

KLASIFIKASI PENERIMA PROGRAM BERAS MISKIN (RASKIN) DI KABUPATEN WONOSOBO DENGAN METODE SUPPORT VECTOR MACHINE MENGGUNAKAN *LibSVM*

Yogi Setiyo Pamuji¹, Diah Safitri², Alan Prahutama³

¹Mahasiswa Jurusan Statistika FSM Universitas Diponegoro

^{2,3}Staff Pengajar Jurusan Statistika FSM Universitas Diponegoro

yogi_setiyo@yahoo.co.id, diahsafitri.fifi@gmail.com, alan.prahutama@gmail.com

ABSTRACT

Beras Miskin (Raskin) Program is a program of social protection, as supporters of other programs such as nutrition improvement, healthy increase, education and productivity improvement of Poor Households. According to Badan Pusat Statistika, there were 14 criteria to determine a household is classified as poor households. Based on these criteria it will be classified of recipient households and non-recipient households of Beras Miskin (Raskin) Program by Support Vector Machine (SVM) method using *LibSVM*. The concept of classification by SVM is search for the best hyperplane which serves as a separator of two classes of data in the input space. Kernel function is used to convert the data into a higher dimensional space to allow a separation. *LibSVM* is a package program created by Chih-Chung Chang and Chih-Jen Lin from Department of Computer Science at National Taiwan University. The method used by *LibSVM* to obtain global solution of duality lagrange problem is decomposition method. To determine the best parameters of kernel function, used k-fold cross validation method and grid search algorithm. In this classification by SVM method using *LibSVM*, obtain the best accuracy value as 83,1933%, which is the kernel function Radial Basis Function (RBF).

Keywords : Beras Miskin (Raskin) Program, Classification, Support Vector Machine (SVM), *LibSVM*, Kernel Function

1. PENDAHULUAN

Program Subsidi Beras bagi Masyarakat Berpendapatan Rendah (Program Raskin) adalah Program Nasional lintas sektoral baik horizontal maupun vertikal, yang bertujuan untuk membantu mencukupi kebutuhan beras masyarakat berpendapatan rendah^[7]. Sasaran program ini tidak hanya rumah tangga miskin, tetapi meliputi rumah tangga rentan atau hampir miskin^[8]. Dalam penetapan rumah tangga miskin terdapat 14 kriteria yang harus terpenuhi^[2]. Mengacu pada kriteria rumah tangga penerima Program Raskin dan bukan penerima Program Raskin yang ada maka penulis bermaksud melakukan klasifikasi dengan menerapkan ilmu statistika menggunakan teknik *data mining*.

Terdapat beberapa metode yang sering digunakan dalam *data mining* khususnya untuk fungsi klasifikasi, metode yang digunakan antara lain *Hierarchical Clustering*, *K-means*, *Fuzzy K-means*, *K-Nearest Neighbor*, Analisis Diskriminan Linier (LDA), *Klasifier Naive Bayes*, *Decision Tree*, *Artificial Neural Network (ANN)* dan *Support Vector Machine (SVM)*^[12]. *Support Vector Machine (SVM)* merupakan bagian dari metode pembelajaran yang digunakan untuk klasifikasi. SVM memetakan vektor input ke sebuah ruang dimensi yang lebih tinggi dimana *hyperplane* pemisah dibangun^[1]. Untuk penerapan metode *Support Vector Machine (SVM)* digunakan fungsi-fungsi dalam *LibSVM*^[6]. *LibSVM* merupakan *package program* yang digunakan untuk mendapatkan solusi optimal dari *dualitas lagrange* pada SVM^[3].

Berdasarkan permasalahan klasifikasi rumah tangga penerima dan rumah tangga bukan penerima Program Beras Miskin (Raskin), maka penulis tertarik untuk mengaplikasikan metode SVM menggunakan *LibSVM* pada data klasifikasi rumah tangga

penerima dan rumah tangga bukan penerima Program Beras Miskin (Raskin) dengan tujuan mendapatkan nilai akurasi klasifikasi hasil prediksi.

2. TINJAUAN PUSTAKA

2.1 Klasifikasi Rumah Tangga Penerima Program Beras Miskin

Dalam rangka pemenuhan hak dan kebutuhan pangan bagi Rumah Tangga Miskin (RTM), Pemerintah melaksanakan Program Beras Miskin (Raskin) sebagai salah satu program perlindungan sosial, yang bertujuan untuk memenuhi sebagian kebutuhan pangan (beras) sehingga diharapkan dapat mengurangi beban pengeluaran Rumah Tangga Miskin (RTM). Program Beras Miskin (Raskin) merupakan program perlindungan sosial, sebagai pendukung program lainnya seperti perbaikan gizi, peningkatan kesehatan, pendidikan dan peningkatan produktivitas Rumah Tangga Miskin (RTM). Untuk itu, Program Beras Miskin (Raskin) harus direncanakan secara terkoordinasi dengan instansi dan Satuan Kerja Perangkat Daerah (SKPD) serta stakeholders terkait lainnya dengan mempertimbangkan kondisi obyektif masing-masing daerah^[14].

Untuk mengimplementasikan berbagai program penanggulangan kemiskinan, informasi mengenai siapa yang miskin dan dimana mereka berada menjadi sangat penting dan akan menjadi modal dasar dalam *targeting* Rumah Tangga Miskin (RTM). Data kemiskinan yang bersumber dari BPS sering menjadi dasar dalam implementasi program penanggulangan kemiskinan oleh Pemerintah^[2].

2.2 Support Vector Machine (SVM)

2.2.1 Konsep Support Vector Machine (SVM)

Konsep klasifikasi dengan SVM dapat dijelaskan secara sederhana sebagai usaha mencari *hyperplane* terbaik yang berfungsi sebagai pemisah dua buah kelas data pada *input space*. *Hyperplane* (batas keputusan) pemisah terbaik antara kedua kelas dapat ditemukan dengan mengukur margin *hyperplane* tersebut dan mencari titik maksimalnya. Margin adalah jarak antara *hyperplane* tersebut dengan data terdekat dari masing-masing kelas^[11].

2.2.2 SVM pada Linearly Separable Data

Data latih dinyatakan oleh (x_i, y_i) dengan $i = 1, 2, \dots, l$, dan $x_i = \{x_{i1}, x_{i2}, \dots, x_{iq}\}^T$ merupakan atribut (fitur) set untuk data latih ke- i , kemudian untuk $y_i \in \{-1, +1\}$ menyatakan label kelas^[11]. *Hyperplane* klasifikasi linier SVM dinotasikan sebagai berikut^[11]:

$$w \cdot x_i + b = 0 \quad (1)$$

Jarak antara dua *hyperplane* pendukung adalah margin m , margin dapat dihitung sebagai proyeksi dari perbedaan antara dua *support vector* dalam direksi vektor normal, sedangkan persamaannya dapat ditulis sebagai berikut^[5]:

$$\begin{aligned} m &= \frac{\|x_1 - x_2\| \cos \gamma}{2} \\ &= \frac{2}{\|w\|} \end{aligned} \quad (2)$$

Persamaan (2) diformulasikan ke dalam problem *Quadratic Programming (QP)* dengan meminimalkan invers persamaan (2), $\frac{1}{2} \|w\|^2$, dibawah konstrain (syarat) seperti berikut^[11]:

$$\min \frac{1}{2} \|w\|^2 \quad (3)$$

syarat : $y_i (w \cdot x_i + b) \geq 1, i=1, 2, \dots, n$

n adalah banyaknya data yang menjadi *support vector*.

Metode *lagrange multiplier* digunakan untuk memecahkan masalah optimasi^[13]. Jadi, permasalahannya menjadi^[13] :

$$L_p(\mathbf{w}, b, \alpha) = \left\{ \frac{1}{2} \|\mathbf{w}\|^2 - \sum_{i=1}^n \alpha_i [y_i(\mathbf{w} \cdot \mathbf{x}_i + b) - 1] \right\} \quad (4)$$

syarat $\alpha_i \geq 0$

Dimana α_i adalah koefisien *lagrange*^[13].

Masalah optimasi (4) masih sulit diselesaikan karena banyaknya parameter (\mathbf{w}, b dan α_i), untuk menyederhanakannya, persamaan optimasi (4) harus ditransformasikan ke dalam fungsi *lagrange multiplier* itu sendiri (disebut dualitas masalah)^[11]:

$$L_d = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (5)$$

syarat $\alpha_i \geq 0$

$$\sum_i \alpha_i y_i = 0$$

2.2.3 SVM pada *Nonlinearly Separable Data*

Untuk mengklasifikasi data yang tidak dapat dipisahkan secara linier, formula SVM harus dimodifikasi karena tidak ada solusi yang ditemukan^[11]. Hal yang menyebabkan proses optimasi tidak dapat diselesaikan karena tidak ada \mathbf{w} dan b yang memenuhi pertidaksamaan $y_i(\mathbf{w} \cdot \mathbf{x}_i + b) \geq 1, i=1, 2, \dots, l$ ^[11].

Model pembelajaran SVM memperkenalkan istilah penalti untuk kesalahan klasifikasi dalam fungsi objektif dengan menggunakan parameter biaya (*cost parameter*), dengan adanya parameter biaya terhadap kesalahan, maka fungsi optimasi SVM menjadi^[9]:

$$\min \frac{1}{2} \|\mathbf{w}\|^2 + c \sum_{i=1}^n \xi_i \quad (6)$$

$\xi_i \geq 0$ merupakan variabel *slack* untuk memungkinkan kesalahan beberapa klasifikasi dan C yang disebut sebagai parameter biaya untuk mengontrol keseimbangan antara margin dan kesalahan klasifikasi^[9].

Dasar dari optimasi permasalahan untuk SVM dengan variabel *slack* akan sama dengan optimasi permasalahan untuk SVM tanpa variabel *slack*, hanya saja syaratnya berubah^[5]. Jadi, persamaan *dualitas lagrange multiplier* yang dimiliki menjadi^[5]:

$$L_d = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1, j=1}^n \alpha_i \alpha_j y_i y_j \mathbf{x}_i \cdot \mathbf{x}_j \quad (7)$$

syarat $0 \leq \alpha_i \leq C$

$$\sum_i \alpha_i y_i = 0$$

2.2.4 Metode Kernel

Fungsi kernel dapat mengubah data ke ruang dimensi yang lebih tinggi sehingga memungkinkan untuk melakukan pemisahan^[1]. Bentuk dualitasnya berubah menjadi^[5]:

$$\max L_d(\alpha) = \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i,j=1}^n \alpha_i \alpha_j y_i y_j K(\mathbf{x}_i, \mathbf{x}_j) \quad (8)$$

dengan syarat

$$\alpha_i \geq 0$$

$$\sum_{i=1}^n \alpha_i y_i = 0$$

Beberapa pilihan fungsi kernel yang banyak digunakan dalam aplikasi adalah sebagai berikut^[4]:

- Linier
 $K(\mathbf{x}_i, \mathbf{x}_j) = \mathbf{x}_i^T \mathbf{x}_j$
- Polynomial
 $K(\mathbf{x}_i, \mathbf{x}_j) = (\gamma(\mathbf{x}_i^T \mathbf{x}_j + c))^d$
- Radial Basis Function (RBF)
 $K(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$
- Tangent Hyperbolic (Sigmoid)
 $K(\mathbf{x}_i, \mathbf{x}_j) = \tanh(\gamma(\mathbf{x}_i^T \mathbf{x}_j) + c)$

2.3 Algoritma Pelatihan Support Vector Machine (SVM)

Untuk menemukan *hyperplane* digunakan suatu bobot, bobot tersebut diperoleh dengan memecahkan *Quadratic Programming* (QP) sebagai berikut^[16]:

$$\min R(\alpha) = \frac{1}{2} \alpha^T Q \alpha - e \alpha^T \quad (9)$$

Metode optimasi *hyperplane* SVM umumnya dipakai untuk menyelesaikan *Quadratic Programming* (QP) dengan kendala yang ditetapkan^[10]. Beberapa pilihan metode yang digunakan adalah *chunking*, metode dekomposisi, dan *Sequential Minimal Optimization* (SMO)^[10].

Berikut ini merupakan metode dekomposisi yang digunakan dalam *LibSVM*^[3]:

1. Ditentukan nilai awal dari proses iterasi yang akan dikerjakan.
2. Jika α^k merupakan titik stasioner dari persamaan (9) maka proses berhenti, sedangkan jika α^k bukan merupakan titik stasioner dari persamaan (9), maka dicari *working set* B dimana $B = \{i, j\}$ dan dibatasi oleh N dengan $N \equiv \{1, \dots, l\} \setminus B$.
3. Jika $\alpha_{ij} \equiv K_{ii} + K_{jj} - 2K_{ij} > 0$; $K_{ij} = K(x_i, x_j)$ maka dengan variabel $\alpha_B = [\alpha_i \alpha_j]^T$ *sub-problem* (20) perlu diselesaikan

$$\min_{\alpha_i, \alpha_j} \frac{1}{2} [\alpha_i \alpha_j] \begin{bmatrix} Q_{ii} & Q_{ij} \\ Q_{ij} & Q_{jj} \end{bmatrix} \begin{bmatrix} \alpha_i \\ \alpha_j \end{bmatrix} + (-e_B + Q_{BN} \alpha_N^k)^T \begin{bmatrix} \alpha_i \\ \alpha_j \end{bmatrix} \quad (10)$$

syarat

$$0 \leq \alpha_i, \alpha_j \leq C$$

$$y_i \alpha_i + y_j \alpha_j = -\mathbf{y}_N^T \alpha_N^k$$

Jika tidak, maka ditambahkan τ sebagai konstanta positif yang kecil dan persamaan (21) perlu diselesaikan.

$$\min_{\alpha_i, \alpha_j} \frac{1}{2} [\alpha_i \alpha_j] \begin{bmatrix} Q_{ii} & Q_{ij} \\ Q_{ij} & Q_{jj} \end{bmatrix} \begin{bmatrix} \alpha_i \\ \alpha_j \end{bmatrix} + (-e_B + Q_{BN} \alpha_N^k)^T \begin{bmatrix} \alpha_i \\ \alpha_j \end{bmatrix} + \frac{\tau - \alpha_{ij}}{4} ((\alpha_i - \alpha_i^k)^2 + (\alpha_j - \alpha_j^k)^2) \quad (11)$$

$$\begin{aligned} \text{syarat} \quad & 0 \leq \alpha_i, \alpha_j \leq C \\ & y_i \alpha_i + y_j \alpha_j = -\mathbf{y}_N^T \boldsymbol{\alpha}_N^k \end{aligned}$$

4. Ditentukan kembali nilai $\boldsymbol{\alpha}_B^k$ yang baru. Kemudian $\boldsymbol{\alpha}_B^k$ dan $\boldsymbol{\alpha}_N^k$ yang baru digabungkan sehingga terbentuk $\boldsymbol{\alpha}^k$. Setelah itu kembali ke langkah 2.

Untuk mengoptimalkan kondisi dari persamaan (9) adalah dengan mencantumkan bahwa $\boldsymbol{\alpha}$ yang dapat dikerjakan merupakan titik stasioner dari persamaan (9), yaitu jika dan hanya jika^[3]:

$$m(\boldsymbol{\alpha}) \leq M(\boldsymbol{\alpha}) \quad (12)$$

dimana

$$m(\boldsymbol{\alpha}) \equiv \max_{i \in I_{up}(\boldsymbol{\alpha})} -y_i \nabla_i f(\boldsymbol{\alpha}) \quad \text{dan} \quad M(\boldsymbol{\alpha}) \equiv \min_{i \in I_{low}(\boldsymbol{\alpha})} -y_i \nabla_i f(\boldsymbol{\alpha})$$

$$\nabla f(\boldsymbol{\alpha}) = \mathbf{Q} \boldsymbol{\alpha} - \mathbf{e}$$

$$\nabla_i f(\boldsymbol{\alpha}) + b y_i \begin{cases} > 0 & \text{jika } \alpha_i < C \\ < 0 & \text{jika } \alpha_i > 0 \end{cases}$$

dan

$$I_{up}(\boldsymbol{\alpha}) \equiv \{t \mid \alpha_t < C, y_t = 1 \text{ atau } \alpha_t > 0, y_t = -1\}$$

$$I_{low}(\boldsymbol{\alpha}) \equiv \{t \mid \alpha_t < C, y_t = -1 \text{ atau } \alpha_t > 0, y_t = 1\}$$

Untuk memilih *working set* B, digunakan prosedur sebagai berikut^[3]:

1. Untuk semua t dan s ditetapkan bahwa :

$$a_{ts} \equiv K_{tt} + K_{ss} - 2K_{ts}, \quad b_{ts} \equiv -y_t \nabla_t f(\boldsymbol{\alpha}^k) + y_s \nabla_s f(\boldsymbol{\alpha}^k) > 0 \quad (13)$$

Selanjutnya ditentukan i dan j sebagai berikut :

$$\begin{aligned} i &\in \arg \max_t \{-y_t \nabla_t f(\boldsymbol{\alpha}^k) \mid t \in I_{up}(\boldsymbol{\alpha}^k)\}, \\ j &\in \arg \min_t \left\{ -\frac{b_{it}^2}{\bar{a}_{it}} \mid t \in I_{low}(\boldsymbol{\alpha}^k), -y_t \nabla_t f(\boldsymbol{\alpha}^k) < -y_i \nabla_i f(\boldsymbol{\alpha}^k) \right\} \end{aligned} \quad (14)$$

2. Diperoleh *working set* B, yaitu $B = \{i, j\}$

Dalam *LibSVM*, ditentukan bahwa $\rho = -b$. Sedangkan perhitungannya adalah sebagai berikut^[3] :

$$\rho = \frac{\sum_{i: 0 < \alpha_i < C} y_i \nabla_i f(\boldsymbol{\alpha})}{|\{i \mid 0 < \alpha_i < C\}|}$$

2.4 Estimasi Parameter

2.4.1 K-Fold Cross Validation

Pengujian dengan metode *k-fold cross validation* adalah dengan cara membagi data *training* sebanyak k bagian, kemudian $k-1$ bagian digunakan sebagai data untuk *training* sistem dan sisanya digunakan sebagai data *test*^[6]. Pengujian ini dilakukan sebanyak k kali dengan mengganti-ganti partisi yang berfungsi sebagai data *training* dan data *testing*^[6].

2.4.2 Algoritma Grid Search

Algoritma *grid search* membagi jangkauan pencarian parameter yang akan dioptimalkan ke dalam *grid* dan melintasi semua titik *grid* untuk mendapatkan nilai optimal, jika parameter c dan g masing-masing memiliki taksiran nilai sebanyak m dan n , algoritma *grid search* akan melatih dan menguji $m \times n$ (semua kombinasi) model SVM untuk menemukan parameter yang optimal^[17].

2.5 Pengukuran Kinerja Klasifikasi

Umumnya pengukuran kinerja klasifikasi dilakukan dengan matriks konfusi (*confusion matrix*)^[10]. *Confusion matrix* memberikan penilaian performance klasifikasi

berdasarkan objek dengan benar atau salah, *confusion matrix* berisi informasi aktual (*actual*) dan prediksi (*predicted*) pada sistem klasifikasi^[15].

Nilai *Accuracy* adalah proporsi jumlah prediksi yang benar, dapat dihitung dengan menggunakan persamaan^[15]:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

2.6 Cara Penggunaan Paket Program *LibSVM*

Penggunaan paket program *LibSVM* mengikuti 2 tahapan, yaitu^[3]:

- Melakukan pelatihan terhadap set data untuk mendapatkan sebuah model
- Menggunakan model yang telah didapatkan untuk memprediksi informasi dalam sebuah set data pelatihan.

3. METODE PENELITIAN

3.1 Jenis dan Sumber Data

Dalam penelitian ini data yang digunakan adalah data sekunder yaitu Data Penerima Program Beras Miskin (Raskin) di Kabupaten Wonosobo tahun 2014 yang diperoleh dari Badan Pusat Statistik (BPS) Provinsi Jawa Tengah. Data tersebut dihimpun oleh Badan Pusat Statistik melalui Survei Sosial Ekonomi Nasional 2014. Terdapat 597 rumah tangga yang diikutsertakan dalam survei dan digolongkan ke dalam rumah tangga penerima program Beras Miskin dan rumah tangga bukan penerima program Beras Miskin.

3.2 Variabel Penelitian

1. Variabel independen (x)

Pada penelitian ini akan terdapat 14 variabel independen (x), yaitu jenis dinding tempat tinggal (x_1), jenis lantai tempat tinggal (x_2), luas lantai bangunan tempat tinggal (x_3), sumber air minum (x_4), fasilitas buang air besar (x_5), sumber penerangan rumah tangga (x_6), bahan bakar untuk memasak sehari-hari (x_7), kepemilikan aset (x_8), lapangan pekerjaan kepala rumah tangga (x_9), Konsumsi daging / susu / ayam seminggu terakhir (x_{10}), kemampuan membeli pakaian dalam 3 bulan terakhir (x_{11}), konsumsi makanan per anggota rumah tangga dalam sehari (x_{12}), kemampuan berobat ke puskesmas / poliklinik (x_{13}), pendidikan tertinggi kepala rumah tangga (x_{14}).

2. Variabel dependen (y)

Pada penelitian ini, data akan dikelompokkan ke dalam dua kelas, yaitu rumah tangga dengan label bukan penerima Program Raskin = -1 dan rumah tangga penerima Program Raskin = 1.

3.3 Langkah-langkah Analisis

Berikut merupakan tahapan analisis dalam penelitian ini :

- Mempersiapkan data klasifikasi penerima dan bukan penerima program beras miskin di Kabupaten Wonosobo yang terdiri dari variabel independen dan variabel dependen.
- Membagi data tersebut menjadi data *training* dan data *testing* dengan proporsi 80:20.

3. Melakukan klasifikasi data penerima dan bukan penerima program beras miskin di Kabupaten Wonosobo dengan metode *Support Vector Machine* menggunakan *LibSVM* melalui tahapan sebagai berikut:
 - a. Menentukan fungsi kernel, nilai-nilai parameter kernel dan parameter *cost* untuk optimasi *hyperplane* pada data *training*.
 - b. Menentukan nilai parameter terbaik untuk klasifikasi data pada penelitian ini dengan menggunakan metode *k-fold cross validation* dan algoritma *grid search* yang diterapkan pada data *training*.
 - c. Menentukan *hyperplane* dengan menggunakan parameter terbaik.
 - d. Menggunakan *hyperplane* dengan parameter terbaik yang diperoleh untuk setiap fungsi kernel pada klasifikasi data *testing*.
 - e. Evaluasi hasil klasifikasi data penerima dan bukan penerima program beras miskin di Kabupaten Wonosobo dengan metode *Support Vector Machine* menggunakan *LibSVM* pada data *testing* untuk mengukur ketepatan klasifikasi dengan nilai akurasi.

4. HASIL DAN PEMBAHASAN

Untuk memperoleh hasil klasifikasi rumah tangga penerima dan rumah tangga bukan penerima Program Beras Miskin (Raskin) di Kabupaten Wonosobo tahun 2014 maka perlu dilakukan beberapa tahapan analisis:

1. Pada penelitian ini data dibagi menjadi data *training* dan data *testing* dengan proporsi 80 : 20.
2. Fungsi kernel yang digunakan pada penelitian ini adalah fungsi kernel Linier, *Polynomial* dan *Radial Basis Function* (RBF).
3. Pada penelitian ini, penentuan parameter terbaik dilakukan dengan metode *grid search* yang dipadukan dengan metode *k-fold cross validation* ($k = 10$) sebagai pengukur kinerja metode klasifikasi SVM pada data *training* dengan bantuan paket program *LibSVM* pada program Matlab.
 - a. SVM dengan Fungsi Kernel Linier
Nilai parameter C (*cost*) yang terbaik pada *hyperplane* SVM dengan fungsi kernel linier adalah $C = 0,05$ dengan *error* terkecil sebesar 0,1590.
 - b. SVM dengan Fungsi Kernel *Polynomial*
Nilai parameter C (*cost*) dan parameter d (*degree*) yang terbaik untuk digunakan pada model SVM dengan fungsi kernel *polynomial* adalah $C=0,005$ dan $d = 2$ dengan *error* terkecil sebesar 0,1527.
 - c. SVM dengan Fungsi Kernel *Radial Basis Function* (RBF)
Nilai parameter C (*cost*) dan parameter γ (*gamma*) yang terbaik untuk digunakan pada model SVM dengan fungsi kernel *Radial Basis Function* (RBF) adalah $C = 1000$ dengan $\gamma = 10^{-3}$ dimana *error* yang didapatkan adalah sebesar 0,1569.
4. Berdasarkan nilai parameter-parameter yang telah diperoleh, maka tahapan selanjutnya adalah menentukan *hyperplane* dengan menggunakan nilai parameter-parameter tersebut.
 - a. SVM dengan Fungsi Kernel Linier

$$d(\mathbf{x}^T) = \mathbf{w} \cdot \mathbf{x}_i^T + 0,17185$$
 dengan

$$\mathbf{w} = [\begin{array}{ccccccccc} 0,0844 & 0,3013 & -0,05 & 0,1349 & 0,2189 & 0 & 0,0734 & 0,0475 & -0,0777 \\ 0,1975 & -0,1878 & -0,3156 & -0,0441 & -0,2507 & & & & \end{array}]$$

- b. SVM dengan Fungsi Kernel *Polynomial*

$$D(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + 0,3215$$

dengan

$$K(\mathbf{x}, \mathbf{x}_i) = (\gamma(\mathbf{x}^T \mathbf{x}_i + c))^d ; \text{ dimana } \gamma = 1, c = 1 \text{ dan } d = 2$$

- c. SVM dengan Fungsi Kernel *Radial Basis Function* (RBF)

$$D(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) - 35,6437$$

dengan

$$K(\mathbf{x}, \mathbf{x}_i) = \exp(-\gamma \|\mathbf{x} - \mathbf{x}_i\|^2) ; \text{ dimana } \gamma = 10^{-3}$$

5. Pada penelitian ini, nilai $D(\mathbf{x})$ yang positif (+) menandakan bahwa data masuk kelas 1, sedangkan nilai $D(\mathbf{x})$ yang negatif (-) menandakan bahwa data masuk kelas -1.
6. Langkah selanjutnya adalah menghitung akurasi klasifikasi pada data *testing* tersebut.

- a. SVM dengan Fungsi Kernel Linier

$$\text{Nilai akurasi} = \frac{15 + 81}{15 + 81 + 18 + 5} = \frac{96}{119} = 0,806723$$

- b. SVM dengan Fungsi Kernel *Polynomial*

$$\text{Nilai akurasi} = \frac{13 + 84}{13 + 84 + 20 + 2} = \frac{97}{119} = 0,815126$$

- c. SVM dengan Fungsi Kernel *Radial Basis Function* (RBF)

$$\text{Nilai akurasi} = \frac{18 + 81}{18 + 81 + 5 + 15} = \frac{99}{119} = 0,831933$$

5. KESIMPULAN

Berdasarkan analisis yang telah dilakukan, diperoleh kesimpulan sebagai berikut :

- Parameter terbaik pada masing-masing fungsi kernel untuk *hyperplane Support Vector Machine* (SVM) yang didapatkan dengan menerapkan metode *k-fold cross validation* dan algoritma *grid search* adalah $C = 0,05$ untuk fungsi kernel Linier, $C=0,005$ dan $d = 2$ untuk fungsi kernel *Polynomial*, serta $C = 1000$ dan $\gamma = 10^{-3}$ untuk fungsi kernel *Radial Basis Function* (RBF).
- Hyperplane* terbaik yang didapatkan untuk klasifikasi data dengan metode *Support Vector Machine* (SVM) menggunakan *LibSVM* adalah sebagai berikut :

- a. SVM dengan Fungsi Kernel Linier

$$d(\mathbf{x}^T) = \mathbf{w} \cdot \mathbf{x}_i^T + 0,17185$$

dengan

$$\mathbf{w} = [\begin{matrix} 0,0844 & 0,3013 & -0,05 & 0,1349 & 0,2189 & 0 & 0,0734 & 0,0475 & -0,0777 \\ 0,1975 & -0,1878 & -0,3156 & -0,0441 & -0,2507 \end{matrix}]$$

- b. SVM dengan Fungsi Kernel *Polynomial*

$$D(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) + 0,3215$$

dengan

$$K(\mathbf{x}, \mathbf{x}_i) = (\gamma(\mathbf{x}^T \mathbf{x}_i + c))^d \quad ; \text{ dimana } \gamma = 1, c = 1 \text{ dan } d = 2$$

- c. SVM dengan Fungsi Kernel *Radial Basis Function* (RBF)

$$D(\mathbf{x}) = \sum_{i=1}^n \alpha_i y_i K(\mathbf{x}, \mathbf{x}_i) - 35,6437$$

dengan

$$K(\mathbf{x}, \mathbf{x}_i) = \exp(-\gamma \|\mathbf{x} - \mathbf{x}_i\|^2) \quad ; \text{ dimana } \gamma = 10^{-3}$$

3. Klasifikasi data penerima program beras miskin (Raskin) di Kabupaten Wonosobo tahun 2014 dengan metode *Support Vector Machine* menggunakan *LibSVM* menghasilkan nilai akurasi terbaik pada fungsi kernel *Radial Basis Function* (RBF), yaitu sebesar 83,1933 %.

6. DAFTAR PUSTAKA

- [1] Bhavsar, H dan Panchal, M.H.2012.*A Review on Support Vector Machine for Data Classification*.International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 1,Issue 10,ISSN: 2278–1323.India
- [2] Badan Pusat Statistika. 2012.*Analisis Data Kemiskinan Berdasarkan Data Pendataan Program Perlindungan Sosial (PPLS) 2011*. [online].[diakses pada 1 Maret 2015].
www.dissos.jabarpov.go.id/php/download.php?downloadfile=../files/714.pdf
- [3] Chang, C.C and Lin, C.J.2013.*LIBSVM: A Library for Support Vector Machines*. Department of Computer Science National Taiwan University. Taiwan. [online]. [diunduh pada 25 Februari 2015].Tersedia pada:
<http://www.csie.ntu.edu.tw/~cjlin/papers/libsvm.pdf>
- [4] Fan, R.E., Chen, P.H dan Lin, C.J.2005.*Working Set Selection Using Second Order Information for Training Support Vector Machines*. Journal of Machine Learning Research, 6 (2005), 1889–1918.Taiwan
- [5] Hamel, L.2009.*Knowledge Discocery with Support Vector Machines*.New Jersey.John Wiley & Sons Inc.
- [6] Jacobus, A dan Winarko, E.2014.*Penerapan Metode Support Vector Machine pada Sistem Deteksi Intrusi secara Real-time*.IJCCS,Vol.8,No.1,pp.13~24ISSN:1978-1520
- [7] Kementerian Koordinator Bidang Kesejahteraan Rakyat.2013.*Pedoman Umum Raskin 2014*. [online].[diakses pada 01 Desember 2014].Tersedia pada:
<http://www.scribd.com/doc/235170805/Pedum-Raskin-2014>
- [8] Kementerian Koordinator Bidang Kesejahteraan Rakyat.2014.*Pedoman Umum Raskin 2015*. [online].[diakses pada 23 Maret 2015].Tersedia pada:
http://www.raskin.web.id/files/pedum/Pedum_Raskin_2015.pdf
- [9] Kurniawan, D dan Supriyanto, C.2013.*Optimasi Algoritma Support Vector Machine (SVM)Menggunakan Adaboost Untuk Penilaian Risiko Kredit*. Jurnal Teknologi Informasi.Volume 9 Nomor 1, ISSN 1414-9999
- [10] Prasetyo, E.2012.*Data Mining Konsep dan Aplikasi Menggunakna Matlab*. Yogyakarta.Andi
- [11] Prasetyo, E.2014.*Data Mining Mengolah Data Menjadi Informasi Menggunakan Matlab*. Yogyakarta: Andhi Publisier
- [12] Santosa, B.2007.*Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu

- [13] Shafie, H dan Samimi, A.2013.*Introducing Linear Support Vector Algorithm And Its Implementation*.International Journal of Advanced Research in IT and Engineering.Vol. 2, No. 12,ISSN: 2278-6244
- [14] Sudarsana.2009.*Proram Raskin Sebagai Upaya Penanggulangan Kemiskinan Di Indonesia*.Universitas Sebelas Maret.Vol 21. No. 2,ISSN : 0215 – 9635
- [15] Suwondo dan Santosa, S.2014.*Credit Scoring Menggunakan Metode Support Vector Machine Dengan Teknik Seleksi Atribut Berbasis Chi Squared Statistic Dan Particle Swarm Optimization*.Jurnal Teknologi Informasi Universitas Dian Nuswantoro. Volume 10 Nomor 1, ISSN 1414-9999
- [16] To, K.N and Lim, C.C.2004.*Perturbation to enhance support vector machines for classifcation*. Journal of Computational and Applied Mathematics 163 (2004) 233–239. Australia
- [17] Yao, Y, *et al*.2014.*An Improved Grid Search Algorithm and Its Application in PCA and SVM Based Face Recognition*.Journal of Computational Information Systems 10: 3 (2014) 1219–1229.China