, 70, dan 80. *Split* dataset 60, yakni 60% data *training* dan 40% data *testing*. *Split* dataset 70, yakni 70% data *training* dan 30% data *testing*. *Split* dataset 80, yakni 80% data *training* dan 20% data *testing*.
3. *Processing* data, yaitu klasifikasi data menggunakan algoritma SVM. Pada langkah ini, diperoleh nilai akurasi parameter dari masing-masing *kernel*. Parameter yang digunakan, yakni  $cost = 0.1, 0.3, 0.5, 0.7, 1.0$ , untuk semua fungsi *kernel*,  $gamma = 0.1, 0.5, dan 1.0$  untuk fungsi *kernel* RBF, serta  $degree = 1, 2, 3, 4, dan 5$  untuk fungsi *kernel* polinomial.
4. Ulangi langkah 1 – 3 sebanyak 1000 kali.
5. Menghitung rata-rata nilai akurasi untuk setiap fungsi *kernel* dari 1000 kali ulangan.
6. Analisis hasil, yakni dengan membandingkan nilai rata-rata akurasi dari masing-masing fungsi *kernel* sehingga diperoleh nilai akurasi tertinggi pada parameter *kernel* terbaik.

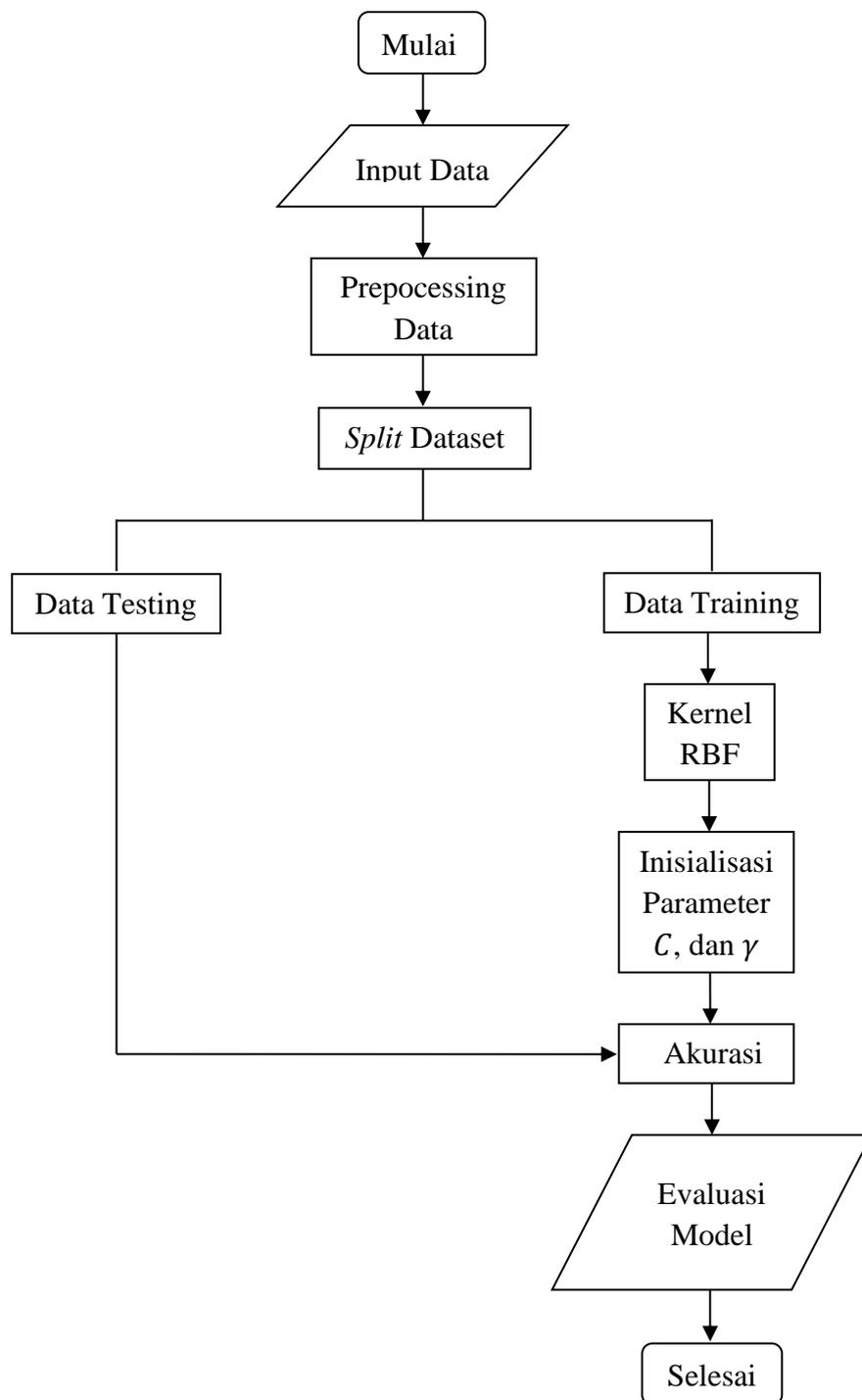
### 3.3.2 Metode Penelitian Analisis Data Penyakit Jantung

Berikut merupakan langkah-langkah penelitian pada data penyakit jantung:

1. *Preprocessing* data.
  - 1) Cek *Missing Value*, apabila terdapat *missing value* maka perlu diselesaikan data tersebut.
  - 2) *Encoding* data kategorik / transformasi data, yakni data berbentuk karakter maka diubah menjadi data bertipe kategorik yang memiliki tingkatan/level dari masing-masing kategorik.
  - 3) Menstandarisasi data / normalisasi data numerik. Normalisasi dilakukan dengan menggunakan normalisasi *scale*, yakni membagi setiap nilai dengan nilai maksimum pada data tersebut.
2. Membagi dataset dengan *split* dataset, yakni 60, 70, dan 80. *Split* dataset 60, yakni 60% data *training* dan 40% data *testing*. *Split* dataset 70, yakni 70% data *training* dan 30% data *testing*. *Split* dataset 80, yakni 80% data *training* dan 20% data *testing*.
3. *Processing* data, yaitu klasifikasi data menggunakan algoritma SVM dari fungsi *kernel* terbaik pada simulasi
4. Evaluasi model SVM, yakni diperoleh nilai akurasi terbaik sesuai dengan fungsi *kernel* terbaik.



Gambar 9. Diagram Alir untuk Tahapan Simulasi.



Gambar 10. Diagram Alir untuk Tahapan Analisis Data Penyakit Jantung.

## V. KESIMPULAN DAN SARAN

### 5.1 Kesimpulan

Berdasarkan hasil dan pembahasan maka diperoleh beberapa kesimpulan, yakni sebagai berikut:

1. Berdasarkan hasil studi simulasi diperoleh bahwa nilai parameter terbaik untuk setiap proporsi *split* dataset dan ketiga fungsi *kernel* adalah sebagai berikut:
  - a. Fungsi *kernel* RBF terbaik dengan nilai akurasi tertinggi
  - b. Proporsi *split* dataset yang terbaik adalah 80 *training* dan 20 *testing*
  - c. Parameter terbaik adalah  $cost = 1$ ,  $degree = 1$ , dan  $gamma = 0,5$ .
2. Berdasarkan analisis data penyakit jantung diperoleh model SVM terbaik adalah fungsi *kernel* RBF dan proporsi *split* dataset 80 dengan nilai akurasi sebesar 1. Dengan menggunakan fungsi *kernel* RBF dan proporsi *split* dataset tersebut dapat diperoleh parameter  $w$  dan  $b$ , yakni sebagai berikut:

$$w_{Age} = -9,429, w_{Gender(1)} = 4,223, w_{Gender(2)} = -4,223,$$

$$w_{Chest\ Pain\ Type(1)} = 6,729, w_{Chest\ Pain\ Type(2)} = -0,925,$$

$$w_{Chest\ Pain\ Type(3)} = -5,445, w_{Resting\ Blood\ Pressure} = -12,496,$$

$$w_{Cholesterol} = 11,264, w_{Fasting\ Blood\ Sugar1} = -4,639, w_{Resting\ ECG1} = -3,769,$$

$$w_{Resting\ ECG2} = -1,456, w_{MaxHR} = 19,413, w_{Exercise\ Angina1} = -9,583,$$

$$w_{Oldpeak} = -25,042, w_{ST\ Slope1} = -13,259, w_{ST\ Slope2} = -1,762, b = 0,0734.$$

Dimana:

*Gender*: 2 = Pria ; 1 = Wanita, *Chest Pain Type* : 1 = TA ; 2 = ATA ; 3 = NAP,

*Resting ECG* : 1 = Normal ; 2 = ST. *Exercise Angina* : 1 = Tidak,

*ST Slope* : 1 = Up ; 2 = Flat ; 3 = Down.

Dengan demikian, dengan menggunakan parameter  $w$  dan  $b$  tersebut dapat mengklasifikasikan seseorang apakah termasuk penderita penyakit jantung atau tidak termasuk penderita penyakit jantung.

## 5.2 Saran

Berdasarkan hasil dan pembahasan, serta kesimpulan terdapat beberapa saran dari penelitian ini, yakni:

1. Dapat dilakukan menerapkan pada dataset pada bidang lain, seperti dataset pada analisis data teks.
2. Diharapkan dapat menggunakan data nonlinear dengan membandingkan fungsi *kernel* RBF dengan fungsi *kernel* sigmoid atau fungsi *kernel* lainnya sehingga dapat sebagai bahan referensi dan menambah wawasan lebih dalam mengenai fungsi *kernel* pada *support vector machine*.
3. Pada peneliti berikutnya, diharapkan dapat menerapkan pendekatan *machine learning* lainnya sebagai pembanding dalam menentukan metode optimasi terbaik.

## **DAFTAR PUSTAKA**

## DAFTAR PUSTAKA

- Awad, M., & Khanna, R. 2015. *Efficient learning machines: Theories, concepts, and applications for engineers and system designers*. Apress Media.
- Aykanat, M., Kılıç, Ö., Kurt B., & Saryal S. 2017. Classification of Lung Sounds Using Convolutional Neural Networks. *Eurasip Journal on Image and Video Processing*, (65), 1-9.
- Basari, A. S. H., Hussin, B., Ananta, I. G. P., & Zeniarja, J. 2013. Opinion Mining of Movie Review Using Hybrid Method of Support Vector Machine and Particle Swarm Optimization, *Procedia Engineering*, **53**, 453-462.
- Ben-Hur, A., & Weston, J. 2010. A User's Guide to Support Vector Machines. *Methods in Molecular Biology* (Clifton, N. J.), **609**, 223-239.
- Brownlee, J. 2016. *Master Machine Learning Algorithms: discover how they work and implement them from scratch*. Jason Brownlee, Melbourne, Australia.
- Burkov A. 2019. *The Hundred-page Machine Learning Book*. Andriy Burkov, Quebec.
- Cervantes, J., Garcia-Lamont, F., Rodríguez-Mazahua, L., & Lopez, A. 2020. A comprehensive survey on support vector machine classification: Applications, challenges and trends. *Artikel Neurocomputing*, 408, 1-27.
- Cristianini, N. and Shawe-Taylor, J. 2000. *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press, Cambridge.
- Das, S., & Nene, M. J. 2017. A survey on types of machine learning techniques in intrusion prevention systems. 2017. *International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET)*, 2296–2299.

- Dhage, S. N. & Raina, C. K. 2016. A review on Machine Learning Techniques. *International Journal on Recent and Innovation Trends in Computing and Communication*, **4** (3), 395-399.
- Enri, U. 2018. Optimasi Parameter Support Vector Machine Untuk Prediksi Nilai Tukar Rupiah Terhadap Dolar Amerika Serikat. *Jurnal Gerbang*, **8** (1), 12-16.
- Ernawati, I. 2019. Naïve Bayes Classifier Dan Support Vector Machine Sebagai Alternatif Solusi Untuk Text Mining. *Jurnal Teknologi Informasi dan Pendidikan*, **12** (2), 32-38.
- Goldberg, D. E., & Holland, J. H. 1988. Genetic algorithms and machine learning. *Machine Learning*, **3** (2), 95-99.
- Guenther, N. & Schonlau, M. 2016. Support Vector Machines. *The Stata Journal: Promoting Communications on Statistics and Stata*, **16** (4), 119-127.
- Gunn, S. 1998. *Support Vector Machines for Classification and Regression*. University of Southampton, Southampton.
- Hamel, L. 2009. *Knowledge Discovery With Support Vector Machines*. John Wiley & Sons, Inc., United States Of America.
- Han, J., Kamber, M., & Pei, J. 2012. *Data Mining Concepts and Techniques* 3<sup>rd</sup> Ed. Morgan Kaufmann, USA.
- Härdle, W. K., Prastyo, D. D., & Hafner, C. M. 2014. Support Vector Machines with Evolutionary Model Selection for Default Prediction. *The Oxford Handbook of Applied Nonparametric and Semiparametric Econometrics and Statistics*, 346-373.
- Herwijayanti, B., Ratnawati, D. E., & Muflikhah, L. 2018. Klasifikasi Berita Online dengan menggunakan Pembobotan TF-IDF dan Cosine Similarity. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer*, **2** (1), 306-312.
- Hsu, C. W., Chang, C. C., & Lin, C. J. 2003. A Practical Guide to Support Vector Classification,” Technical Report, Department of Computer Science and Information Engineering, University of National Taiwan, Taipei.
- Huang, G.-B., Zhu, Q.-Y., & Siew, C.-K. 2006. Extreme learning machine: theory and applications. *Neurocomputing*, **70** (1–3), 489-501.

- Indriani, A. F. & Muslim, M. A. 2019. SVM Optimization Based on PSO and AdaBoost to Increasing Accuracy of CKD Diagnosis. *Lontar Komputer : Jurnal Ilmiah Teknologi Informasi*, **10** (2), 119-127.
- Kim, E. 2014. Everything You Wanted to Know about the Kernel Trick So , What is a Kernel Anyway ? Linear SVM , Binary Classification. 1, 1-11.
- Kotsiantis, S. B. 2007. Supervised Machine Learning: A Review of Classification Techniques. *Informatica*, **31** (3), 249-268.
- Kowalczyk, A. 2017. *Support vector machines succinctly*. SynCFusion Inc., USA.
- Lu, Y., Zeng, N., Liu, X., & Yi, S. 2015. A new hybrid algorithm for bankruptcy prediction using switching particle swarm optimization and Support Vector Machines. *Discrete Dynamics in Nature and Society*, 1-7.
- Maalouf, M. & Siddiqi, M. 2014. Weighted logistic regression for large-scale imbalanced and rare events data. *Knowledge-Based Syst.*, **59**, 142-148.
- Maimon O. 2010. *Data Mining And Knowledge Discovery Handbook*. Springer, New York Dordrecht Heidelberg London.
- Meyer, D. & Wien, F. T. 2015. Support vector machines. *The Interface to libsvm in package e1071*, 28, 20.
- Mohamed, A. 2017. Comparative Study of Four Supervised Machine Learning Techniques for Classification. *International Journal of Applied Science and Technology*, **7** (2), 5-18.
- Nachev, A., & Teodosiev, T. 2015. *International Journal of Engineering and Advanced Technology* (IJEAT), **4** (4), 183-190.
- Nahar, J., Imam, T., Tickle, K. S., & Chen, Y-P. P. 2013. Computational intelligence for heart disease diagnosis: A medical knowledge driven approach. *Expert Systems with Applications* 40, 96-104.
- Nasteski, V. 2017. An overview of the supervised machine learning methods. 1-11.
- Nugraha Y. R., Wibawa, A. P., & Zaeni, I. A. E. 2019. Particle Swarm Optimization-Support Vector Machine (PSO-SVM) Algorithm for Journal Rank Classification. Proceedings - 2019 2nd International Conference of Computer and Informatics Engineering: Artificial Intelligence Roles in Industrial Revolution 4.0, IC2IE 2019, 69-73.

- Nugroho, A. S., Witarto, A. B., & Handoko, D. 2003. Support Vector Machine Teori dan Aplikasinya dalam Bioinformatika. IlmuKomputer.Com.
- Nugroho, A. S. 2007. Pengantar Support vector Machine. Chukyo University Japan.
- Prasetyo, E. 2013. *Pengolahan Citra Digital dan Aplikasinya Menggunakan Matlab*. Andi, Yogyakarta.
- Prasetyo, E., 2014. *Data Mining Mengolah Data Menjadi Informasi Menggunakan Matlab*. Andi, Yogyakarta.
- Pratiwi, N. & Setyawan, Y. 2021. Analisis Akurasi Dari Perbedaan Fungsi Kernel dan Cost Pada Support Vector Machine Studi Kasus Klasifikasi Curah Hujan di Jakarta. *Journal of Fundamental Mathematics and Applications (JFMA)*, **4** (2), 203-212.
- Rebentrost P., Mohseni, M., & Lloyd S. 2014. Quantum Support Vector Machine for Big Data Classification. *Physical Review Letters*, **113** (13), pp. 130503-1 - 130503-5.
- Ritonga, A. S. & Purwaningsih, E. S. 2018. Penerapan Metode Support Vector Machine (Svm) Dalam Klasifikasi Kualitas Pengelasan Smaw (Shield Metal Arc Welding). *Jurnal Ilmiah Edutic*, **5** (1), 17-25.
- Sandhya, Y. 2020. Prediction of Heart Diseases using Support Vector Machine. *International Journal for Research in Applied Science & Engineering Technology (IJRASET)*, **8** (11), 126-135.
- Santosa, B. 2007. *Data Mining: Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Graha Ilmu, Yogyakarta.
- Scholkopf, B. & Smola, A. 2002. *Learning with Kernels –Support Vector Machines, Regularization, Optimization and Beyond*. The MIT Press, Cambridge.
- Suwardika, G. 2017. Pengklasifikasian Pada Data Echocardiogram Dengan Menggunakan Support Vector Machine dan Analisis Diskriminan. *International Journal of Natural Science and Engineering*, **1** (1), 1-7.
- Somvanshi, M. & Chavan, P. 2016. A review of machine learning techniques using decision tree and support vector machine. 2016. *International*

*Conference on Computing Communication Control and Automation (ICCUBEA)*, 1–7.

Vapnik, V. 1995. *The Nature of Statistical Learning Theory*. Springer-Verlag, New York.

Vapnik, V., Golowich, S., & Smola, A. 1997. Support vector method for function approximation, regression estimation, and signal processing. *Proceedings of the 1996 Neural Information Processing Systems Conference (NIPS 1996)*, MIT Press, Cambridge.

Vapnik V. N. 1999. *The Nature of Statistical Learning Theory*, 2<sup>nd</sup> ed. Springer-Verlag, New York.

Widayani, W. & Harliana. 2021. Kernel Support Vector Machine Dalam Melakukan Klasifikasi Penundaan Biaya Kuliah Mahasiswa. *Jurnal Sains dan Informatika*, **7** (1), 20-27.