p-ISSN 2085-7829
e-ISSN 2798-3455

Klasifikasi Status Pembayaran Kredit Barang Elektronik dan *Furniture* Menggunakan *Support Vector Machine*

Classification of Credit Payment Status for Electronic and Furniture Using Support Vector Machine

Indah Putri Casuarina[1], Memi Nor Hayati[2], dan Surya Prangga[3]
[1,2]Laboratorium Statistika Terapan FMIPA Universitas Mulawarman
[3]Laboratorium Statistika Komputasi FMIPA Universitas Mulawarman
E-mail: indahputricasuarina@gmail.com[1]

ABSTRACT

Classification is the process of finding a model or function that can describe and differentiate data into classes. One application of classification is Support Vector Machine (SVM). SVM is a learning system that uses a hypothetical space in the form of linear functions in a high-dimensional feature space, trained with a learning algorithm based on optimization theory by implementing machine learning derived from statistical learning theory. The concept of classification with SVM is to find the best hyperplane to separate the two data classes and use a support vector approach. This study uses the proportion of the distribution of training data and testing data, namely 50%:50%, 70%:30%, 90%:10% and uses the SVM algorithm Polynomial kernel function with parameters =0.01, r=0.5, d =2, and C=1. This study aims to determine the results of the classification of the credit payment status of electronic goods and furniture and the level of classification accuracy in the SVM method. The data used is the debtor data of PT. KB Finansia Multi Finance Bontang in 2020 as many as 133 data with current and non-current credit payment status and using 7 independent variables, namely age, number of dependents, length of stay, income, years of service, large credit payments, and length of credit borrowing. The results of the SVM classification show an average accuracy value of 72.25% and the best accuracy chosen is the proportion of training data distribution and testing data 90%:10%, which is 84.62%.

Keywords: classification, machine learning, support vector machine, credit.

Pendahuluan

Pengelompokkan atau klasifikasi merupakan salah satu metode yang dapat digunakan dalam *data mining*. Teknik klasifikasi mempunyai beberapa metode, diantaranya adalah klasifikasi dengan pohon keputusan, klasifikasi *Naive Bayes*, Regresi Logistik, *K-nearest Neighbor*, *Support Vector Machine*, dan masih banyak lagi metode yang lainnya. *Support Vector Machine* (SVM) adalah sistem pembelajaran yang menggunakan ruang hipotesis berupa fungsi-fungsi linier dalam sebuah ruang fitur (*feature space*) berdimensi tinggi, dilatih dengan algoritma pembelajaran yang didasarkan pada teori optimasi dengan mengimplementasikan *machine learning* yang berasal dari teori pembelajaran statistik. SVM dalam pengklasifikasian dapat bekerja pada kasus klasifikasi linier maupun nonlinier. Konsep klasifikasi dengan SVM adalah dengan cara mencari *hyperplane* terbaik untuk memisahkan dua kelas data. SVM bekerja dengan memaksimalkan margin yang merupakan jarak pemisah antara kedua kelas data tersebut (Cristianini, 2000). SVM merupakan metode klasifikasi yang mempunyai keunggulan dari metode yang lain karena selain menggunakan jarak sebagai penentunya juga menggunakan pendekatan *support vector*. Oleh karena itu, akurasi yang dihasilkan dapat lebih baik dibandingkan dengan metode yang lain.

Dewasa ini kebutuhan masyarakat berkembang pesat, tidak hanya kebutuhan pokok tetapi juga tingkat kebutuhan sekunder bahkan tersier. Salah satu kebutuhan sekunder masyarakat di Indonesia adalah sepeda motor, mobil, barang-barang elektronik, dan *furniture* (perabotan rumah tangga). Kebutuhan tersebut tidak seluruhnya dapat dipenuhi atau dibeli secara tunai karena kebutuhan yang banyak dan kompleks belum tentu didukung dengan penghasilan yang mencukupi. Oleh karena itu, terdapat lembaga perbankan maupun lembaga keuangan nonbank yang memberikan fasilitas kredit maupun pembiayaan kepada masyarakat untuk memenuhi kebutuhannya (Fuady, 2002). Lahirnya perusahaan pembiayaan maka masyarakat yang membutuhkan barang-barang elektronik dan *furniture* tidak harus membayar secara tunai dari harga barang tersebut namun dapat secara kredit (Salim, 2006).

Kredit merupakan suatu kemampuan seseorang dalam melakukan pembelian terhadap sesuatu yang dilakukan dengan cara menerima pinjaman melalui sebuah kesepakatan pembayaran sesuai dengan jangka waktu yang ditentukan (Astiko, 1996). Dengan adanya kemudahan dalam mendapatkan pembiayaan

p-ISSN 2085-7829
e-ISSN 2798-3455

secara kredit tersebut lantas tidak pula dapat terhindar dari timbulnya risiko bagi perusahaan, termasuk perusahaan penyedia jasa kredit barang elektronik dan *furniture*. Risiko dari peminjaman kredit yang terjadi banyak berasal dari perilaku debitur yang tidak membayar angsuran dengan tepat waktu sehingga hal itulah yang dapat menyebabkan kredit macet. Kredit yang macet sangat mempengaruhi keadaan perusahaan karena berpotensi menimbulkan terjadinya penurunan pendapatan. Perusahaan harus bersikap lebih berhati-hati dalam menentukan calon debitur. Perusahaan dituntut untuk dapat memprediksi kemungkinan terjadinya kredit macet sehingga benar- benar harus selektif dalam melakukan pemilihan calon debitur (Maryandi, Supriyono, & Yaya, 2019). Hal itu agar pembayaran pinjaman kredit yang dilakukan lancar sehingga tidak menghambat keuangan atau pendapatan. Sehingga perusahaan seharusnya wajib menentukan layak atau tidaknya calon debitur dalam menerima kredit (Heryono & Kardianawati, 2018) Oleh karena itu, peneliti melakukan klasifikasi status pembayaran kredit barang Elektronik dan *Furniture* Menggunakan SVM.

Support Vector Machine (SVM)

SVM merupakan metode yang dikembangkan oleh Boser, Guyon, dan Vapnik yang dipresentasikan pertama kali pada tahun 1992 di acara *Annual Workshop on Computational Learning Theory*. Konsep khusus SVM adalah meminimalkan kesalahan klasifikasi empiris dan memaksimalkan margin geometrik. Oleh karena itu SVM disebut *Maximum Margin Classifiers.* SVM adalah metode *machine learning* yang bekerja dengan prinsip *Structural Risk Minimization* (SRM). SVM diasumsikan bahwa apabila semakin besar margin atau jarak antara *hyperplane*, maka semakin baik pula generalisasi kesalahan pengklasifikasiannya (Durgesh & Lekha, 2009)

Tujuan SVM adalah menemukan *hyperplane* terbaik yang memisahkan dua buah kelas atau kelompok pada *input space*. *Hyperplane* adalah garis batas pemisah data antar kelas. SVM bekerja dengan memaksimalkan *margin* yang merupakan jarak pemisah antara kedua kelas data tersebut. *Hyperplane* terbaik adalah *hyperplane* yang terletak di tengah-tengah antara dua set obyek dari dua kelas. *Hyperplane* pemisah terbaik antara kedua kelas dapat ditemukan dengan mengukur *margin hyperplane* tersebut dan mencari titik maksimal *margin*. *Margin* adalah jarak antara *hyperplane* terbaik dengan *pattern* terdekat dari masing-masing kelas. *Pattern* yang paling dekat disebut sebagai *support vector*. Pada dasarnya, *training* yang dilakukan oleh SVM tidak menggunakan keseluruhan data *training*, namun hanya *data training* yang terpilih untuk digunakan dimana data berupa vektor yang mampu memisahkan antara kedua kelas. Pendekatan tersebut disebut *support vector machine.* Selain itu, usaha dalam mencari lokasi *hyperplane* adalah inti dari proses pembelajaran SVM (Cristianini, 2000). Dua kondisi yang dapat diselesaikan oleh metode SVM adalah linier maupun nonlinier. Pada SVM linier, SVM dapat dibedakan menjadi dua yaitu kondisi yang dapat dipisahkan secara linier (*linearly separable data)* dan kondisi yang tidak dapat dipisahkan secara linier (*nonlinearly separable data).*

SVM Pada Data Terpisah Secara Linier

Misalkan data yang ada pada himpunan data *training* dinotasikan sebagai $\boldsymbol{x_i} \in \Re^q$ sedangkan label masing-masing kelas dinyatakan sebagai $y_i \in \{-1, +1\}$ Model linier secara umum yang dipakai dalam metode SVM untuk menghasilkan *hyperplane* (Fitriyah, Maruddani, & Warsito, 2020) yaitu:

$$y_i = \boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b \ , \ i = 1,2, \dots , n \qquad (1)$$

Keterangan:

$y \in \{-1, +1\}$: label kelas dari himpunan data

$\boldsymbol{w}$: $[w_1, w_2, \dots, w_q]$ merupakan vektor bobot yang tegak lurus terhadap *hyperplane* berupa vektor kolom berukuran $q \times 1$ dimana q merupakan banyaknya variabel bebas

$\boldsymbol{x_i}$: $[x_{i1}, x_{i2}, \dots, x_{iq}]$ merupakan vektor kolom berukuran $q \times 1$

b : *error* atau bias dimana merupakan skalar

n : banyaknya data

Kedua kelas $+1$ dan -1 diasumsikan dapat terpisah secara sempurna oleh *hyperplane* berdimensi q atau data yang bertempat di *hyperplane*, maka dinotasikan sebagai berikut:

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b = 0 \qquad (2)$$

Pattern $\boldsymbol{x_i}$ yang termasuk kelas negatif yaitu -1 dapat dirumuskan sebagai berikut:

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}^{-} + b \le -1 \qquad (3)$$

Pattern $\boldsymbol{x_i}$ yang termasuk kelas positif yaitu $+1$ dapat dirumuskan sebagai berikut:

$$\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x}^{+} + b \ge +1 \qquad (4)$$

Pada permasalahan SVM, margin terbesar dapat ditemukan dengan memaksimalkan *margin* atau jarak antara dua set objek dari kelas yang berbeda. Nilai *margin* antara bidang pembatas adalah $\frac{2}{\|w\|}$. Memaksimalkan nilai *margin* sama dengan meminimumkan nilai $\boldsymbol{w}$**.** Persamaan untuk mencari *hyperplane* terbaik dengan nilai *margin* terbesar pada permasalahan linier di dalam *primal*

p-ISSN 2085-7829
e-ISSN 2798-3455

space dapat dilihat pada persamaan (5) dengan memperhatikan *constraint* pada pertidaksamaan (6). Meminimumkan nilai $\boldsymbol{w}$ digunakan metode *Quadratic Programming* (QP) yaitu:

$$\min_{\boldsymbol{w}} \tau(\boldsymbol{w}) = \frac{1}{2}\|\boldsymbol{w}\|^2 \quad (5)$$

dengan syarat,

$$y_i\,(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b) \geq 1\,, \forall_i\,, i = 1,2,\dots,n \quad (6)$$

Persoalan tersebut akan menjadi lebih mudah diselesaikan apabila diubah ke dalam formula *lagrangian* yaitu menggunakan teknik komputasi *lagrange multiplier*. Solusi untuk optimalisasi hal tersebut adalah sebagai berikut:

$$L(w,b,\alpha) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{l}\alpha_i\,(y_i((\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b) - 1)) \quad (7)$$

Persamaan tersebut dapat dicari nilai optimalnya yang dapat dihitung dengan cara meminimumkan L terhadap $\boldsymbol{w}$ dan b sekaligus memaksimalkan L terhadap α_i. Hal tersebut sama seperti kasus *dual problem* $\max\limits_{\boldsymbol{\alpha}} \; (\min\limits_{\boldsymbol{w},b})$ dimana nilai minimum *lagrange* adalah dengan syarat sebagai berikut:

$$\max_{\boldsymbol{\alpha}} Ld = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\,\alpha_j y_i y_j (\boldsymbol{x_i}^{\boldsymbol{T}}\boldsymbol{x_j}) \quad (8)$$

dimana

$$\alpha_i \geq 0,\, i = 1,2,\dots,n \text{ dan } \sum_{i=1}^{n}\alpha_i y_j = 0 \quad (9)$$

Data *training* dengan:
$\alpha_i > 0$ berarti terletak pada *hyperplane* yang disebut *support vector*
$\alpha_i = 0$ berarti tidak terletak pada *hyperplane*
(Fitriyah, Maruddani, & Warsito, 2020)

SVM Pada Data Tidak Terpisah Secara Linier

Untuk kasus data yang tidak terpisah secara linier (*feasible*) diasumsikan bahwa kelas pada *input space* tidak dapat dipisahkan secara sempurna. Hal tersebut menyebabkan persamaan (6) tidak dapat terpenuhi. Oleh karena itu, untuk mengklasifikasi dua kelas yang terpisah secara nonlinier dan lebih tahan terhadap pencilan maka dapat menggunakan teknik *soft margin* yaitu dengan mengubah masalah *Quadratic Programming* (QP) dan batasan diatas dengan menambahkan *slack variable* yaitu ξ_i dimana $\xi_i > 0$ sehingga menjadi persamaan berikut (Cortes & Vapnik, 1995):

$$\min \tau(\boldsymbol{w},\xi) = \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{n}\xi_i \quad (10)$$

dengan batasan

$$y_i\,(\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b) \geq 1 - \xi_i, \forall_i\,, i = 1,2,\dots n \quad (11)$$

parameter C berfungsi untuk mengontrol optimasi (*trade off*) antara *margin* dan kesalahan klasifikasi ξ. Semakin besar nilai C maka semakin besar pula penalti terhadap kesalahan (*error*) klasifikasi.

Untuk mencari nilai $\boldsymbol{w}$ dan b yang optimal untuk persamaan (10) dengan batas bawah persamaan (11) dapat diselesaikan dengan *lagrange multiplier* sebagai berikut (Cortes & Vapnik, 1995):

$$\min L(\boldsymbol{w},b,\alpha) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n}\alpha_i\,(y_i((\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} + b) - 1)) \quad (12)$$

$$\min L(\boldsymbol{w},b,\alpha) = \frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n}\alpha_i\,y_i\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} - b\sum_{i=1}^{n}\alpha_i y_i + \sum_{i=1}^{n}\alpha_i \quad (13)$$

dengan $\alpha_i \geq 0$ (nilai koefisien *lagrange*)

Dari persamaan (13), kemudian dicari nilai b yang meminimumkan L sebagai berikut:

$$\frac{\partial}{\partial b}L(\boldsymbol{w},b,\alpha) = 0$$

$$\frac{\partial}{\partial b}\Big(\frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n}\alpha_i\,y_i\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} - b\sum_{i=1}^{n}\alpha_i y_i + \sum_{i=1}^{n}\alpha_i\Big) = 0$$

$$\sum_{i=1}^{n}\alpha_i y_i = 0 \quad (14)$$

Dari persamaan (13), juga dicari nilai **w** yang meminimumkan L sebagai berikut:

$$\frac{\partial}{\partial \boldsymbol{w}}L(\boldsymbol{w},b,\alpha) = 0$$

$$\frac{\partial}{\partial w}\Big(\frac{1}{2}\|\boldsymbol{w}\|^2 - \sum_{i=1}^{n}\alpha_i\,y_i\boldsymbol{w}^{\boldsymbol{T}}\boldsymbol{x_i} - b\sum_{i=1}^{n}\alpha_i y_i + \sum_{i=1}^{n}\alpha_i\Big)=0$$

$$\boldsymbol{w} - \sum_{i=1}^{n}\alpha_i\,y_i\boldsymbol{x}_i = 0$$

$$\boldsymbol{w} = \sum_{i=1}^{n}\alpha_i\,y_i\boldsymbol{x}_i \quad (15)$$

Karena $\boldsymbol{w}$ sering bernilai besar dan nilai α_i terhingga, maka *Lagrange* L *(primal problem)* diubah ke dalam *Lagrange* L_d*(dual problem)* untuk mencari pemisah terbaik dengan demikian persamaan dimodifikasi sebagai berikut (Cortes & Vapnik, 1995)

$$L_d = \sum_{i=1}^{n}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{n}\alpha_i\,\alpha_j y_i y_j (\boldsymbol{x_i}^{\boldsymbol{T}}\boldsymbol{x_j}) \quad (16)$$

dengan batas

$$0 \leq \alpha_i \leq C, i = 1,2,\dots,n \;dan\; \sum_{i=1}^{n}\alpha_i y_j = 0 \quad (17)$$

Berdasarkan persamaan (17) maka akan diperoleh nilai α_i (nilai koefisien *lagrange*) untuk setiap data *training*. Nilai tersebut yang digunakan untuk menentukan $\boldsymbol{w}$**.** Data *training* yang memiliki nilai $\alpha_i > 0$ disebut *support vector*. Setelah solusi permasalahan QP ditemukan (nilai α_i), maka kelas dari suatu data x testing dapat ditentukan dengan persamaan sebagai berikut:

$$f(\boldsymbol{x}) = \sum_{s=1}^{v}\alpha_s\,y_s\boldsymbol{x}_s.\boldsymbol{x}_t + b \quad ; \quad s = 1,2,\dots,v\,; t = 1,2,\dots,n_t \quad (18)$$

dimana:

$\boldsymbol{x}_s$: data training yang termasuk support vector dimana $\boldsymbol{x_s} = [x_{s1}, x_{s2}, \dots, x_{sq}]$

p-ISSN 2085-7829
e-ISSN 2798-3455

dan merupakan *vector* kolom berukuran $q \times 1$

y_s : label kelas dari himpunan data $\boldsymbol{x_s}$; $y_s \in \{-1, +1\}$

α_s : α_i yang termasuk support vector

v : jumlah support vector

x_t : data testing (data yang akan diklasifikasikan) dimana $[x_{t1}, x_{t2}, \dots, x_{tq}]$

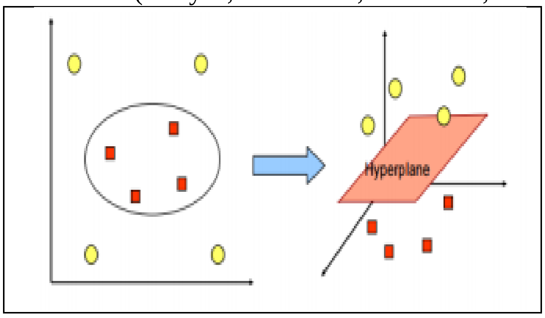
(Fitriyah, Maruddani, & Warsito, 2020)

Nonlinear Classification pada SVM

Prinsip dasar SVM adalah *linear classifier*. Namun kasus dunia nyata (*real world problem*) pada umumnya bersifat *nonlinear*. Oleh karena itu, SVM dapat dikembangkan untuk menyelesaikan *nonlinear classifier*. SVM dimodifikasi dengan memasukkan fungsi *kernel* yang sering disebut dengan konsep *Kernel Trick* dimana pada ruang berdimensi lebih tinggi dapat menangani *nonlinear classifier*. Berikut fungsi yang secara umum mengkonversi himpunan data pada ruang masukan (*input space*) ke dalam ruang fitur (*feature space*) yang berdimensi lebih tinggi dan diilustrasikan pada Gambar 1:

$$\Phi: \Re^q \to \Re^p, \text{dimana } q < p \qquad (19)$$

(Fitriyah, Maruddani, & Warsito, 2020)



Gambar 1 Fungsi Φ yang Memetakan Data ke Ruang Vektor yang Berdimensi Lebih Tinggi Sehingga Kedua Kelas Dapat Dipisahkan Secara Linier oleh Sebuah *Hyperplane*
(Handoko, Nugroho, & Witarto, 2003).

Pada Gambar 1 diperlihatkan bahwa data pada kelas kuning dan data pada kelas merah yang berada pada *input space* berdimensi dua tidak dapat dipisahkan secara linier. Kemudian fungsi Φ yang memetakan tiap data pada *input space* ke ruang vektor baru yang berdimensi lebih tinggi yaitu berdimensi tiga. Kedua kelas pun dapat dipisahkan secara linier oleh sebuah *hyperplane*. Pada ruang vektor yang baru, SVM mencari *hyperplane* yang memisahkan kedua kelas secara linier (Santosa, 2007).

Pada dasarnya proses pembelajaran pada SVM dalam menemukan *support vector* hanya bergantung pada *dot product* dari data pada ruang fitur, yaitu $\Phi(\boldsymbol{x_i}).\Phi(\boldsymbol{x_j})$. Oleh karena transformasi Φ tidak dapat diketahui dan sangat sulit dipahami maka perhitungan *dot product* dapat digantikan dengan fungsi *Kernel* $\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$ yang mendefinisikan secara implisit fungsi transformasi Φ. Formulasi *Kernel Trick* adalah sebagai berikut(Cortes & Vapnik, 1995):

$$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \Phi(\boldsymbol{x_i}).\Phi(\boldsymbol{x_j}) \qquad (20)$$

Keterangan:

$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$: Fungsi *kernel*

$\Phi(\boldsymbol{x_i}).\Phi(\boldsymbol{x_j})$: *dot product* dari data pada ruang fitur

$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$ merupakan *kernel* yang akan digunakan untuk mengubah data dimana data input hasil pemetaan ke ruang fitur diharapkan akan dapat terpisah secara linear sehingga dapat dicari *hyperplane* yang optimal. Fungsi *kernel* memungkinkan dalam kasus yang *non-linearly* pada ruang input, diharapkan dapat menjadi *linearly separable* pada ruang fitur. Selanjutnya, dapat menghitung *hyperplane* sebagai *decision boundary* secara efisien.

Penggunaan fungsi Kernel pada persamaan (20) mengakibatkan persamaan *Lagrange* pada (16) dimodifikasi menjadi:

$$L_d = \sum_{i=1}^{n} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{n} \alpha_i \alpha_j y_i y_j \mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) \qquad (21)$$

Fungsi yang dihasilkan dari *training* adalah sebagai berikut:

$$sign(f(\boldsymbol{x})) = \sum_{i,j=1}^{n} \alpha_i y_i \mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) + b\ ; \quad i, j = 1, 2, \dots, n \qquad (22)$$

dengan batas

$$0 \le \alpha_i \le C, i, j = 1, 2, \dots, n \ dan \ \sum_{i=1}^{n} \alpha_i y_i = 0 \qquad (23)$$

Berdasarkan persamaan (22) dan (23) maka akan diperoleh nilai α_i (nilai koefisien *lagrange*) untuk setiap data *training*. Data *training* yang memiliki nilai $\alpha_i > 0$ disebut *support vector*. Setelah solusi permasalahan QP ditemukan (nilai α_i), maka kelas dari suatu data x testing dapat ditentukan dengan persamaan sebagai berikut:

$$f(\boldsymbol{x}) = \sum_{s=1}^{v} \alpha_s y_s \mathrm{K}(\boldsymbol{x_s}.\boldsymbol{x_t}) + b\ ;$$
$$s = 1, 2, \dots, v \ t = 1, 2, \dots, \boldsymbol{n_t} \qquad (24)$$

dimana:

$\boldsymbol{x_s}$: data training yang termasuk support vector dimana $\boldsymbol{x_s} = [x_{s1}, x_{s2}, \dots, x_{sq}]$ dan merupakan *vector* kolom berukuran $q \times 1$

y_s : label kelas dari himpunan data $\boldsymbol{x_s}$; $y_s \in \{-1, +1\}$

α_s : α_i yang termasuk support vector

v : jumlah support vector

p-ISSN 2085-7829
e-ISSN 2798-3455

$\boldsymbol{x}_t$: data testing (data yang akan diklasifikasikan) dimana $[x_{t1}, x_{ts2}, \dots, x_{tq}]$
$\boldsymbol{n}_t$: jumlah data *testing*
$f(\boldsymbol{x}) \geq 0$: kelas positif
$f(\boldsymbol{x}) < 0$: kelas negatif

Beberapa fungsi *kernel* yang dapat digunakan pada umumnya yaitu:

1. Linier
$$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \boldsymbol{x_i}^{\boldsymbol{T}} \boldsymbol{x_j} \tag{25}$$
2. *Polynomial*
$$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = (\gamma \boldsymbol{x_i}^{\boldsymbol{T}} \boldsymbol{x_j} + r)^d, \gamma > 0 \quad 26)$$
3. *Radial Basis Function atau Gaussian*
$$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \exp\left(-\gamma \|\boldsymbol{x_i} - \boldsymbol{x_j}\|^2\right), \gamma > 0 \tag{27}$$
4. *Sigmoid* atau *Kernel Tangent Hyperbolic*
$$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j}) = \tanh[\gamma \boldsymbol{x_i}^{\boldsymbol{T}} \boldsymbol{x_j} + r] \tag{28}$$

Keterangan:
$\mathrm{K}(\boldsymbol{x_i}, \boldsymbol{x_j})$ = Fungsi Kernel
$\boldsymbol{x_i}, \boldsymbol{x_j}$ = Pasangan Data Training
r, γ, d = Parameter SVM pada Fungsi Kernel

Pemilihan fungsi *kernel* yang tepat menjadi hal yang sangat penting karena hal tersebut yang akan menentukan *feature space*, dimana fungsi *classifier* akan dicari. Apabila fungsi *kernel* cocok maka SVM akan beroperasi secara optimal (Santosa, 2007). Fungsi *kernel* RBF memiliki kelebihan dimana secara otomatis menentukan nilai rentang tak terhingga dan menghindari *overfitting* dengan memilih nilai parameter C dan γ dengan tepat. RBF dapat memetakan hubungan tidak linier, serta RBF lebih *robust* terhadap *outlier* (Prasetyo, 2012).

Evaluasi Performansi Hasil Klasifikasi

Evaluasi terhadap hasil klasifikasi perlu dilakukan untuk mengetahui tingkat akurasi yang dihasilkan pada penelitian. Akurasi klasifikasi adalah ukuran ketepatan klasifikasi yang menunjukkan performansi teknik klasifikasi secara keseluruhan (Handoko, Nugroho, & Witarto, 2003).

$$Akurasi = \frac{TP+TN}{TP+FP+TN+FN} \tag{29}$$

Keterangan:
True Positive (TP) :proporsi positif dalam data set yang diklasifikasikan positif.
True Negative (TN) :proporsi negatif dalam data set yang diklasifikasikan negatif.
False Positive (FP) :proporsi negatif dalam data set yang diklasifikasikan potitif.
False Negative (FN) :proporsi positif dalam data set yang diklasifikasikan negatif.

Data *Training* dan Data *Testing*

Data *training* berguna untuk membangun model klasifikasi. Data *testing* berguna untuk menghitung *error rate* dari klasifikasi. Model klasifikasi dibangun berdasarkan data *training* kemudian kinerjanya diukur berdasarkan data *testing*. Proporsi pembagian data *training* dan *testing* biasanya adalah diskrit, misalnya (70:30) artinya 70% dari keseluruhan data menjadi data *training* dan 30% lainnya menjadi data *testing* (Prasetyo, 2012).

Standarisasi Data

Teknik standarisasi data digunakan agar semua variabel terdapat di dalam jangkauan yang sama sehingga proporsi pengaruh pada fungsi klasifikator dapat seimbang. Tanpa dilakukan standarisasi data, bisa jadi salah satu variabel akan mendominasi fungsi pada klasifikator. (Prasetyo, 2012).

Cara menghitung nilai standarisasi data ($\hat{x}_{ig}$) adalah sebagai berikut:

$$\bar{X}_g = \frac{1}{n} \times \sum_{i=1}^{n} x_{ig} \tag{30}$$
$$s_g = \sqrt{\frac{1}{n-1} \times \sum_{i=1}^{n} (x_{ig} - \bar{X}_g)^2} \tag{31}$$
$$\hat{x}_{ig} = \frac{x_{ig} - \bar{X}_g}{s_g} \tag{32}$$

dimana:
$\bar{X}_g$:Rata-rata pada variabel ke-g
x_{ig} :Data ke-i pada variabel ke-g
n :Banyak pengamatan
s_g :Simpangan Baku pada variabel ke-g
$\hat{x}_{ig}$:Standarisasi Data ke-i pada variabel ke-g

Metode Penelitian

Sampel dalam penelitian ini adalah 133 debitur di PT. KB Finansia Multi Finance Bontang Tahun 2020. Data diperoleh di PT. KB Finansia Multi Finance Bontang. Sedangkan Variabel penelitian yang digunakan dalam penelitian ini terdiri dari variabel terikat (*Y*) dengan dua kategori dan tujuh variabel bebas (*X*) yang secara rinci pada Tabel 1 berikut:

Tabel 1 Variabel Penelitian

Variabel	Keterangan	Tipe Data
Y	Status pembayaran kredit dibagi menjadi 2 kategori yakni kategori kredit lancar (L) dan kategori kredit tidak lancar (TL)	Nominal
X_1	Usia	Diskrit
X_2	Jumlah tanggungan	Diskrit
X_3	Lama tinggal	Diskrit
X_4	Masa kerja	Diskrit
X_5	Pendapatan	Diskrit
X_6	Besar Pembayaran	Diskrit
X_7	Lama peminjaman	Diskrit

p-ISSN 2085-7829
e-ISSN 2798-3455

Penelitian ini menggunakan bantuan *Microsoft Excel* dan *software R* dengan langkah-langkah analisis sebagai berikut:

1. Standarisasi Data
2. Pengacakan Data
3. Membagi Data *Training* dan Data *Testing* berdasarkan proporsi
4. Menentukan parameter fungsi Kernel. Dalam hal ini menggunakan fungsi Kernel *polynomial* pada persamaan (26) dengan parameter $\gamma = 0{,}01$, $r = 0{,}5$, dan $d = 2$.
5. Menentukan parameter penalti C, dalam hal ini C=1.
6. Membangun fungsi SVM menggunakan data *training* berdasarkan solusi QP pada persamaan (22) dengan syarat pada persamaan (23) untuk mendapatkan nilai α_i dan b dengan bantuan *software* R.
7. Menentukan hasil klasifikasi berdasarkan persamaan (24) menggunakan data *testing*.
8. Menghitung akurasi hasil klasifikasi menggunakan persamaan (29) dan memilih akurasi yang terbaik.

Hasil dan Pembahasan

Standarisasi Data

Standarisasi data dilakukan agar semua variabel berada dalam jangkauan yang sama sehingga proporsi pengaruh pada fungsi klasifikator dapat seimbang. Proses standarisasi data hanya dilakukan pada seluruh variabel bebas menggunakan persamaan (32).

Pengacakan Data serta Pembagian Data *Training* dan Data *Testing*

Pengacakan data dilakukan menggunakan *software R* yang bertujuan agar semua data memiliki kesempatan yang sama untuk menjadi data *training* dan data *testing*. Data hasil pengacakan tersebut kemudian dilakukan pembagian data *training* dan data *testing* dengan menggunakan 3 proporsi yaitu 50% : 50%, 70% : 30%, dan 90% : 10%. Data yang berada pada urutan pertama akan digunakan menjadi data *training* dan sisanya akan menjadi data *testing*.

Klasifikasi SVM

Klasifikasi SVM pada penelitian menggunakan fungsi kernel *Polynomial* berdasarkan persamaan (26). Parameter SVM fungsi kernel *Polynomial* yang digunakan adalah $\gamma = 0{,}01$, $r = 0{,}5$, dan $d = 2$ sedangkan parameter pinalti yang digunakan adalah $C = 1$. Pada Metode SVM fungsi kernel *Polynomial* menggunakan data *training* akan dibangun berdasarkan fungsi SVM pada persamaan (22). Berdasarkan fungsi SVM dengan syarat pada persamaan (23) menggunakan pendekatan QP dengan bantuan *software* R akan membentuk model SVM fungsi kernel *polynomial* sebagai berikut:

Tabel 2 Model SVM Fungsi Kernel *Polynomial*

Proporsi	Model SVM Fungsi Kernel *Polynomial* $(\boldsymbol{\gamma = 0{,}01, r = 0{,}5,\ d = 2\ , C = 1})$	Jumlah *Support Vector*
50%:50% (67 *training*)	$sign(f(x)) = \sum_{i,j=1}^{67} \alpha_i\, y_i (0{,}01 \times \Phi({x_i}^T)\Phi(x_j) + 0{,}5)^2 + 0{,}933$	50
70%:30% (93 *training*)	$sign(f(x)) = \sum_{i,j=1}^{93} \alpha_i\, y_i (0{,}01 \times \Phi({x_i}^T)\Phi(x_j) + 0{,}5)^2 + 0{,}844$	75
90%:10% (120 *training*)	$sign(f(x)) = \sum_{i,j=1}^{120} \alpha_i\, y_i (0{,}01 \times \Phi({x_i}^T)\Phi(x_j) + 0{,}5)^2 + 0{,}815$	96

Berdasarkan model SVM fungsi kernel *polynomial* dan jumlah *support vector* pada Tabel 2, α_i yang termasuk dalam *support vector* (α_s) dan nilai b yang selanjutnya akan digunakan untuk mengklasifikasikan data *testing* menggunakan persamaan (24) dengan bantuan *software* R.

Pemilihan Akurasi Terbaik

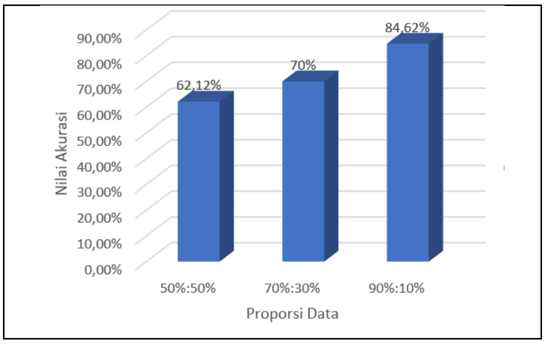
Pemilihan akurasi terbaik dilakukan dengan melihat dan membandingkan nilai akurasi pada klasifikasi SVM fungsi kernel *polynomial* berdasarkan proporsi data. Berikut nilai akurasi dapat dilihat pada Tabel 3.

Tabel 3 Nilai Akurasi SVM

Proporsi	SVM *Polynomial* $(\boldsymbol{\gamma = 0{,}01, r = 0{,}5,\ d = 2\ ,\ C = 1})$
50%:50%	62,12%
70%:30%	70%
90%:10%	84,62%
Rata-rata	72,25%

Berdasarkan Tabel 3 dapat diketahui bahwa klasifikasi SVM fungsi kernel *Polynomial* dengan parameter $\gamma = 0{,}01, r = 0{,}5,\ d = 2$, dan nilai pinalti $C = 1$ memiliki rata-rata nilai akurasi sebesar 72,25% dengan akurasi terrtingginya adalah 84,62% dan terendahnya adalah 62,12%. Berikut Gambar 2 untuk dapat melihat lebih jelas perbandingannya

p-ISSN 2085-7829
e-ISSN 2798-3455



Gambar 2 Perbandingan *Nilai Akurasi*

Berdasarkan Gambar 2 dan Tabel 3 dapat diambil kesimpulan bahwa akurasi terbaik yang dipilih dari klasifikasi SVM fungsi kernel *Polynomial* adalah dengan proporsi data 90%:10% yaitu sebesar 84,62%.

Kesimpulan dan Saran

Hasil pengklasifikasian status pembayaran kredit barang elektronik dan *furniture* oleh debitur pada PT. KB Finansia Multi Finance Bontang Tahun 2020 berdasarkan akurasi yang terbaik dari SVM fungsi kernel *polynomial* yaitu dengan proporsi data 90%:10% menghasilkan 11 debitur tepat diklasifikasikan serta terdapat 2 debitur tidak tepat diklasifikasikan. Adapun hasil pengukuran tingkat akurasi yaitu sebesar 84,62%.

Saran pada penelitian ini, yaitu dalam penelitian selanjutnya dapat menggunakan kombinasi nilai parameter (*tuning)* kernel dan fungsi kernel yang lainnya pada metode *Support Vector Machine.* Selain itu dapat menggunakan metode klasifikasi lain seperti Artificial Neural Network (ANN), *Modified K-Nearest Neighbors* atau metode lainnya.

Daftar Pustaka

Astiko. (1996). *Manajaemen Pengkreditan*. Yogyakarta: Andi Offset

Chang, C.C., Hsu, C. W., & Lin, C. J. (2003). *A Practical Guide to Support Vector Classification.* Taiwan*: Departemen of Computer Science National Taiwan University*

Cortes, C., & Vapnik, V. (1995). *Support Vector Machine Networks*. *Machine Learning. Vol 20, 273-297*

Cristianini, N. (2000). *An Introduction to Support Vector Machine*. UK: *Cambridge University Press*

Durgesh, K.S., & Lekha, B. (2009). *Data Classification Using Support Vector Machine. Journal of Theoretical and Applied Information Technology*

Fitriyah, N., Maruddani, D.A.I., & Warsito, B. (2020). Analisis Sentimen Gojek Pada Media Sosial Twitter Dengan Klasifikasi *Support Vector Machine* (SVM). *Jurnal Gaussian*. *9(3),376-390*. Science (ETCS), International Workshop., vol 1, pp.836-839

Fuady, M. (2002). *Hukum Tentang Pembiayaan Dalam Teori Dan Praktek*. Bandung: Citra Aditya Bakti

Gorenescu, F. (2011). *Data Mining: Concept, Model, And Technique.* Verlag Berlin Heidelberg: Springer.

Handoko, D., Nugroho, A.S., & Witarto, A.B. (2003). Support Vector Machine dan Aplikasinya Dalam Bioinformatika. *Jurnal Ilmu Komputer Vol. 1*

Heryono & Kardianawati, A. (2018). Implementasi Metode *Naïve Bayes* Untuk Klasifikasi Kredit Motor. *Journal of Information System Vol. 3, pp 10-21*

Maryandi, M.S, Supriyono, E., & Yaya, R. (2019). Analisis Faktor Internal Bank Terhadap *Non Performing Loan* Berdasarkan *Generalized Method of Moment. Jurnal Keuangan dan Perbankan. 20(3),496-506*

Prasetyo, E. (2012). *Data Mining*: *Konsep dan Aplikasi Menggunakan Matlab*. Yogyakarta: Andi

Santosa, B. (2007). *Data mining: Teknik Pemanfaatan Data Untuk Keperluan Bisnis*. Yogyakarta: Graha Ilmu

Zhang, Y., & Wang, W. (2010). Pattern Classification of Electroencephalography from the Typical Specialized Students. Education Technology and Computer Science (ETCS), International Workshop., vol 1, pp.836-839

p-ISSN 2085-7829
e-ISSN 2798-3455