

## Klasifikasi Gangguan Jiwa Skizofrenia Menggunakan Algoritme *Support Vector Machine* (SVM)

Daisy Kurniawaty<sup>1</sup>, Imam Cholissodin<sup>2</sup>, Putra Pandu Adikara<sup>3</sup>

Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Brawijaya  
Email: <sup>1</sup>daisyniaa@gmail.com, <sup>2</sup>imamcs@ub.ac.id, <sup>3</sup>adikara.putra@ub.ac.id

### Abstrak

Gangguan jiwa termasuk penyakit yang banyak dijumpai di semua lapisan masyarakat. Salah satu gangguan jiwa adalah skizofrenia yang mana proses diagnosis gangguan jiwa skizofrenia cukup sulit, karena tidak ada ciri khusus perilaku maupun penampilan bagi penderitanya, beberapa penderitanya dapat berperilaku dan berperenampilan layaknya orang normal selain itu pemeriksaan dan pengobatan yang cukup mahal. Hal tersebut dapat mengakibatkan kondisi pasien semakin memburuk. Untuk mengatasi masalah tersebut dibuat suatu sistem klasifikasi gangguan jiwa skizofrenia menggunakan algoritme *support vector machine* (SVM). Dalam penelitian ini menggunakan 75 data yang terbagi menjadi dua jenis gangguan jiwa skizofrenia yaitu *paranoid* dan *simplex*. Metode yang digunakan adalah algoritme SVM yang mana termasuk dalam kategori metode klasifikasi yang baik, yang mana memberikan suatu pendekatan statistik dalam pengenalan pola, dan merupakan metode yang bersifat linier, tetapi SVM menyediakan *kernel trick*, yang dapat menyelesaikan masalah terkait klasifikasi non-linier. Hasil pengujian akhir menghasilkan rata-rata akurasi optimal yaitu 100% dengan menggunakan rasio perbandingan 90%:10%, nilai  $\gamma = 0,00001$ ,  $\lambda = 3$ , nilai  $C = 0,01$ , *kernel polynomial of degree*, dan iterasi maksimal adalah 1000.

**Kata kunci:** klasifikasi, gangguan jiwa, skizofrenia, *support vector machine*.

### Abstract

*Insanity is the most common disease. One of insanity is schizophrenia. The process of diagnosis of schizophrenia is difficult, because there is no specific characteristic of behavior or appearance for the sufferer, some sufferer can behave and look like normal people and expensive treatment. It will make the patient's condition worse. To resolve this issue, this can be done with schizophrenia classification using support vector machine (SVM) algorithm. In this study there are 75 data that is divided into two types of schizophrenia, that is paranoid and simplex. The method in this study using support vector machine algorithm, which is to the category of good classification method, provides a statistical approach in pattern recognition, and is a linear method, but SVM provides kernel trick, which can solve problems related to non-linear classification. The result obtained using SVM 100% accuracy using ratio data 90%:10%,  $\gamma = 0,00001$ ,  $\lambda = 3$ ,  $C = 0,01$ , kernel polynomial of degree, maximum iteration is 1000.*

**Keywords:** classification, insanity, schizophrenia, *support vector machine*.

### 1. PENDAHULUAN

Kesehatan merupakan suatu hal yang penting dalam kehidupan, baik kesehatan fisik maupun kesehatan mental. Kesehatan fisik dan mental harus berjalan secara beriringan, jika salah satunya adalah mengalami gangguan tentu akan membuat hidup seseorang tidak stabil. Gangguan jiwa termasuk penyakit yang banyak dijumpai di semua lapisan masyarakat.

Berdasarkan hasil rekam medis tahun 2016 di RSUD Dr. Soeroto Ngawi, tercatat bahwa jumlah pasien penderita gangguan jiwa sebanyak 511 pasien. Salah satu di antara penyakit jiwa adalah skizofrenia, merupakan gangguan jiwa yang menunjukkan gangguan pada fungsi kognitif, yakni mengenai pembentukan arus dan juga isi pikiran. Angka kematian penderita gangguan jiwa skizofrenia 8 kali lebih tinggi dari angka kematian penduduk secara umum (Hawari, 2014).

Proses diagnosis gangguan jiwa skizofrenia ini cukup sulit, karena tidak ada ciri khusus penampilan atau perilaku bagi penderita skizofrenia, beberapa di antaranya dapat berperampilan dan berperilaku layaknya orang normal (Maramis, 2009). Proses diagnosis gangguan jiwa skizofrenia, pihak medis akan melakukan pemeriksaan dan mengumpulkan data yang berupa gejala yang mengindikasikan pasien yang menderita gangguan jiwa skizofrenia, akan tetapi dalam hal ini terkadang dokter memiliki keterbatasan waktu atau jam kerja, sehingga tidak selalu berada di rumah sakit. Hal tersebut akan membuat pasien menunggu terlalu lama dan dapat mengakibatkan kondisi pasien semakin memburuk, selain itu pemeriksaan dan juga pengobatan yang cukup mahal, dan kebanyakan penyakit gangguan jiwa dialami oleh golongan masyarakat dengan ekonomi rendah. Untuk itu dibutuhkan suatu alat bantu yang mampu menentukan gangguan jiwa skizofrenia yang berupa sistem klasifikasi gangguan jiwa skizofrenia.

Penelitian serupa pernah dilakukan menggunakan metode *forward chaining* oleh (Kurniawan, 2016) yang dimulai mengamati fakta yang ada dalam mendiagnosis penyakit skizofrenia. Cara kerja dari sistem yaitu dengan cara memasukkan gejala yang didapat dari hasil wawancara sebanyak 11 gejala. Hasil keluaran dari sistem terdapat 4 jenis penyakit skizofrenia, yakni skizofrenia *paranoid*, *catatonic*, *hebepgrenic*, dan skizofrenia tak terperinci. Hasil akurasi yang didapat yaitu sebesar 87% untuk fungsi dari sistem diagnosis penyakit skizofrenia, sedangkan untuk tampilan sistem pakar tersebut menghasilkan tingkat akurasi sebesar 83%.

Apabila gangguan kejiwaan tidak segera diperhatikan, kemungkinan akan berdampak buruk bagi penderitanya. Beberapa dampak buruk yang dialami bisa saja penderita akan dikucilkan oleh masyarakat setempat, gangguan jiwa yang diderita akan lebih parah, sehingga semakin sulit untuk pengobatan. Dampak yang paling berbahaya apabila gangguan jiwa cukup berat, dapat mengakibatkan suatu keinginan untuk melakukan menyakiti diri sendiri maupun keluarganya. Dengan adanya sistem klasifikasi, dapat memudahkan untuk mengklasifikasi jenis gangguan jiwa skizofrenia, sehingga tidak membutuhkan biaya maupun waktu yang cukup lama. Sistem yang akan dibuat dalam penelitian ini akan menyimpan gejala pada pasien serta

akan menentukan jenis penyakit jiwa skizofrenia.

Berdasarkan pemaparan beberapa penelitian di atas, penulis melihat suatu peluang untuk membuat suatu sistem klasifikasi untuk mengklasifikasikan gangguan jiwa skizofrenia dengan memasukkan gejala-gejala dengan menggunakan algoritme *Support Vector Machine* (SVM). SVM termasuk dalam kategori metode klasifikasi yang baik, yang mana memberikan suatu pendekatan statistik dalam pengenalan pola, dan merupakan metode yang bersifat linear, tetapi SVM menyediakan *kernel trick*, yang dapat menyelesaikan masalah terkait klasifikasi non-linear. SVM merupakan suatu metode baru yang memiliki tingkat performansi baik dibandingkan dengan metode klasifikasi lainnya (Burges, 1998). Data yang digunakan dalam penelitian ini adalah data pasien penderita gangguan jiwa skizofrenia yang berasal dari Rekam Medik RSUD Dr. Soeroto Ngawi pada tahun 2016. Hasil keluaran dari sistem ini yaitu berupa jenis gangguan jiwa berdasarkan gejala yang dimasukkan oleh pengguna.

## 2. GANGGUAN JIWA

Berdasarkan Pedoman Penggolongan dan Diagnosis Gangguan Jiwa di Indonesia III (PPDGJ III), gangguan jiwa adalah suatu penyakit yang banyak dijumpai di masyarakat, dapat dialami oleh siapapun dan di manapun. Beberapa macam gangguan jiwa adalah gangguan *skizotipal*, waham, *disosiatif*, skizofrenia dan lain sebagainya (Depkes, 1993).

Menurut Nasir & Muhith (2011), gangguan jiwa adalah manifestasi dari menyimpangnya tingkah laku yang diakibatkan distorsi emosi, sehingga terjadi ketidakwajaran dalam berperilaku. Seseorang mengalami gangguan jiwa apabila terdapat fungsi mental yang terganggu, yakni meliputi: perasaan, pola pikir, tingkah laku, emosi, motivasi, keinginan, tilik diri dan persepsi. Hal ini dapat ditandai dengan turunnya kondisi fisik yang disebabkan oleh pencapaian keinginan yang gagal, serta dapat mengakibatkan menurunnya fungsi kejiwaan.

### 2.1 Skizofrenia

Skizofrenia merupakan gangguan jiwa yang menunjukkan gangguan pada fungsi kognitif, yakni mengenai pembentukan arus dan juga isi pikiran. Selain itu ditemukan juga gejala pada persepsi, wawasan diri, emosi serta keinginan. Kebanyakan gangguan jiwa skizofrenia terjadi

pada usia 15-35 tahun (Nasir & Muhith, 2011). Faktor-faktor yang dapat memicu gangguan jiwa skizofrenia antara lain (Hawari, 2014):

- a. Faktor keturunan.
- b. Adanya virus.
- c. *Auto-antibody*.
- d. Kekurangan gizi.

### 2.2.1 Gejala Skizofrenia

Gejala psikotik ditandai dengan adanya abnormalitas dalam bentuk serta isi dari pikiran, pemahaman, perasaan, dan tingkah laku. Berikut beberapa gejala yang dapat diamati oleh pasien penderita skizofrenia (Maramis, 2009):

1. Penampilan dan Perilaku Umum. .
2. Gangguan Pembicaraan.
3. Gangguan Afek.
4. Gangguan Persepsi.
5. Gangguan Pikiran.

## 2.2 Klasifikasi

Klasifikasi merupakan suatu kegiatan atau proses dalam mengelompokkan data ke dalam kelas tertentu berdasarkan ciri-ciri atau sifat serta pola yang terdapat dalam kelas tersebut. Proses dalam klasifikasi dilakukan dengan cara memberikan suatu data sebagai acuan dalam membuat aturan untuk pengklasifikasian data. Menurut Kuswani (2007), menerangkan bahwa klasifikasi merupakan suatu teknik melihat kelakuan serta atribut dari data yang sudah dikelompokkan, yang bertujuan untuk memberikan klasifikasi pada sebuah data baru dengan cara memanipulasi data yang sudah diklasifikasikan, dan hasilnya akan digunakan sebagai aturan. Proses dari klasifikasi dibagi menjadi 2 fase, menurut yakni *learning* dan *testing* (Pramudiono, 2003). Terdapat beberapa metode yang dapat digunakan untuk pengklasifikasian antara lain, *Naive Bayes*, *Decision Tree*, *Neural Network*, *Vupport Vector Machine* (SVM), dan *K-Nearest Neighbor*.

### 2.3 Support Vector Machine

*Support Vector Machine* (SVM) adalah suatu pembelajaran menggunakan hipotesis yakni fungsi linier pada ruang fitur yang memiliki dimensi tinggi, dan dilatih menggunakan algoritme yang didasarkan pada optimasi.

#### 2.3.1 Sequential Training

Terdapat beberapa jenis proses *training* pada *Support Vector Machine* (SVM), yakni

*Quadratic Programming*, *Sequential Minimal Optimization*, dan *Sequential Training*. Penggunaan *Quadratic Programming* memerlukan waktu yang cukup lama, selain itu algoritmenya cukup kompleks. *Sequential Minimal Optimization* merupakan pengembangan dari *Quadratic Programming*, dan memilih untuk penyelesaian optimasi yang kecil, serta algoritme juga cukup kompleks. Sedangkan pada *Sequential Training* algoritme yang dimiliki lebih sederhana, serta tidak membutuhkan waktu yang begitu lama. Berikut adalah Algoritme *Sequential Training* (Vijayakumar & , 1999).

1. Melakukan inisialisasi terhadap  $\sigma_1$  lalu parameter lainnya, seperti  $\lambda, \gamma, C, \varepsilon$ .
2. Menghitung matriks *Hessian* dengan persamaan

$$D_{ij} = y_i y_j (K(x_i, x_j) + \lambda^2) \quad (1)$$

3. Menghitung nilai *error* menggunakan persamaan berikut

$$a. E_i = \sum_{j=1}^n \alpha_j D_{ij} \quad (2)$$

$$b. \delta \alpha_i = \min(\max[\gamma(1 - E_i), a_i], C - a_i) \quad (3)$$

$$c. a_i = a_i + \delta \alpha_i \quad (4)$$

4. Langkah ketiga dilakukan secara berulang-ulang sampai mendapatkan iterasi maksimum atau  $(\delta \alpha_i \leq \varepsilon)$ .
5. Kemudian akan didapatkan nilai support vector *SV*,  $SV = (a_i > thresholdSV)$  misal  $thresholdSV = 0$ .

#### 2.3.2 Testing SVM

1. Langkah pertama menghitung nilai  $f(x)$  dengan rumus sebagai berikut:

$$f(x) = \sum_{i=1}^m a_i y_i K(x_i, x) + b \quad (5)$$

Untuk mencari nilai  $b$  (bias) dapat menggunakan rumus sebagai berikut

$$b = -\frac{1}{2} \left[ \sum_{i=1}^m a_i y_i K(x_i, x^+) + \sum_{i=1}^m a_i y_i K(x_i, x^-) \right] \quad (6)$$

2. Langkah selanjutnya menghitung nilai  $K(x_i, x_{testj})$

$$\sum_{i=1}^m y_i y_j K(x_i, x) \quad (7)$$

### 3. Menghitung nilai $f(x)_{test}$

$$f(x_{test}) = \sum_{i=1}^m a_i y_j K(x_i, x) + b \quad (8)$$

Keterangan:

$a_i$  = alfa ke- $i$

$y_i$  = kelas data ke- $i$

$m$  = jumlah data yang merupakan SV

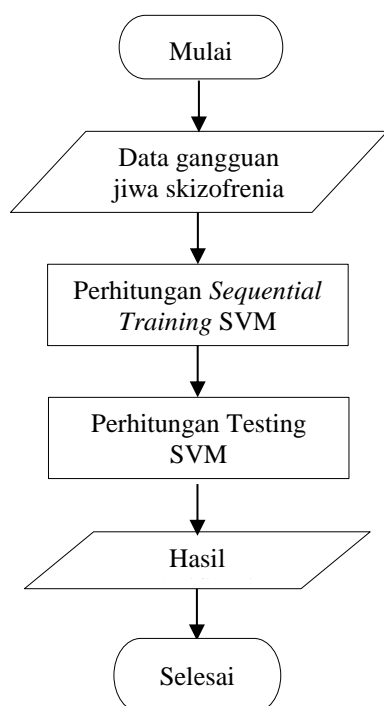
$K(x_i, x)$  = fungsi *kernel*

$b$  = nilai *bias*

### 3. PERANCANGAN IMPLEMENTASI

### DAN

Langkah awal yang dilakukan dalam proses SVM adalah mengambil *dataset*, kemudian tahap selanjutnya perhitungan dengan *kernel* yang digunakan yaitu *kernel polynomial* yang akan menghasilkan matriks *Hessian*. Tahapan proses SVM dapat dilihat pada Gambar 1.



**Gambar 1** Alur Proses Algoritme SVM

Berdasarkan Gambar yang pertama adalah masukkan *dataset* dengan format .xls yang mana data dari penelitian ini berupa data pasien penderita gangguan jiwa skizofrenia yang berjumlah 75 data dan terbagi menjadi dua jenis

gangguan jiwa skizofrenia yaitu *paranoid* dan *simplex*. Data tersebut berasal dari rekam medis RSUD Ngawi dan untuk pemilihan data latih dan data uji dilakukan secara acak. Tahap kedua yaitu melakukan perhitungan *sequential training* SVM yang terdiri dari matriks *Hessian* yang mana perhitungan matriks *Hessian* akan digunakan dalam perhitungan selanjutnya, lalu menghitung nilai  $E_i$ , yang juga akan digunakan dalam proses perhitungan berikutnya yaitu menghitung nilai  $\delta\alpha_i$  dan nilai  $\alpha_i$ . Setelah menghitung *sequential training* langkah selanjutnya yaitu melakukan perhitungan terhadap *testing* SVM, langkah pertama dalam proses *testing* adalah menghitung nilai *bias* dengan Persamaan 13, kemudian menghitung nilai  $K(x_i, x_{testj})$  yaitu dengan melihat nilai terbesar pada kelas positif dan negatif dengan menggunakan fungsi *kernel* yang digunakan. Setelah nilai  $K(x_i, x_{testj})$  didapatkan langkah selanjutnya adalah menghitung nilai  $f(x)_{test}$ , kemudian jika sudah didapat nilai  $f(x)_{test}$  maka sudah dapat diklasifikasikan berdasarkan hasil dari nilai  $f(x)_{test}$  tersebut. Jika nilai  $> 0$  maka data tersebut masuk pada kategori positif yaitu *paranoid* sedangkan jika bernilai  $< 0$  maka data tersebut masuk dalam kategori negatif yaitu *simplex*.

### 4. PENGUJIAN DAN ANALISIS

Pada penelitian ini dilakukan pengujian serta analisis terhadap sistem klasifikasi gangguan jiwa menggunakan algoritme *Support Vector Machine* (SVM). Pengujian yang dilakukan dalam penelitian ini yaitu pengujian terhadap rasio perbandingan, fungsi *kernel*, pengujian terhadap nilai parameter  $\lambda$  (*lamda*), pengujian terhadap nilai parameter  $C$  (*Complexity*), pengujian terhadap nilai parameter  $\gamma$  (*gamma*), serta pengujian terhadap jumlah iterasi maksimal pada algoritme SVM.

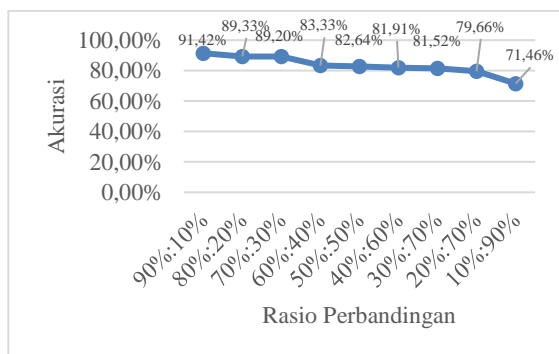
#### 4.1 Pengujian Rasio Perbandingan

Pengujian ini berdasarkan pada data latih dan data uji yang digunakan agar dapat mengetahui pada berapa rasio perbandingan data yang mendapatkan hasil akurasi tertinggi. *Dataset* dalam penelitian ini sebanyak 75 data yang terbagi menjadi dua yaitu data latih dan data uji. Rasio perbandingan yang digunakan antara lain 90%:10%, 80%:20%, 70%:30%, 60%:40%, 50%:50%, 40%:60%, 30%:70%,



20%:80%, serta 10%:90%. Nilai parameter yang digunakan dalam pengujian ini menggunakan rasio perbandingan 90%:10%, nilai  $\lambda = 0,1$ , nilai  $\gamma = 0,00001$ , nilai  $C = 0,01$ , dan jumlah iterasi maksimal sebanyak 30 kali. Percobaan uji coba yang dilakukan sebanyak 5 kali.

Analisis terhadap pengujian rasio perbandingan adalah nilai akurasi tertinggi pada rasio perbandingan yaitu sebesar 91,42% pada rasio perbandingan 90%:10%, hal ini menunjukkan bahwa sebaran data latih dan data uji sangat baik, sedangkan akurasi terkecil yaitu sebesar 71,46% yang diperoleh dari rasio perbandingan 10%:90%. Data latih yang banyak akan menghasilkan nilai akurasi yang baik pula, dan sebaliknya jika data latih sedikit, akurasi yang didapat akurasi yang kurang optimal. Pada pengujian rasio perbandingan 80%:20% sampai 10%:90% cenderung menurun, hal ini disebabkan oleh jumlah data latih yang berkurang dan mengakibatkan sulitnya dalam mengenali ciri pada jenis datanya. Jika semakin banyak data latih, maka akan menghasilkan nilai akurasi yang tinggi, hal ini dikarenakan semakin banyak data latih semakin banyak pula data yang masuk untuk menjadikan pertimbangan dalam proses klasifikasi. Hasil pengujian terhadap rasio perbandingan dapat dilihat pada Gambar 2.

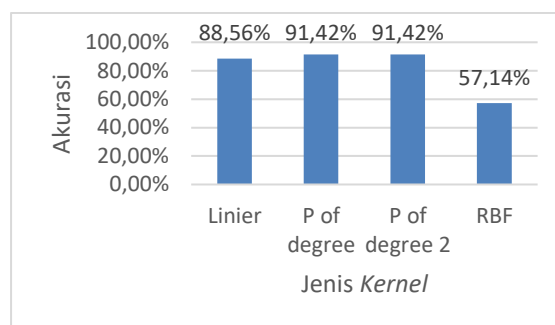


Gambar 2 Hasil pengujian rasio perbandingan

#### 4.2 Pengujian Fungsi Kernel

Fungsi *kernel* yang digunakan dalam penelitian ini adalah *kernel linier*, *polynomial of degree*, *polynomial of degree 2*, dan *kernel RBF*. Nilai parameter yang digunakan dalam pengujian ini menggunakan rasio perbandingan 90%:10% nilai  $\lambda = 0,1$ ,  $\gamma = 0,00001$ ,  $C = 0,01$ , dan iterasi sebanyak 30 kali. Percobaan uji coba yang dilakukan sebanyak 5 kali. Hasil pengujian dapat dilihat pada Gambar 3.

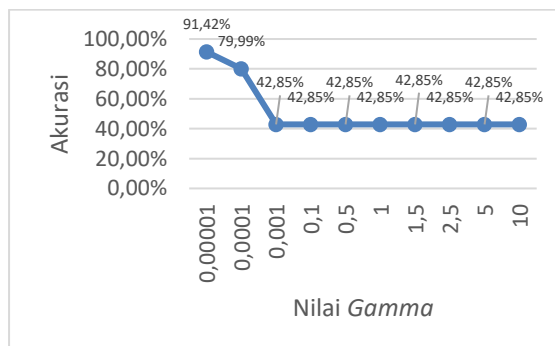
Hasil dari pengujian fungsi *kernel* terhadap empat fungsi *kernel* tersebut telah terlihat bahwa *kernel linier* menghasilkan nilai akurasi 88,56%, *kernel polynomial of degree* dan *kernel polynomial of degree 2* menghasilkan akurasi sebesar 91,42%, serta *kernel RBF* menghasilkan akurasi sebesar 57,14%. Hasil akurasi yang terbaik dalam penelitian ini yaitu menggunakan fungsi *kernel polynomial of degree* yaitu 91,42%, maka *kernel polynomial* cocok digunakan untuk data pada penelitian ini, yang mana *kernel polynomial* termasuk dalam *kernel non-stationary*, yakni *kernel* tersebut baik apabila memiliki sebaran data dan pola yang tidak tetap (Souza, 2010). Untuk melihat pola yang tidak tetap dapat dilihat dari variasi kombinasi 0 dan 1 dari semua fitur yang ada



Gambar 3 Hasil pengujian kernel

#### 4.3 Pengujian Nilai Gamma

Pengujian terhadap nilai parameter  $\gamma$  dilakukan untuk melihat skenario mana yang menghasilkan nilai terbaik. Pengujian yang dilakukan pada nilai konstanta  $\gamma = 0,00001, 0,0001, 0,001, 0,1, 0,5, 1, 1,5, 2,5, 5$ , dan 10. Nilai parameter yang digunakan dalam pengujian ini menggunakan rasio perbandingan 90%:10%, fungsi *kernel* yang digunakan *kernel polynomial of degree*, nilai  $\lambda = 0,1$ ,  $C = 0,01$ , dan iterasi sebanyak 30 kali. Hasil pengujian dapat dilihat pada Gambar 4.

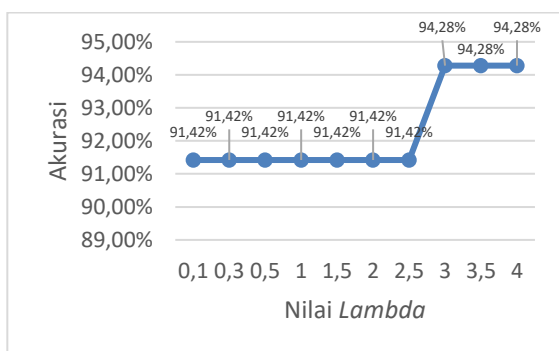


Gambar 4 Hasil pengujian gamma

Analisis yang dilakukan terhadap pengujian nilai parameter  $\gamma$  yaitu untuk mengetahui tingkat akurasi terhadap pengaruh nilai  $\gamma$ . Pengujian ini dengan nilai 0,00001, 0,0001, 0,001, 0,1, 0,5, 1, 1,5, 2,5, 5, dan 10 mendapatkan hasil akurasi tertinggi sebesar 91,42% yang diperoleh dari pengujian nilai  $\gamma = 0,00001$ . Nilai  $\gamma$  ini berpengaruh terhadap hasil akurasi, hal ini diketahui pada pengujian nilai  $\gamma$  0,00001 sampai 0,001 akurasi yang didapat cenderung menurun. Jadi dapat disimpulkan bahwa nilai  $\gamma$  semakin tinggi maka nilai *learning rate* atau laju pembelajaran akan semakin cepat, maka ketelitian pada sistem akan semakin berkurang, namun sebaliknya jika nilai  $\gamma$  semakin kecil, nilai ketelitian pada sistem akan semakin besar, sehingga akurasi yang didapatkan semakin tinggi. Tingkat akurasi tertinggi yaitu 91,42% pada nilai  $\gamma = 0,00001$  yang nantinya akan digunakan untuk pengujian terhadap parameter  $\lambda$ .

#### 4.4 Pengujian Nilai $\lambda$

Pengujian nilai parameter nilai  $\lambda$  dilakukan agar mengetahui nilai berapakah yang menghasilkan akurasi tertinggi pada algoritme *support vector machine* dalam penelitian ini. Nilai parameter yang digunakan dalam pengujian ini adalah rasio perbandingan 90%:10%, nilai  $\gamma = 0,00001$ ,  $C = 0,01$ , dan iterasi maksimal sebesar 30 kali, serta menggunakan fungsi *kernel polynomial of degree*. Hasil pengujian dapat dilihat pada Gambar 5.



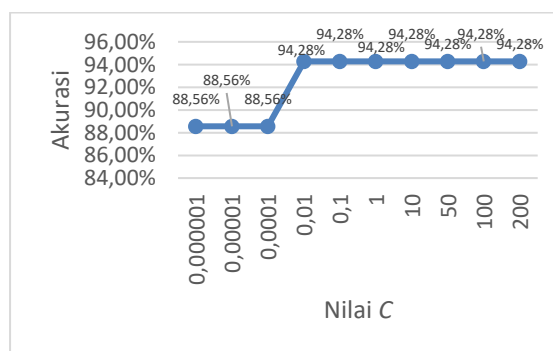
Gambar 5 Hasil pengujian  $\lambda$

Analisis yang dilakukan terhadap pengujian nilai parameter  $\lambda$  yaitu untuk mengetahui tingkat akurasi terhadap pengaruh nilai  $\lambda$ . Pengujian ini dengan nilai 0,1, 0,3, 0,5, 1, 1,5, 2, 2,5, 3, 3,5, dan 4 mendapatkan hasil akurasi tertinggi sebesar 91,42% yang diperoleh dari

pengujian nilai  $\lambda = 3, 3,5$ , dan 4. Nilai  $\lambda$  ini berpengaruh terhadap hasil akurasi, hal ini diketahui pada pengujian nilai  $\lambda$  0,1 sampai dengan 2,5 menghasilkan tingkat akurasi yang stabil yaitu 88,56%, kemudian setelah melewati nilai 2,5 cenderung naik sehingga pada nilai  $\lambda$  3, 3,5, dan 4 tingkat akurasi naik secara drastis yaitu sebesar 91,42%. Berdasarkan hasil pengujian nilai  $\lambda$ , semakin besar nilai  $\lambda$  akan menghasilkan hasil yang optimal, namun jika nilai  $\lambda$  terlalu besar dapat mengakibatkan waktu komputasi pada perhitungan matriks *Hessian* lebih lama, yang dikarenakan *augmented factor* dapat menjadikan sistem lambat dalam mencapai konvergensi dan tidak stabilnya proses pembelajaran (Vijayakumar & , 1999), hal ini terbukti pada saat pengujian menggunakan nilai  $\lambda = 100$  menghasilkan rata-rata nilai akurasi sebesar 42,85%. Tingkat akurasi tertinggi nilai  $\lambda$  yaitu 3 yang nantinya akan digunakan untuk pengujian terhadap parameter  $C$ .

#### 4.5 Pengujian Nilai $C$

Pengujian terhadap nilai parameter  $C$  dilakukan agar dapat mengetahui hasil pengujian yang terbaik dalam penelitian ini, nilai  $C$  yang diuji yaitu 0,000001, 0,00001, 0,0001, 0,01, 0,1, 1, 50, 100, dan 200. Nilai parameter lain yang digunakan dalam pengujian ini adalah rasio perbandingan 90%:10%, nilai  $\gamma = 0,00001$ ,  $\lambda = 3$ , iterasi maksimal sebanyak 30 kali, dan menggunakan fungsi *kernel polynomial of degree*. Hasil dari pengujian dapat dilihat pada Gambar 6.



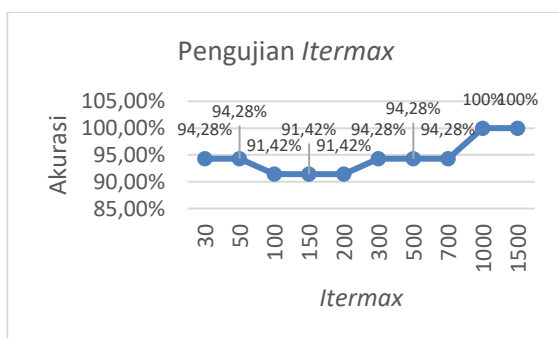
Gambar 6 Hasil pengujian nilai  $C$

Analisis yang dilakukan terhadap pengujian nilai parameter  $C$  yaitu untuk mengetahui tingkat akurasi terhadap pengaruh nilai  $C$ . Pengujian ini dengan nilai 0,000001, 0,00001, 0,0001, 0,01, 0,1, 1, 10, 50, 100 dan 200 mendapatkan hasil akurasi tertinggi sebesar 94,28%. Pada saat nilai

$C$  menjauhi nilai 0, maka margin pemisah (*hyperplane*) akan semakin lebar, hal ini dikarenakan parameter  $C$  memiliki tujuan yaitu untuk meminimalkan adanya nilai *error*. Nilai  $C$  semakin besar maka akan memberikan penalti yang besar terhadap nilai *error* pada klasifikasi (Nugroho, et al., 2003). Tingkat akurasi tertinggi yaitu nilai 0,01 yang nantinya akan digunakan untuk pengujian terhadap jumlah *itermax*.

#### 4.6 Pengujian Jumlah *Itermax*

Pengujian terhadap jumlah *itermax* dilakukan agar dapat mengetahui hasil akurasi yang terbaik untuk nilai *itermax*, yaitu 30, 50, 100, 150, 200, 300, 500, 700, 1000, dan 1500. Nilai parameter lain yang digunakan dalam pengujian ini adalah rasio perbandingan 90%:10%, nilai  $\lambda = 3$ , nilai  $\gamma = 0,00001$ , dan nilai  $C = 0,01$ . Pengujian terhadap jumlah *itermax* ini nantinya akan digunakan sebagai acuan untuk akurasi sistem secara keseluruhan, karena pengujian ini merupakan pengujian yang paling akhir. Hasil pengujian terhadap jumlah *itermax* dapat dilihat pada Gambar 7.



Gambar 7 Hasil pengujian *itermax*

Analisis yang dilakukan terhadap pengujian jumlah *itermax* yaitu untuk mengetahui tingkat akurasi yang didapat terhadap pengaruh jumlah *itermax*. Hasil pengujian ini menghasilkan nilai akurasi tertinggi yaitu 100% dengan jumlah *itermax* 1000 dan 1500. Nilai *itermax* dipengaruhi oleh beberapa nilai parameter lainnya, yaitu rasio perbandingan 90%:10%,  $\lambda = 3$ ,  $\gamma = 0,00001$ , dan nilai  $C = 0,001$ . Semakin besar jumlah *itermax* maka akurasi yang didapat akan semakin besar juga, namun pada pengujian ini terjadi ketidakstabilan tingkat akurasi yaitu pada saat *itermax* 30 sampai 50 menghasilkan akurasi sebesar 94,28%, pada *itermax* 100 sampai 200, akurasi yang didapat cenderung menurun, kemudian pada *itermax* 300 sampai 700 akurasi yang didapat cenderung

meningkat, hal ini disebabkan karena nilai  $\alpha_i$  belum mencapai nilai konvergen, yang mana dapat didefinisikan dari tingkat perubahan nilai  $\alpha_i$ . Selain itu adanya kemungkinan pemilihan data latih dan data uji secara acak.

Analisis yang dilakukan pada pengujian akhir ini yaitu menggunakan rasio perbandingan data 90%:10% yaitu 68 data latih dan 7 data uji, nilai parameter yang digunakan  $\lambda = 3$ , nilai  $\gamma = 0,00001$ , nilai  $C = 0,01$ , dan jumlah *itermax* = 1000, serta menggunakan fungsi *kernel polynomial of degree*. Pengujian dari penelitian ini menghasilkan nilai akurasi yang paling tinggi yaitu sebesar 100%.

## 5. KESIMPULAN

Sesuai dengan hasil pengujian yang sudah dilakukan terhadap sistem klasifikasi gangguan jiwa skizofrenia menggunakan algoritme *support vector machine* (SVM) dapat ditarik suatu kesimpulan, yaitu sebagai berikut.

1. Algoritme *Support Vector Machine* (SVM) dapat diimplementasikan pada klasifikasi gangguan jiwa skizofrenia dengan 2 jenis, yaitu skizofrenia *paranoid* dan skizofrenia *simplex*. Tahapan implementasi untuk penelitian ini adalah yang pertama mengumpulkan data pasien gangguan jiwa skizofrenia yang didapat dari hasil rekam medik RSUD Ngawi, langkah berikutnya pemilihan untuk data latih dan data uji secara acak sesuai dengan rasio perbandingan yang digunakan. Kemudian melakukan perhitungan *kernel polynomial*, setelah itu menghitung tahapan *sequential training* dan *testing* pada SVM. Hasil akhir berupa akurasi sistem dengan membandingkan kelas klasifikasi prediksi dan kelas klasifikasi aktual.
2. Hasil pengujian yang sudah dilakukan terhadap sistem klasifikasi gangguan jiwa skizofrenia menggunakan algoritme *support vector machine* menghasilkan akurasi terbaik berdasarkan pemilihan rasio perbandingan 90%:10%, nilai parameter  $\gamma = 0,00001$ , nilai  $\lambda = 3$ , nilai  $C = 0,01$ , dan jumlah *itermax* adalah 1000, serta menggunakan *kernel polynomial*. Berdasarkan hasil pengujian yang sudah dilakukan, algoritme SVM sangat cocok untuk klasifikasi data pada penelitian ini yang menghasilkan akurasi optimal yaitu sebesar 100%.

**DAFTAR PUSTAKA**

- Burges, C. J., 1998. *A Tutorial on Support Vector Machines for Pattern Recognition..* s.l.:Bell Laboratories.
- Depkes, 1993. *Pedoman Penggolongan dan Diagnosis Gangguan Jiwa di Indonesia III (PPDGJ III)*. Jakarta: Direktorat Jendral Pelayanan Medik.
- Hawari, D., 2014. *Skizofrenia Pendekatan Holistik (BPSS) BIO-PSIKO-SOSIAL-SPIRITUAL*. Jakarta: Fakultas Kedokteran Universitas Indonesia.
- Kurniawan, D. C., 2016. *Sistem Pakar untuk Mendiagnosa Penyakit Skizofrenia dengan Metode Forward Chaining*. Surakarta: Fakultas Ilmu Komunikasi dan Informatika Universitas Muhammadiyah Surakarta.
- Kusnawi, 2007. *Pengantar Solusi Data Mining*. Yogyakarta: STMIK AMIKOM.
- Maramis, W. F., 2009. *Catatan Ilmu Kedokteran Jiwa Edisi 2*. Surabaya: Airlangga University Press.
- Nasir, A. & Muhith, A., 2011. *Dasar-Dasar Keperawatan Jiwa Pengantar dan Teori*. Jakarta: Salemba Medika.
- Nugroho, A. S., Witarto, A. B. & Handoko, D., 2003. *Support Vector Machine Teori dan Aplikasinya dalam Bioinformatika*. s.l.:s.n.
- Pramudiono, I., 2003. *Pengantar Data Mining: Menambang Permata Pengetahuan di Gunung Data*. s.l.:Ilmu Komputer.com.
- Souza, C., 2010. *Kernel Functions for Machine Learning Applications*. [Online] Available at: <http://crsouza.com/2010/03/17/kernel-functions-for-machine-learning-applications/#polynomial> [Diakses 7 Juli 2017].
- Vijayakumar, S. & S. W., 1999. Sequential Support Vector Classifier and Regression. *Processing International Conference on Soft Computing*, pp. 610-619.