ISSN: 2339-2541
JURNAL GAUSSIAN, Volume 12, Nomor 3, Tahun 2023, Halaman 372 - 381
Online di: https://ejournal3.undip.ac.id/index.php/gaussian/

PERBANDINGAN METODE K-NEAREST NEIGHBOR DAN SUPPORT VECTOR MACHINES PADA STATUS PENERIMAAN BANTUAN DARI PEMERINTAH

Wahyu Anwar Ridho[1*], Triastuti Wuryandari [2], Arief Rachman Hakim[3]
[1,2,3] Departemen Statistika, Fakultas Sains dan Matematika, Universitas Diponegoro
***e-mail :** wahyuanwarridho@gmail.com

DOI: 10.14710/j.gauss.12.3.372-381

Article Info:
Received: 2022-10-17
Accepted: 2024-02-13
Available Online: 2024-02-26

Keywords:
SUSENAS; *Classification;*
K-Nearest Neighbor;
Support Vector Machines

Abstract: The government program in the form of social assistance (bansos) is part of the effort to improve the welfare of the community and ensure basic needs and improve the standard of living of the recipients. However, there are often cases of mistargeting of social assistance programs by the government. Improper data management and Data Terpadu Kesejahteraan Sosial (DTKS) which are not used as the cause of the distribution of social assistance are not well targeted. The data can be analyzed using the classification method to determine whether or not the family accepts the ban from the government. This study classifies the SUSENAS data by comparing K-Nearest Neighbor (KNN) and Support Vector Machines (SVM). The advantage of the KNN method lies in the level of accuracy to solve problems with large data while the SVM method has better performance in various fields of application such as bioinformacs, handwriting recognition, text classification and so on. Based on training data and testing data comparison 85%:15% showed that KNN method had a better classification performance than the SVM method. The accuracy value of KNN method is 80,95% higher than the accuracy value of SVM method is 78,79%.

1. PENDAHULUAN

Program pemerintah berupa bantuan sosial (bansos) merupakan bagian dari usaha guna menyejahterakan masyarakat dan menjamin kebutuhan dasar serta meningkatkan taraf hidup penerimanya. Program yang seharusnya ditujukan untuk masyarakat miskin, ternyata masih terdapat kasus dimana program bantuan sosial dapat dinikmati oleh masyarakat yang secara ekonomi dapat dikatakan sebagai keluarga mampu. Badan Pemeriksa Keuangan (BPK) menemukan kesalahan penyaluran bansos pemerintah yang mengakibatkan kerugian negara hingga 6,9 triliun rupiah. Persoalan penyaluran bansos terkait pada pengelolaan data. Data Terpadu Kesejahteraan Sosial (DTKS) kemungkinan besar tidak diperbarui sehingga menyebabkan penyaluran bansos tidak tepat sasaran (Purnama, 2022). Permasalahan penyaluran bansos tidak tepat sasaran dapat diminimalisir dengan pembaruan data di DTKS secara terpadu. Data tersebut juga dapat dianalisis menggunakan metode klasifikasi untuk mengetahui tepat tidaknya keluarga menerima bansos dari pemerintah.

Penelitian mengenai klasifikasi status penerimaan bantuan dari pemerintah perlu dilakukan untuk mengetahui tepat atau tidaknya suatu keluarga menerima bantuan dengan melihat akurasi klasifikasi tersebut. Penelitian ini akan membandingkan kinerja dua metode klasifikasi yaitu *K-Nearest Neighbor* (KNN) dan *Support Vector Machines* (SVM). KNN merupakan salah satu metode klasifikasi dimana sebuah objek baru diberi label berdasarkan k objek tetangga terdekatnya. Algoritma KNN adalah salah satu algoritma yang paling sederhana dari semua algoritma pembelajaran mesin yang lain, karena algoritma KNN hanya mengklasifikasikan suatu objek dengan suara mayoritas tetangga terdekatnya. SVM merupakan algoritma yang bekerja dengan menggunakan pemetaan nonlinier untuk

mengubah data pelatihan asli ke dimensi yang lebih tinggi. SVM dapat memberikan kinerja generalisasi yang baik dalam masalah penanganan pola, yang memberikan fitur unik di antara mesin pembelajaran yang lain (Gorunescu, 2011).

Penelitian sebelumnya yang membandingkan metode KNN dan SVM telah dilakukan oleh Aulia (2015) pada klasifikasi penyakit Diabetes Retinopati. Penelitian Aulia (2015) menggunakan 60 citra data latih dan 160 citra data uji menghasilkan akurasi maksimum 62% pada metode SVM dan akurasi maksimum 65% pada metode KNN dengan k=9. Penelitian ini akan mengklasifikasikan status penerimaan bantuan pemerintah dengan membandingkan metode KNN dan SVM, sehingga penelitian ini diharapkan dapat mengetahui kinerja metode klasifikasi terbaik dalam mengklasifikasikan status penerimaan bantuan pemerintah.

2. TINJAUAN PUSTAKA

Survei Sosial Ekonomi Nasional (SUSENAS) merupakan survei yang dirancang untuk mengumpulkan data sosial kependudukan yang relatif sangat luas. Data yang dikumpulkan antara lain menyangkut bidang-bidang pendidikan,kesehatan/gizi, perumahan, sosial ekonomi lainnya, kegiatan sosial budaya, konsumsi/pengeluaran dan pendapatan rumah tangga, perjalanan, dan pendapat masyarakat mengenai kesejahteraan rumah tangganya (BPS, 2022).

Data mining merupakan proses untuk memanipulasi data dengan mengekstraksi informasi yang sebelumnya tidak diketahui dari dataset berukuran besar. Metode pada data mining di kelompokkan dalam dua kategori yaitu *predictive* dan *descriptive*. Kategori *predictive* menggunakan variabel yang sudah ada untuk memprediksi suatu nilai dari variabel lain, sedangkan kategori *descriptive* mengungkap pola tersembunyi dari data sehingga mudah dipahami oleh pengguna. Kategori *predictive techniques* dapat dibagi dalam dua kelompok utama, operasi *classification* atau *discrimination* dan operasi *prediction* atau *regression* (Gorunescu, 2011). Klasifikasi adalah bentuk analisis data yang mengekstrak model dengan menggambarkan kelas data penting, dimana pengklasifikasi memprediksi label kelas bertipe kategori (Han *et al*., 2011).

Synthetic Minority Over-sampling Technique (SMOTE) adalah metode *over-sampling* dimana data pada kelas minoritas diperbanyak dengan menggunakan data sintetik yang berasal dari replikasi data pada kelas minoritas. *Over-sampling* pada SMOTE mengambil *instance* dari kelas minoritas lalu mencari *k-nearest neighbor* dari setiap *instance*, kemudian menghasilkan *instance* sintetik kelas minoritas. Algoritma SMOTE akan mengambil nilai selisih antara vektor dari fitur pada kelas minoritas dan nilai *nearest neighbor* dari kelas minoritas, lalu mengalikan nilai tersebut dengan angka acak antara 0 sampai 1. Hasil perhitungan tersebut ditambahkan dengan vektor fiturnya sehingga didapatkan hasil nilai vektor yang baru (Jishan, et al., 2015).

$$\boldsymbol{x}_{new} = \boldsymbol{x}_i + (\widehat{\boldsymbol{x}}_i - \boldsymbol{x}_i) \times \delta \quad (1)$$

dengan

$\boldsymbol{x}_i$: vektor dari fitur pada kelas minoritas

$\widehat{\boldsymbol{x}}_i$: salah satu *k-nearest neighbor* untuk $\boldsymbol{x}_i$

δ : angka acak antara [0,1]

Algoritma KNN merupakan metode yang menggunakan algoritma *supervised* dimana hasil sampel uji yang baru diklasifikasikan berdasarkan mayoritas dari kategori pada KNN (Han *et al*,. 2011). Tujuan KNN adalah mengklasifikasikan objek baru berdasakan atribut dan sampel latih. Algoritma KNN menggunakan klasifikasi ketetanggaan

sebagai nilai prediksi dari sampel uji yang baru. Penelitian ini menggunakan jarak *euclidean* yang diberikan oleh persamaan berikut.

$$d_{ij} = \sqrt{\sum_{k=1}^{m}(x_{ik} - x_{jk})^2} \quad (2)$$

dimana

d_{ij} : jarak *euclidean* objek data ke-*i* dan objek data ke-*j*
m : banyaknya peubah/parameter yang digunakan
x_{ik} : objek data ke-*i* pada peubah ke-*k*
x_{jk} : objek data ke-*j* pada peubah ke-*k*

Himpunan berpasangan n $(x_1, \theta_1), \dots, (x_n, \theta_n)$ diberikan, dimana x_i' merupakan ruang matriks X dengan mendefinisikan matriks d, dan θ_i' mengambil nilai di $\{1,2, \dots, M\}$. Setiap θ_i dianggap sebagai indeks dari kategori individu ke-i, dan setiap x_i merupakan hasil dari serangkaian pengukuran yang dilakukan. Pasangan baru (x, θ) diberikan, dimana hanya x yang dapat diamati oleh peneliti dan mengestimasi θ dengan memanfaatkan informasi yang terkandung di dalam kumpulan titik yang diklasifikasikan secara benar.

$x_n' \in \{x_1, x_2, \dots, x_n\}$

merupakan tetangga terdekat untuk x jika:

$\min d(x_i, x) = d(x_n', x), i = 1,2, \dots, n$ (Cover, 1967).

SVM adalah algoritma yang bekerja dengan menggunakan pemetaan nonlinier untuk mengubah data pelatihan asli ke dimensi yang lebih tinggi. Pada dimensi baru ini, SVM mencari *hyperplane* pemisah optimal linier yang memisahkan satu kelas dari kelas yang lain. Pemetaan nonlinier yang sesuai ke dimensi yang lebih tinggi dari dua kelas selalu dapat dipisahkan oleh *hyperplane*. SVM menemukan *hyperplane* menggunakan *support vector* dan *margin* yang didefinisikan oleh *support vector* (Han *et al*., 2011). Misalkan himpunan data *training* dari dua kelas yang akan diklasifikasikan dengan SVM,

$$D = \{(x_1, y_1), \dots, (x_l, y_l)\},\ x \epsilon R^n,\ y \epsilon \{-1,1\}, \quad (3)$$

dengan *hyperplane*,

$$\boldsymbol{w}.\boldsymbol{x} + b = 0. \quad (4)$$

Himpunan vektor dikatakan dipisahkan secara optimal oleh *hyperplane* jika dipisahkan tanpa kesalahan dan jarak antara vektor terdekat dengan *hyperplane* memiliki nilai maksimal. Hyperplane pemisah dirumuskan dengan fungsi,

$$y_i(\boldsymbol{w}.\boldsymbol{x_i} + b) \geq 1, i = 1, 2, \dots, l. \quad (5)$$

Jarak titik $\boldsymbol{x}$ dari hyperplane dirumuskan,

$$d(\boldsymbol{w}.b, \boldsymbol{x}) = \frac{|\boldsymbol{w}.\boldsymbol{x_i} + b|}{||\boldsymbol{w}||}. \quad (6)$$

Sehingga, *hyperplane* pemisah optimal adalah *hyperplane* yang meminimalkan,

$$\phi(\boldsymbol{w}) = \frac{1}{2}||\boldsymbol{w}||^2. \quad (7)$$

Solusi optimasi dari persamaan 7 dengan syarat persamaan 5 adalah menggunaan fungsi Lagrange,

$$\phi(\boldsymbol{w}, b, \alpha) = \frac{1}{2}\left|\left|\boldsymbol{w}\right|\right|^2 - \sum_{i=1}^{l} \alpha_i(y_i[\boldsymbol{w}.\boldsymbol{x_i} + b] - 1), \quad (8)$$

dengan α merupakan pengali Lagrange. Fungsi Lagrange diminimumkan terhadap $\boldsymbol{w}$ dan b dengan syarat $\alpha \geq 0$. Dualitas Lagrangian memungkinkan *primal problem* untuk ditransformasikan ke *dual problem* sehingga lebih mudah diselesaikan. Fungsi *dual problem* diberikan oleh,

$$\max_{\alpha} W(\alpha) = \max_{\alpha}(\min_{\boldsymbol{w},b} \phi(\boldsymbol{w}, b, \alpha)). \quad (9)$$

Fungsi *dual problem* diminimumkan terhadap *w* dan *b* sehingga diperoleh $\frac{\delta\phi}{\delta\boldsymbol{w}} = 0$ dan $\frac{\delta\phi}{\delta\boldsymbol{b}} = 0$. Turunan pertama fungsi *dual problem* terhadap **w**,

$$\boldsymbol{w} = \sum_{i=1}^{l} \alpha_i y_i \boldsymbol{x_i}, \quad (10)$$

turunan pertama fungsi *dual problem* terhadap **b**,

$$\sum_{i=1}^{l} \alpha_i y_i = 0. \quad (11)$$

Fungsi *dual problem* dirumuskan oleh,

$$\max_{\alpha} W(\alpha) = \max_{\alpha} -\frac{1}{2}\sum_{i=1}^{l}\sum_{j=1}^{l} \alpha_i \alpha_j y_i y_j \left(\boldsymbol{x_i}.\boldsymbol{x_j}\right) + \sum_{k=1}^{l} \alpha_k,$$

dengan syarat,

$\alpha_i \geq 0$, i=1,2,...,l

$\sum_{j=1}^{l} \alpha_j y_j = 0.$

Hyperplane pemisah optimal diberikan oleh,

$$\boldsymbol{w} = \sum_{i=1}^{l} \alpha_i y_i \boldsymbol{x_i},$$

$$b = -\frac{1}{2}(\boldsymbol{w}.\boldsymbol{x}_r + \boldsymbol{x}_s),$$

dimana $\boldsymbol{x}_r$ dan $\boldsymbol{x}_s$ adalah jumlah *support vector* dari masing-masing kelas,

$\alpha_r, \alpha_s > 0, y_r = -1, y_s = 1.$

Support Vector (SV) merupakan data yang memiliki pengali Lagrange bukan nol. Semua SV akan terletak pada margin dan memungkinkan bahwa jumlah SV bisa sangat kecil. Jika data dapat dipisahkan secara linier, maka berlaku persamaan,

$$||\boldsymbol{w}||^2 = \sum_{i=1}^{l} \alpha_i = \sum_{i \epsilon SV} \alpha_i = \sum_{i \epsilon SV} \sum_{j \epsilon SV} \alpha_i \alpha_j y_i y_j \left(\boldsymbol{x_i}.\boldsymbol{x_j}\right). \quad (12)$$

Keterangan :

$\boldsymbol{w}$: vektor bobot

$\boldsymbol{x}$: vektor data training

b : nilai bias

$\boldsymbol{x_i}$: *support vector* ke-i

y_i : kelas ke-i

α_i : pengali lagrange ke-i

Klasifikasi linier SVM ketika terdapat data yang tidak dapat dikelompokan dengan benar (*nonseparable case*), rumusan SVM ditambah dengan adanya variabel *slack* (ξ) dengan tujuan untuk meminimalkan fungsi:

$$\phi(\boldsymbol{w}, \xi) = \frac{1}{2}||\boldsymbol{w}||^2 + C\sum_{i=1}^{l} \xi_i,$$

dimana

$\xi_i \geq 0$ (Gunn, 1998).

Variabel *slack* (ξ) digunakan untuk mengukur penyimpangan suatu titik dari kondisi ideal pola yang terpisah (Gorunescu, 2012). SVM menggunakan fungsi kernel untuk mengatasi masalah *non linier*. *Hyperplane* pemisah optimal di *feature space* diberikan oleh,

$$f(x) = sign(\sum_{\boldsymbol{i} \in \boldsymbol{SV}} \boldsymbol{\alpha_i} K(\boldsymbol{x_i}.\boldsymbol{x}) + b),$$

dimana $K(\boldsymbol{x}.\boldsymbol{x}')$ merupakan fungsi kernel yang memetakan masalah *non linier* ke *feature space* dengan syarat,

$0 \leq \alpha_i \leq C, i = 1, \ldots, l.$

Fungsi kernel yang digunakan yaitu:

a. *Polynomial*

$$K(\boldsymbol{x_i}.\boldsymbol{x}) = (\boldsymbol{x_i}.\boldsymbol{x} + 1)^d$$

b. *Radial Basis Function* (RBF)

$$K(\boldsymbol{x_i}.\boldsymbol{x}) = exp\,(-\gamma||\boldsymbol{x_i}.\boldsymbol{x}||^2)$$

Cross-Validation merupakan metode statistik yang mengevaluasi dan membandingkan algoritma pembelajaran dengan cara membagi data menjadi dua, satu data digunakan untuk melatih model dan data lainnya digunakan untuk memvalidasi model. Bentuk dasar *cross-*

validation adalah *k-fold cross-validation*. *K-fold cross-validation* dipartisi menjadi k bagian yang berukuran sama atau hampir sama. Sejumlah k bagian digunakan untuk proses validasi, sementara k-1 bagian digunakan untuk proses pelatihan (Hastie, 2017). Penelitian ini menggunakan 10-*fold cross-validation* pada metode *sampling* model SVM, dan digunakan untuk proses tuning *hyperparameter* menggunakan *grid search*.

Algoritma *grid-search* melatih semua kombinasi parameter dan biasanya diukur menggunakan *cross-validation*. *Cross-validation* memastikan bahwa model terlatih memperoleh sebagian besar pola dari dataset. *Grid-search* membangun partisi dari semua kombinasi parameter yang diberikan, kemudian menghitung skor setiap model untuk dievaluasi, dan memilih model yang memberikan hasil terbaik. Parameter terbaik yang diperoleh kemudian digunakan dalam model aktual. *Grid-search* memberikan jaminan deteksi *hyperparameter* terbaik, namun sangat lemah dalam konvergensi dan dimensi yang tinggi (Elgeldawi *et al*., 2021). Algoritma *grid-search* untuk mendapatkan global optimum (Mesafint, 2021) yaitu:

a. Mulailah dengan ruang yang luas untuk pencarian dan skala fase.
b. Berdasarkan hasil sebelumnya, *hyperparameter* yang berkinerja baik mempersempit ruang pencarian dan ukuran fase.
c. Ulangi langkah b beberapa kali sampai nilai optimal tercapai.

Pengukuran kinerja klasifikasi menggunakan tabel *confusion matrix*. Tabel *confusion matrix* adalah alat untuk menganalisis seberapa baik pengklasifikasi dapat mengenali kelas dari data yang berbeda (Han *et al*., 2011).

Tabel 1. *Confusion matrix*

		Kelas Hasil Prediksi (j)	
		Kelas 0	Kelas 1
Kelas Asli (i)	Kelas 0	TP	FN
	Kelas 1	FP	TN

Dimana:
True Positive (TP): jumlah kelas ‘ya’ yang diprediksi sebagai ‘ya’.
True Negative (TN): jumlah kelas ‘tidak’ yang diprediksi sebagai ‘tidak’.
False Positive (FP): jumlah kelas ‘tidak’ yang diprediksi sebagai ‘ya’.
False Negative (FN): jumlah kelas ‘ya’ yang diprediksi sebagai ‘tidak’.

Akurasi merupakan jumlah data uji yang diklasifikasikan secara benar oleh pengklasifikasi (Han *et al*., 2011). Nilai akurasi dapat dicari dengan persamaan berikut:

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

Error Rate merupakan nilai yang gagal diprediksi oleh sistem sehingga hasil analisis berupa jumlah data yang tidak dapat diklasifikasikan secara benar dari nilai aktual. Nilai *error rate* dapat dicari dengan persamaan berikut:

$$\textit{Error rate} = \frac{\text{FP+FN}}{\text{TP+TN+FP+FN}}$$

Sensitivity/recall merupakan tingkat positif yang sebenarnya, yaitu proporsi dari kelas positif yang dapat diidentifikasi secara benar (Han *et al*., 2011). Nilai *sensitivity/recall* dapat dicari dengan persamaan berikut:

$$\textit{Sensitivity/Recall} = \frac{\text{TP}}{\text{TP+FN}}$$

Specificity merupakan tingkat negatif yang sebenarnya, yaitu proporsi dari kelas negatif yang dapat diidentifikasi secara benar (Han *et al*., 2011). Nilai *specivicity* dapat dicari dengan persamaan berikut:

$$\textit{Specivicity} = \frac{\text{TN}}{\text{TN+FP}}$$

Presisi dapat dianggap sebagai ukuran ketepatan, yaitu jumlah persentase kelas yang diberi label positif yang sebenarnya (Han *et al*., 2011). Nilai presisi dapat dicari dengan persamaan berikut:

$$\text{Presisi} = \frac{\text{TP}}{\text{TP+FP}}$$

3. METODE PENELITIAN

Data yang digunakan pada penelitian ini adalah data sekunder. Data sekunder adalah data yang diperoleh secara tidak langsung melalui media perantara (dihasilkan oleh pihak lain) atau digunakan oleh lembaga lain yang bukan merupakan pengelolanya tetapi dapat dimanfaatkan oleh penelitian tertentu. Dalam hal ini data sekunder berasal dari BPS Kabupaten Pekalongan.

Penelitian ini akan mengaplikasikan metode klasifikasi KNN dan SVM. Berikut merupakan tahapan analisis data menggunakan metode KNN:

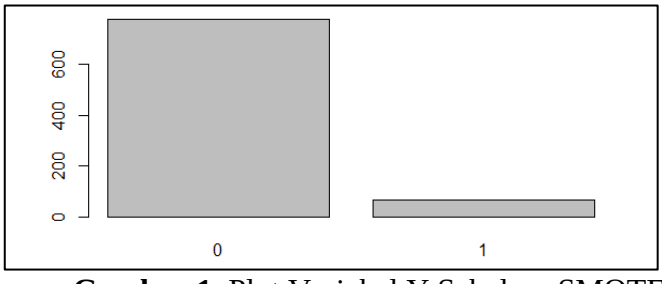
a. Membagi data *training* dan data *testing* dengan proporsi 85%:15%.
b. Menentukan nilai *k* terbaik dengan optimasi parameter menggunakan *grid search*.
c. Menghitung jarak *euclidean* antara data *training* dengan data *testing*.
d. Mengurutkan jarak *euclidean* dari yang terkecil sampai terbesar.
e. Menentukan jarak tetangga terdekat sebanyak nilai *k* yang ditentukan.
f. Menetapkan kelas mayoritas dari *k* sebagai kelas data *testing*.
g. Evaluasi hasil klasifikasi dengan metode KNN pada data testing untuk mengukur ketepatan klasifikasi.

Tahapan analisis data menggunakan metode SVM adalah sebagai berikut:

a. Membagi data *training* dan data *testing* dengan proporsi 85%:15%.
b. Menentukan fungsi kernel dan nilai-nilai parameter kernel untuk optimasi *hyperplane*.
c. Menentukan nilai parameter terbaik menggunakan *grid search*.
d. Menentukan *hyperplane* dengan menggunakan parameter terbaik.
e. Menggunakan *hyperplane* dengan parameter terbaik yang diperoleh untuk setiap fungsi kernel pada klasifikasi data testing.
f. Evaluasi hasil klasifikasi dengan metode SVM pada data testing untuk mengukur ketepatan klasifikasi..

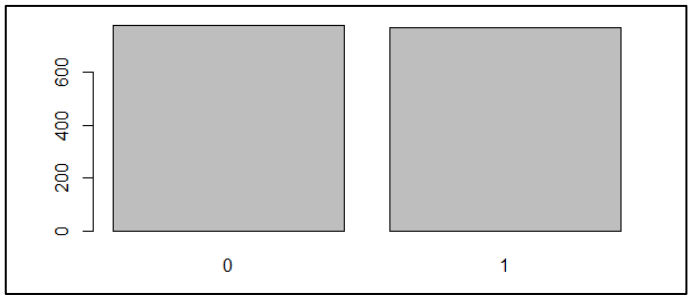
4. HASIL DAN PEMBAHASAN

Karakteristik data SUSENAS dapat dilihat dari banyaknya jumlah pengamatan dan pola persebaran data tiap variabel. Variabel Y menunjukkan status penerimaan bantuan dari pemerintah. Nilai “0” merupakan jumlah keluarga yang tidak menerima bantuan dari pemerintah dan nilai “1” merupakan jumlah keluarga yang menerima bantuan dari pemerintah.



Gambar 1. Plot Variabel Y Sebelum SMOTE

Pada gambar 1 terlihat bahwa jumlah keluarga yang tidak menerima bantuan dari pemerintah adalah 775 dan jumlah keluarga yang menerima bantuan dari pemerintah adalah 64. Selisih antara 2 kelas sangat jauh atau data yang digunakan merupakan *class imbalance*. Data kemudian ditambah data baru menggunakan algoritma SMOTE untuk menangani *class imbalance*.



Gambar 2. Plot Variabel Y Setelah SMOTE

Pada gambar 2 terlihat bahwa jumlah keluarga yang tidak menerima bantuan dari pemerintah adalah 775 dan jumlah keluarga yang menerima bantuan dari pemerintah adalah 768. Selisih antar 2 kelas sudah seimbang sehingga dapat dilakukan proses klasifikasi.

Algoritma KNN menggunakan parameter jarak sebagai ukuran kedekatan data dengan tetangga terdekatnya. Jarak yang digunakan pada penelitian ini adalah jarak *euclidean.* Selain parameter jarak, nilai k yang merupakan jumlah tetangga terdekat akan ditentukan oleh peneliti. Nilai k paling sedikit adalah 1 dan nilai k paling besar adalah hasil akar kuadrat dari jumlah data *training* (Hassanat *et al*., 2014). Penelitian menggunakan 1312 data *training* yang merupakan hasil dari 85% jumlah seluruh *record* setelah dilakukan SMOTE. Nilai k paling besar adalah akar kuadrat dari 1312 yaitu 35, sehingga penelitian ini menggunakan nilai k=1,3,…,35 untuk mendapatkan nilai k terbaik. Nilai k ganjil digunakan untuk menghindari kemungkinan jumlah tetangga terdekat masing-masing kelas bernilai sama (Hafiz, 2021).

Nilai k terbaik ditentukan berdasarkan nilai akurasi yang paling besar. Proses dilakukan dengan *hyperparameter tuning* menggunakan algoritma *grid search*. Hasil optimasi parameter menggunakan program R Studio memiliki akurasi paling besar adalah k=1 dengan nilai akurasi sebesar 0.8458797.

Tabel 2. *Confusion Matrix* Metode KNN

		Kelas Hasil Prediksi	
		Kelas 0 (Bukan Penerima Bantuan)	Kelas 1 (Penerima Bantuan)
Kelas Asli	Kelas 0 (Bukan Penerima Bantuan)	99	29
	Kelas 1 (Penerima Bantuan)	15	88

Berdasarkan Tabel 2 dapat dihitung ketepatan kinerja metode KNN pada klasifikasi data SUSENAS. Berikut perhitungan akurasi, *error rate*, *sensitivity*, *specivicity* dan presisi.

Akurasi = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{99+88}{99+88+15+29} = \frac{187}{231} = 0{,}809524$

Error rate = $\frac{FP+FN}{TP+TN+FP+FN} = \frac{15+29}{99+88+15+29} = \frac{187}{231} = 0{,}190476$

Sensitivity/recall = $\frac{TP}{TP+FN} = \frac{99}{99+29} = \frac{99}{128} = 0{,}773438$

Specivicity = $\frac{TN}{TN+FP} = \frac{88}{88+15} = \frac{88}{103} = 0{,}854369$

Presisi = $\frac{TP}{TP+FP} = \frac{99}{99+15} = \frac{99}{114} = 0{,}868421$

Klasifikasi metode KNN memiliki tingkat akurasi yang cukup tinggi yaitu 0,809524 yang berarti jumlah data uji yang diklasifikasikan secara benar oleh pengklasifikasi bernilai 0,809524. Nilai *error rate* sebesar 0,190476 artinya nilai sebesar 0,190476 gagal diprediksi oleh sistem. Nilai *sensitivity/recall* sebesar 0,773438 artinya proporsi dari kelas positif yang dapat diidentifikasi secara benar bernilai 0,773438. Nilai *specivicity* sebesar 0,854369 artinya proporsi dari kelas negatif yang dapat diidentifikasi secara benar bernilai 0,854369. Nilai presisi sebesar 0,868421 artinya jumlah persentase kelas yang diberi label positif yang sebenarnya memiliki nilai sebesar 0,868421.

Klasifikasi SVM dengan fungsi kernel polynomial menggunakan parameter d (*degree*) dan C (*cost*). Nilai parameter d yang dicoba yaitu $\mathrm{d} = 1,2,3,4,5$ dan nilai parameter C yang dicoba yaitu $\mathrm{C} = 0{,}001; 0{,}01; 0{,}1; 1; 10$. Seluruh kombinasi parameter d dan C dicoba untuk mendapatkan kombinasi parameter terbaik yang akan digunakan pada proses klasifikasi. Kombinasi parameter terbaik ditentukan berdasarkan nilai *error rate* terkecil. Proses dilakukan menggunakan *grid search* pada proses tuning *hyperparameter* dan metode *sampling* 10-*fold cross-validation* untuk mendapatkan kombinasi parameter terbaik. Nilai *error rate* terkecil didapatkan pada nilai $\mathrm{d} = 4$ dan nilai $C = 10$ dengan nilai bias sebesar -0.5553801 dan nilai error rate sebesar 0.2553956 (lampiran 3). Fungsi hyperplane kernel polynomial yaitu :

$f(x) = sign(\sum_{i \in SV} \boldsymbol{\alpha_i} K(\boldsymbol{x_i}. \boldsymbol{x}) - 0.5553801)$

Klasifikasi SVM dengan fungsi kernel *Radial Basis Function* (RBF) menggunakan parameter γ (*gamma*) dan C (*cost*). Nilai parameter γ yang dicoba yaitu $\gamma = 0{,}001; 0{,}01; 0{,}1; 1; 10$ dan nilai parameter C yang dicoba yaitu $\mathrm{C} = 0{,}001; 0{,}01; 0{,}1; 1; 10$. Seluruh kombinasi parameter γ dan C dicoba untuk mendapatkan kombinasi parameter terbaik yang akan digunakan pada proses klasifikasi. Kombinasi parameter terbaik ditentukan berdasarkan nilai *error rate* terkecil. Proses dilakukan menggunakan *grid search* pada proses tuning *hyperparameter* dan metode *sampling* 10-*fold cross-validation* untuk mendapatkan kombinasi parameter terbaik. Nilai *error rate* terkecil didapatkan pada nilai $\gamma = 1$ dan nilai $C = 10$ dengan nilai bias sebesar 0.4870422 dan nilai *error rate* sebesar 0.1577724 (lampiran 3). Fungsi *hyperplane* kernel RBF yaitu :

$f(x) = sign(\sum_{i \in SV} \boldsymbol{\alpha_i} K(\boldsymbol{x_i}. \boldsymbol{x}) + 0.4870422)$.

Hasil klasifikasi menggunakan kernel RBF karena memiliki nilai *error rate* yang lebih kecil dari nilai *error rate* kernel polynomial, sehingga kernel RBF dengan nilai γ= 1 dan nilai C= 10 digunakan untuk klasifikasi.

Tabel 3. *Confusion Matriks* Metode SVM

		Kelas Hasil Prediksi	
		Kelas 0 (Bukan Penerima Bantuan)	Kelas 1 (Penerima Bantuan)
Kelas Asli	Kelas 0 (Bukan Penerima Bantuan)	91	37
	Kelas 1 (Penerima Bantuan)	12	91

Berdasarkan Tabel 3 dapat dihitung ketepatan kinerja metode SVM pada klasifikasi data SUSENAS. Berikut perhitungan akurasi, *error rate*, *sensitivity*, *specivicity* dan presisi.

Accuracy = $\frac{TP+TN}{TP+TN+FP+FN} = \frac{91+91}{91+91+12+37} = \frac{182}{231} = 0{,}787879$

Error rate = $\frac{FP+FN}{TP+TN+FP+FN} = \frac{12+37}{91+91+12+37} = \frac{49}{231} = 0{,}212122$

Sensitivity/recall = $\frac{TP}{TP+FN} = \frac{91}{91+37} = \frac{14}{16} = 0{,}710938$

Specivicity = $\frac{TN}{TN+FP} = \frac{91}{91+12} = \frac{91}{103} = 0{,}883495$

Precision = $\frac{TP}{TP+FP} = \frac{91}{91+12} = \frac{91}{103} = 0{,}883495$

Klasifikasi metode SVM memiliki tingkat akurasi yang tinggi yaitu 0,787879 yang berarti jumlah data uji yang diklasifikasikan secara benar oleh pengklasifikasi bernilai 0,787879. Nilai error rate sebesar 0,212122 artinya nilai sebesar 0,212122 gagal diprediksi oleh sistem. Nilai sensitivity/recall sebesar 0,710938 artinya proporsi dari kelas positif yang dapat diidentifikasi secara benar bernilai 0,710938. Nilai specivicity sebesar 0,883495 artinya proporsi dari kelas negatif yang dapat diidentifikasi secara benar bernilai 0,883495. Nilai presisi sebesar 0,883495 artinya jumlah persentase kelas yang diberi label positif yang sebenarnya memiliki nilai sebesar 0,883495.

Hasil pengujian antara metode KNN dan SVM kemudian dibandingkan untuk mendapatkan metode terbaik dalam mengklasifikasikan data SUSENAS.

Tabel 4. Perbandingan Kinerja Metode KNN dan SVM

	KNN	SVM
Akurasi	0,809524	0.787879
Error rate	0,190476	0.212122
Sensitivity	0,773438	0.710938
Specivicity	0,854369	0,883495
Presisi	0,868421	0,883495

Pada Tabel 4 dapat diketahui bahwa metode KNN memiliki tingkat akurasi sebesar 0,809524 lebih tinggi dari tingkat akurasi metode SVM sebesar 0.787879. Nilai *error rate* metode KNN sebesar 0,190476 lebih kecil dari nilai *error rate* metode SVM sebesar 0.212122. Nilai *sensitivity* metode KNN sebesar 0,773438 lebih tinggi dari nilai *sensitifity* metode SVM sebesar 0,710938. Nilai *specivicity* metode KNN bernilai 0,854369 lebih kecil dari nilai *specivicity* metode SVM sebesar 0,883495. Nilai presisi metode KNN sebesar 0,868421 lebih kecil dari nilai presisi metode SVM sebesar 0,883495. Berdasarkan uraian sebelumnya, maka metode klasifikasi yang paling optimal untuk mengklasifikasikan data SUSENAS adalah metode KNN, karena metode KNN memiliki kinerja klasifikasi yang lebih unggul dibandingkan metode SVM.

5. KESIMPULAN

Berdasarkan hasil analisis yang telah dilakukan, klasifikasi data SUSENAS dengan proporsi data *training* sebesar 85% dan proporsi data *testing* sebesar 15%, dengan membandingkan metode KNN dan SVM, didapatkan hasil bahwa metode KNN memiliki tingkat akurasi sebesar 0,809524 lebih tinggi dari tingkat akurasi metode SVM sebesar 0.787879. Nilai *error rate* metode KNN sebesar 0,190476 lebih kecil dari nilai *error rate* metode SVM sebesar 0.212122. Nilai *sensitivity* metode KNN sebesar 0,773438 lebih tinggi dari nilai *sensitifity* metode SVM sebesar 0,710938. Nilai *specivicity* metode KNN bernilai 0,854369 lebih kecil dari nilai *specivicity* metode SVM sebesar 0,883495. Nilai presisi metode KNN sebesar 0,868421 lebih kecil dari nilai presisi metode SVM sebesar 0,883495. Metode klasifikasi yang paling optimal dari penelitian ini adalah metode KNN, karena metode KNN memiliki kinerja klasifikasi yang lebih unggul dibandingkan metode SVM.

DAFTAR PUSTAKA

Aulia, S., Hadiyoso, S., Ramadan, D. N. 2015. *Analisis Perbandingan KNN dengan SVM untuk Klasifikasi Penyakit Diabetes Retinopati berdasarkan Citra Eksudat dan Mikroaneurisma*. Jurnal ELKOMIKA, No. 1, Vol. 3.

Badan Pusat Statistik. 2022. BPS, Jakarta. https://www.bps.go.id/index.php/subjek/81 (diakses pada 31 Oktober 2022).

Cover, T. M., dan Hart, P. E. 1967. *Nearest Neighbor Pattern Classification*. IEEE Transactions on Information Theory, Vol. IT-13, No. 1.

Elgeldawi, E., Sayed, A., Galal, A. R. & Zaki, A. M. 2021. *Hyperparameter Tuning for Machine Learning Algorithms Used for Arabic Sentiment Analysis*. Informatics 2021, 8, 79. https://doi.org/10.3390/informatics8040079.

Gorunescu, F. 2011. *Data Mining: Concepts, Models and Techniques Vol. 12*. Berlin, Heidelberg: Springer Berlin Heidelberg.

Gunn, S. R. 1998. *Support Vector Machines for Classification and Regression*. Southampton: Department of Electronics and Computer Science University of Southampton.

Han, J., Kamber, M., & Pei, J. 2011. *Data Mining: Concepts and Techniques* (3rd ed.). Elsevier. https://doi.org/10.1016/B978-0-12-381479-1.00001-0

Hafiz, M. I. 2021. *Implementasi Algoritma K-Nearest Neighbor* (KNN) *Dalam Penentuan Jenis Kucing*. Palangkaraya: Program Studi Teknik Informatika, Sekolah Tinggi Manajemen Informatika dan Komputer [Skripsi].

Hassanat, A. B., Abbadi, M. A., Altarawneh, G. A. 2014. *Solving the Problem of the K Parameter in the KNN Classifier Using an Ensemble Learning Approach*. International Journal of Computer Science and Information Security (IJCSIS), Vol. 12, No. 8.

Hastie, T., Tibshirani, R., Friedman, J. 2017. *The Elements of Statistical Learning*. Springer Series in Statistics.

Mesafint, D. & Mnjaiah, D. H. 2021. *Grid search in hyperparameter optimization of machine learning models for prediction of HIV/AIDS test results*. International Journal of Computers and Applications 44(1):1-12.

Purnama, N. A. 2022. *Bansos Tidak Tepat Sasaran Adalah Maladministrasi*. https://ombudsman.go.id/perwakilan/news/r/pwkinternal--bansos-tidak-tepat-sasaran-adalah-maladministrasi (diakses pada 31 Oktober 2022).