

# Klasifikasi Nyeri Punggung Bawah Menggunakan Algoritma *K-Nearest Neighbor* Dan *Support Vector Machine*

Nadiyah Fadhilah<sup>1</sup>, Anita Desiani<sup>1\*</sup>

<sup>1</sup>Matematika, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas Sriwijaya  
Jl. Raya Palembang – Prabumulih KM.32 Indralaya Indah, Kabupaten Ogan Ilir, Sumatera Selatan, Indonesia  
E-mail: [anita\\_desiani@unsri.ac.id](mailto:anita_desiani@unsri.ac.id)

Naskah Masuk: 24 Juni 2024 ; Diterima: 01 November 2024; Terbit: 31 Maret 2025

---

## ABSTRAK

---

**Abstrak** - Nyeri Punggung Bawah merupakan salah satu masalah kesehatan yang bisa menyebabkan terhambatnya aktivitas sehari-hari. Nyeri Punggung Bawah tidak menyebabkan kematian, tetapi menyebabkan seseorang yang mengalaminya menjadi tidak produktif dalam melakukan pekerjaan. Penelitian ini bertujuan untuk mengklasifikasi Nyeri Punggung Bawah seseorang apakah dalam kondisi abnormal atau normal dengan menggunakan algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) dengan metode training persentase split dan *K-Fold Cross Validation*. Pada metode KNN percobaan dilakukan sebanyak 12 nilai k dan mata metode SVM kernel yang digunakan yaitu kernel linier. Adapun atribut yang digunakan yaitu, Pelvic Incidence, Pelvic Tilt, Lumbar Lordosis Angle, Sacral Slope, Pelvic Radius, Degree Spondylolisthesis, Pelvic Slope, Direct Tilt, Thoracic Slope, Cervical Tilt, Sacrum Angle, Scoliosis Slope, dan class. Hasil pengolahan data dengan algoritma *K-Nearest Neighbor* (KNN) yaitu akurasi sebesar 87,1 %, presisi abnormal 87% dan normal 88%, serta recall abnormal 83% dan normal 87%, sedangkan dengan menggunakan algoritma *Support Vector Machine* (SVM) akurasi sebesar 83,8%, presisi abnormal 86% dan normal 88%, serta recall abnormal 92% dan normal 72%. Hal ini menunjukkan bahwa model dan atribut yang digunakan baik dalam mengklasifikasi Nyeri Punggung Bawah.

**Kata kunci:** Klasifikasi, *K-Nearest Neighbor*, Nyeri Punggung Bawah, *Support Vector Machine*

---

## ABSTRACT

---

**Abstract** - Lower back pain is one of the health problems that can hinder daily activities. Lower back pain does not cause death, but causes someone who experiences it to be unproductive in doing work. This study aims to classify a person's Low Back Pain whether in abnormal or normal conditions using the *K-Nearest Neighbor* (KNN) and *Support Vector Machine* (SVM) algorithms with the split percentage training method and *K-Fold Cross Validation*. The attributes used are, Pelvic Incidence, Pelvic Tilt, Lumbar Lordosis Angle, Sacral Slope, Pelvic Radius, Degree Spondylolisthesis, Pelvic Slope, Direct Tilt, Thoracic Slope, Cervical Tilt, Sacrum Angle, Scoliosis Slope, and class. The results of data processing with the *K-Nearest Neighbor* (KNN) algorithm are 87.1% accuracy, 87% abnormal precision and 88% normal, and 83% abnormal recall and 87% normal, while using the *Support Vector Machine* (SVM) algorithm the accuracy by 83.8%, 86% abnormal precision and 88% normal, and 92% abnormal recall and 72% normal. This shows that the models and attributes used are good in classifying Low Back Pain.

**Keywords:** Classification, *K-Nearest Neighbor*, Lower Back Pain, *Support Vector Machine*

Copyright © 2025 Jurnal Teknik Elektro dan Komputasi (ELKOM)

---

## 1. PENDAHULUAN

Nyeri Punggung Bawah (NPB) ialah gangguan musculoskeletal yang salah satunya disebabkan oleh kegiatan tubuh yang kurang baik. Tidak hanya itu NPB bisa pula diakibatkan oleh bermacam penyakit musculoskeletal, gangguan psikologis serta aktivasi yang tidak benar. [1]. NPB merupakan salah satu masalah kesehatan yang sangat umum, yang menyebabkan banyak orang kehilangan gaya hidup normal dan terhambatnya aktivitas sehari-hari [2]. Berdasarkan Hasil Riset Kesehatan Dasar (2018), prevalensi penyakit musculoskeletal di Indonesia yang pernah di diagnosis oleh tenaga kesehatan ialah 11,9% serta bersumber pada diagnosis ataupun gejala ialah 24,7%. Jumlah pengidap nyeri punggung bawah di Indonesia tidak diketahui pasti, tetapi diperkirakan antara 7,6% hingga 37%. Salah satu cara mendiagnosis [3]. NPB bisa dilakukan dengan melakukan penukuran menggunakan skala numerik dengan memberikan angket mengani faktor fisik sert faktor psikologis, dalam teknik pembelajaran mesin pada data tersebut bisa

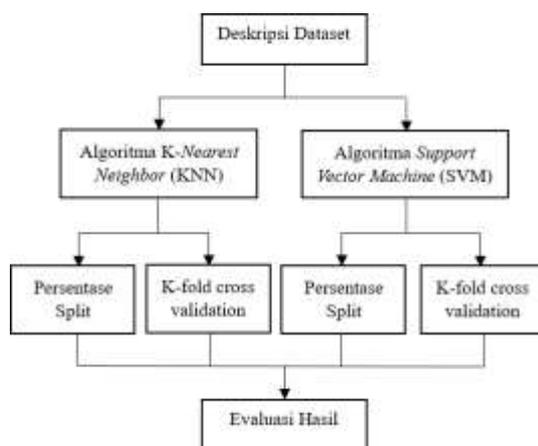
mendapatkan hubungan antara pengukuran ini yang bisa membantu dalam diagnosis serta klasifikasi NPB yang benar [2].

Klasifikasi merupakan suatu proses untuk mengenali dan mengelompokkan objek-objek yang memiliki karakteristik serupa ke dalam kategori yang telah ditetapkan sebelumnya [4]. Pengklasifikasian NPB sebelumnya telah dilakukan oleh [5] menggunakan algoritma Decision Tree, namun penelitian ini menghasilkan nilai akurasi dibawah 80%. Algoritma Decision Tree rentan terhadap overfitting, terutama jika memiliki kedalaman yang berlebihan [6]. Algoritma klasifikasi lainnya yang menggunakan regularisasi yang dapat membantu mencegah overfitting yaitu Support Vector Machine (SVM). Algoritma Support Vector Machine (SVM) memiliki kelebihan menyimpan sebagian kecil data latih yang digunakan saat prediksi, karena tidak semua data latih akan dilibatkan saat pelatihan [7]. Algoritma SVM berlandaskan teori pembelajaran statistik yang memberi hasil lebih baik dari metode lain, selain itu SVM bekerja dengan baik pada data berdimensi tinggi dengan menggunakan teknik kernel [8]. Algoritma SVM memiliki kelemahan yaitu sulit dipakai untuk jumlah sampel yang berskala besar, serta SVM dikembangkan untuk masalah klasifikasi dengan dua kelas [9]. Selain itu algoritma lainnya seperti K-Nearest Neighbor menggunakan parameter k yang dapat disesuaikan, sehingga mengurangi sensitivitas terhadap data pelatihan dan memberikan hasil klasifikasi yang lebih robust.

Pada penelitian ini akan mengklasifikasi Lower Back Pain Symptoms Dataset menggunakan algoritma K-Nearest Neighbor (KNN) dan Support Vector Machine (SVM) untuk menentukan apakah nyeri punggung bawah seseorang abnormal atau normal. Terdapat 13 atribut dalam dataset ini yang dimana 12 sebagai atribut fitur dan 1 sebagai atribut label. Atribut-atribut tersebut merupakan parameter atau nilai dari kondisi seseorang yang akan digunakan dalam proses klasifikasi.

**2. METODE PENELITIAN**

Secara umum, metodologi penelitian ini dimulai dari deskripsi dataset, dilanjutkan dengan pengolahan datas, analisis data, hingga evaluasi hasil. Tahapan-tahapan tersebut dapat dilihat pada Gambar 1.



Gambar 1. Tahapan metodologi penelitian

**2.1. Deskripsi Data**

Dataset dalam penelitian ini diambil dari situs Kaggle (<https://www.kaggle.com/datasets/sammy123/lower-back-pain-symptoms-dataset>). Dataset terdiri dari 310 data, sebanyak 210 data abnormal dan 100 data normal, serta terdiri dari 13 atribut dengan 1 label atau kelas dan 12 atribut lainnya. Atribut dan informasi data yang digunakan pada artikel ini disajikan pada Tabel 1.

Tabel 1. Atribut pada dataset

<b>Nama Atribut</b>	<b>Keterangan</b>	<b>Tipe Data</b>
<i>Pelvic Incidence</i>	Parameter dari insiden panggul	Numerik
<i>Pelvic Tilt</i>	Parameter dari kemiringan panggul	Numerik
<i>Lumbar Lordosis Angle</i>	Parameter dari sudut lumbar lordosis	Numerik
<i>Sacral Slope</i>	Parameter dari kemiringan sakral	Numerik
<i>Pelvic Radius</i>	Parameter dari radius panggul	Numerik
<i>Degree Spondylolisthesis</i>	Parameter dari <i>Spondylolisthesis</i>	Numerik
<i>Pelvic Slope</i>	Parameter dari <i>Pelvic Slope</i>	Numerik
<i>Direct Tilt</i>	Parameter dari <i>Direct Tilt</i>	Numerik

<b>Nama Atribut</b>	<b>Keterangan</b>	<b>Tipe Data</b>
<i>Thoracic Slope</i>	Parameter dari <i>Thoracic Slope</i>	Numerik
<i>Cervical Tilt</i>	Parameter dari <i>Cervical Tilt</i>	Numerik
<i>Sacrum Angle</i>	Parameter dari <i>Sacrum Angle</i>	Numerik
<i>Scoliosis Slope</i>	Parameter dari <i>Scoliosis Slope</i>	Numerik
<i>Class</i>	Kategori klasifikasi abnormal dan normal	Kategori

**2.2. Metode Training**

a. Persentase Split

Persentase split merupakan teknik membagi data menjadi 2 bagian yaitu data training dan data testing. Adapun persentase split yang digunakan pada penelitian ini yaitu 80%, artinya 80% dari data untuk menjadi data training set kemudian sisa 20% menjadi data test set, pembagian dataset ini dilakukan secara acak dengan random seed sebesar 4.

b. K-Fold Cross Validation

K-fold cross validation merupakan teknik validasi untuk menilai bagaimana hasil analisis statistik mengeneralisasi kumpulan data. Nilai k yang digunakan pada penelitian ini yaitu 1-12.

**2.3. Pemodelan Klasifikasi**

a. *K-Nearest Neighbor* (KNN)

Algoritma *K-Nearest Neighbor* (KNN) diklasifikasi berdasarkan hasil query instance dari kategori algoritma KNN, dimana kelas hasil klasifikasi didasarkan oleh kelas yang paling banyak muncul [10]. Untuk hasil prediksi algoritma KNN bergantung pada jarak antara data uji yang baru dengan data latih yang ada, jauh atau dekatnya dapat dihitung dengan jarak Euclidian [11]. Langkah-langkah untuk menghitung algoritma KNN sebagai berikut [12].

1. Tentukan banyaknya tetangga k.
2. Menentukan jarak Euclid masing-masing objek terhadap training data yang diberikan menggunakan persamaan 1 [13].

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \tag{1}$$

Keterangan

- d : jarak kedekatan
- n : jumlah dimensi
- $x_i$  : nilai ke-i pada data x
- $y_i$  : nilai ke-i pada data y
- i : indeks

3. Lalu mengurutkan objek tersebut dari nilai tertinggi ke rendah.
4. Mengumpulkan label kelas Y (Klasifikasi *K-Nearest Neighbor*).
5. Dengan menggunakan kategori *K-Nearest Neighbor* yang paling mayoritas maka dapat diprediksi nilai query instance yang telah dihitung.

b. *Support Vector Machine* (SVM)

*Support Vector Machine* (SVM) adalah salah satu metode klasifikasi yang menggunakan metode *machine learning* untuk mengklasifikasi kelas berdasarkan pola dari hasil training yang diciptakan oleh Vladimir Vapnik [14]. SVM digunakan untuk menemukan hyperlane sehingga dapat memisahkan ke dalam dua kelas yang berbeda serta memaksimalkan margin antara dua kelas tersebut [15].

Asumsikan kedua kelas -1 dan +1 dapat terpisah secara sempurna oleh *hyperplane* berdimensi d, yang didefinisikan pada persamaan 2 [16].

$$w \cdot x_i + b = 0 \tag{2}$$

Pattern i yang termasuk kelas -1 memenuhi pertidaksamaan (3):

$$w \cdot x_i + b \leq -1 \tag{3}$$

Sedangkan pattern i yang termasuk kelas +1 memenuhi pertidaksamaan (4):

$$w \cdot x_i + b \geq 1 \tag{4}$$

Langkah-langkah dalam algoritma SVM yaitu [17]:

1. Menentukan titik data ( $x_1, x_2, \dots, x_n$ ) yang merupakan atribut pada data.
2. Menentukan kelas data  $y_i \in \{-1, +1\}$  yang dimana -1 sebagai abnormal dan 1 sebagai normal.
3. Tentukan data kelas berdasarkan persamaan (5).

$$\{(x_i, y_i)\}_{i=1}^N \tag{5}$$

Dimana N merupakan banyaknya data

4. Memaksimalkan fungsi berdasarkan persamaan (6).

$$Ld = \sum_{i=1}^N d_i - \sum_{i=1}^N \sum_{j=1}^N k_i k_j y_i y_j K(x_i, y_i) \tag{6}$$

\*syarat  $0 \leq k_i \leq C$  dan  $\sum_{i=1}^N k_i y_i = 0$

Keterangan

Ld : daulitas Langrange Multiplier

n : nilai bobot setiap titik data

C : nilai konstanta

- Menghitung nilai w dan b berdasarkan persamaan (7) dan (8).

$$w = \sum_{i=1}^N a_i y_i \cdot x_i \tag{7}$$

$$b = \frac{-1}{2} (w \cdot x^i + w \cdot x) \tag{8}$$

- Menghitung fungsi keputusan klasifikasi *sign* (f (x)) dengan persamaan (9).

$$f(x) = w \cdot x + b \tag{9}$$

atau dengan persamaan (10)

$$f(x) = \sum_{i=1}^m a_i y_i K(x, x_i) + b \tag{10}$$

Keterangan

m : jumlah titik data yang dimiliki  $a_i > 0$

$K(x, x_i)$  : fungsi kernel

Pada penelitian ini kernel yang digunakan yaitu kernel linier. Adapun pendefinisian kernel linier yaitu pada persamaan (11) [18].

$$K(x, y) = x \cdot y \tag{11}$$

#### 2.4. Evaluasi Hasil

Dalam penelitian ini, perhitungan evaluasi meliputi nilai akurasi, presisi, dan recall berdasarkan hasil dari uji klasifikasi KNN dan SVM yang bertujuan untuk mengevaluasi keakuratan metode yang diusulkan sehingga diterapkan *Confusion Matrix*. *Confusion Matrix* merupakan sebuah matriks yang menampilkan performa dari sebuah algoritma secara spesifik, setiap baris matrix tersebut akan merepresentasikan kelas aktual dari data dan setiap kolom merepresentasikan kelas prediksi dari data [19]. *Confusion matrix* dapat dilihat pada Tabel 2.

Tabel 2. *Confusion matrix*

		Prediction	
		Negative	Positive
Actual	Negative	True Negative (TN)	False Positive (FP)
	Positive	False Negative (FN)	True Positive (TP)

Dengan *Confusion matrix* dapat diperoleh data lain yang digunakan untuk mengukur performa dari sebuah model yang digunakan, yaitu [19]:

- Akurasi, yaitu total dari keseluruhan seberapa sering model benar mengklasifikasi, akurasi didefinisikan dengan persamaan (12).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{12}$$

- Presisi didefinisikan dengan persamaan (13).

$$Precision = \frac{TP}{TP+FP} \tag{13}$$

- Recall atau sensitivity, yaitu Ketika nilai kelas actualny positif, akan dihitung seberapa sering model memprediksi positif, recall didefinisikan dengan persamaan (14).

$$Recall = \frac{TP}{TP+FN} \tag{14}$$

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Algoritma K-Nearest Neighbor (KNN)

Pada algoritma K-Nearest Neighbor (KNN) metode training yang digunakan yaitu persentase split dan K-Fold Cross Validation, dengan percobaan dilakukan sebanyak 12 nilai k, yaitu 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12. Sehingga didapatkan nilai presisi, recall, dan akurasi seperti pada Tabel 3.

Tabel 3. Nilai Presisi, Recall, dan Akurasi Algoritma KNN

K	Target Class	Percentage Split			K	Target Class	Percentage Split		
		Presisi	Recall	Akurasi			Presisi	Recall	Akurasi
1	Abnormal	85%	85%	80,6%	1	Abnormal	86%	81%	78%
	Normal	71%	71%			Normal	64%	72%	
2	Abnormal	80%	98%	82,2%	2	Abnormal	82%	92%	81,3%
	Normal	92%	52%			Normal	78%	58%	
3	Abnormal	80%	88%	77,4%	3	Abnormal	89%	82%	81,3%
	Normal	71%	57%			Normal	68%	79%	
6	Abnormal	82%	88%	79%	6	Abnormal	85%	86%	79,7%
	Normal	72%	62%			Normal	69%	67%	
7	Abnormal	85%	83%	79%	7	Abnormal	87%	83%	80,0%
	Normal	68%	71%			Normal	67%	74%	
8	Abnormal	84%	93%	83,8%	8	Abnormal	85%	86%	79,6%
	Normal	82%	67%			Normal	69%	67%	
9	Abnormal	89%	80%	80,6%	9	Abnormal	87%	81%	79%
	Normal	68%	81%			Normal	65%	75%	
10	Abnormal	87%	95%	87,1%	10	Abnormal	84%	83%	77,4%
	Normal	88%	71%			Normal	65%	66%	
11	Abnormal	92%	80%	82,2%	11	Abnormal	87%	80%	78,4%
	Normal	69%	86%			Normal	64%	74%	
12	Abnormal	90%	85%	83,8%	12	Abnormal	86%	82%	78,4%
	Normal	74%	81%			Normal	65%	71%	

Dari Tabel 3. dapat dilihat bahwa nilai keakuratan tertinggi hasil klasifikasi yang ditinjau dari parameter k=1 sampai k=12, untuk metode training persentase split adalah 87,1%, yaitu pada parameter k=10, dengan nilai presisi untuk abnormal yaitu 87% dan normal 88%, sedangkan untuk nilai recall abnormal yaitu 83% dan normal 87%. *Confusion matrix* dari algoritma KNN dengan metode training persentase split dapat dilihat pada Tabel 4.

Tabel 4. *Confusion matrix* persentase split

Kelas		Prediksi Sebenarnya	
		Abnormal	Normal
Label	Abnormal	39	2
Prediksi	Normal	6	15

Dari Tabel 4. dapat dilihat *Confusion matrix* persentase ada 39 data diprediksi secara benar seagai kelas abnormal, 6 data yang seharusnya masuk kelas normal tetapi dikenali sebagai kelas abnormal, lalu 15 data berhasil diklasifikasi sebagai kelas normal, dan 2 data dikenali sebagai kelas normal yang seharusnya masuk dalam klasifikasi kelas abnormal. Berdasarkan Tabel 3. Nilai akurasi tertinggi metode *k-fold cross validation* nilai akurasi tertinggi pada parameter k=3 sebesar 82,3% yang lebih kecil dibandingkan dengan metode persentase split, dengan nilai presisi untuk abnormal yaitu 89% dan normal 68%, sedangkan untuk nilai recall abnormal yaitu 82% dan normal 79%. *Confusion matrix* dari algoritma KNN dengan metode k- fold validation dapat dilihat pada Tabel 5.

Tabel 5. *Confusion matrix k-fold cross validation*

Kelas		Prediksi Sebenarnya	
		Abnormal	Normal
Label	Abnormal	173	37
Prediksi	Normal	21	79

Dari Tabel 5. dapat dilihat *Confusion matrix* pada k-fold cross validation ada 252 data diprediksi secara benar dan 58 data diprediksi dalam kelas yang salah, dengan 173 data diprediksi secara benar sebagai kelas abnormal, 21 data yang seharusnya masuk kelas normal tetapi dikenali sebagai kelas abnormal, lalu 79 data berhasil diklasifikasi sebagai kelas normal, dan 37 data dikenali sebagai kelas normal yang seharusnya masuk dalam klasifikasi kelas abnormal.

### 3.2. Algoritma Support Vector Machine (SVM)

Pada penelitian ini kernel yang digunakan yaitu kernel linier dari model algoritma SVM. Metode training yang digunakan yaitu persentase split dan K-Fold Cross Validation. *Confusion matrix* dari algoritma SVM menggunakan metode persentase split bisa dilihat pada Tabel 6.

Tabel 6. *Confusion matrix* persentase split

	Kelas	Prediksi Sebenarnya	
		Abnormal	Normal
Label	Abnormal	52	9
Prediksi	Normal	8	24

Dari Tabel 7. dapat dilihat pada *Confusion matrix* persentase split ada 52 data diprediksi secara benar sebagai kelas abnormal, 8 data yang seharusnya masuk kelas normal tetapi dikenali sebagai kelas abnormal, lalu 24 data berhasil diklasifikasi sebagai kelas normal, dan 9 data dikenali sebagai kelas normal yang seharusnya masuk dalam klasifikasi kelas abnormal. Dari *Confusion matrix* persentase split dapat dihitung nilai akurasi dengan algoritma SVM yang diperoleh dari jumlah data yang diprediksi secara benar, yaitu 83,8% yang lebih besar dibandingkan hasil akurasi algoritma KNN. Hasil perhitungan *Recall* untuk masing- masing kelas yaitu abnormal 92% dan normal 72%, serta hasil perhitungan *precision* untuk masing-masing kelas yaitu 83% dan 86%. *Confusion matrix* dari algoritma SVM menggunakan metode K-Fold Cross Validation bisa dilihat pada Tabel 7.

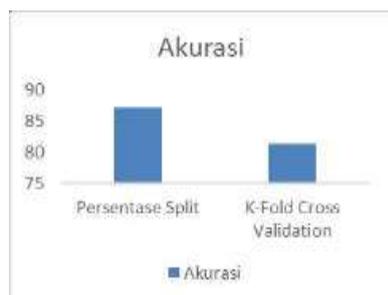
Tabel 7. *Confusion matrix k-fold cross validation*

	Kelas	Prediksi Sebenarnya	
		Abnormal	Normal
Label	Abnormal	180	30
Prediksi	Normal	24	76

Berdasarkan Tabel 7. dapat dilihat pada *Confusion matrix* pada k-fold cross validation terdapat 180 data diprediksi secara benar sebagai kelas abnormal, 24 data yang seharusnya masuk kelas normal tetapi dikenali sebagai kelas abnormal, lalu 76 data berhasil diklasifikasi sebagai kelas normal, dan 30 data dikenali sebagai kelas normal yang seharusnya masuk dalam klasifikasi kelas abnormal. Dari *Confusion matrix* k-fold cross validation dapat dihitung nilai akurasi yang diperoleh dari jumlah data yang diprediksi secara benar, yaitu 82,5%. Hasil ini menunjukkan algoritma KNN cukup baik dalam klasifikasi nyeri punggung bawah. Hasil perhitungan *Recall* untuk masing-masing kelas yaitu abnormal 86% dan normal 76%, serta hasil perhitungan *precision* untuk masing-masing kelas yaitu 72% dan 76%.

**3.3. Perbandingan Hasil Kedua Algoritma**

Hasil prediksi dua algoritma KNN dan SVM, terlihat bahwa dalam mengklasifikasi nyeri punggung bawah algoritma KNN dan SVM dapat bekerja dengan baik. Nilai akurasi dari kedua algoritma bisa dilihat pada Gambar 2 dan 3.

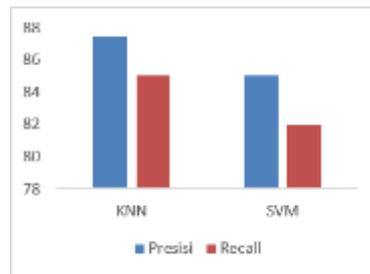


Gambar 2. Nilai akurasi algoritma K-Nearest Neighbor (KNN)

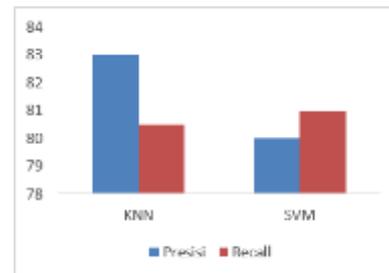


Gambar 3. Nilai akurasi algoritma Support Vector Machine (SVM)

Pada gambar 2 dan 3 dapat dilihat nilai akurasi dari algoritma KNN dan SVM. Pada kedua algoritma metode training persentase split memiliki nilai akurasi yang lebih besar dibandingkan dengan k-fold cross validation. Pada algoritma KNN nilai akurasi dengan persentase split sebesar 87,1% yang lebih besar dibandingkan algoritma SVM dengan persentase split sebesar 83,8%. Kemudian hasil *precision* dan *recall* pada algoritma KNN dan SVM dapat dilihat secara ringkas pada Gambar 4 dan 5.



Gambar 4. Nilai rata-rata presisi dan recall dengan metode persentase split



Gambar 5. Nilai rata-rata presisi dan recall dengan metode k-fold cross validation

Pada gambar 4 dapat dilihat bahwa nilai rata-rata presisi dan recall dari algoritma KNN dengan metode training persentase split sebesar 87,5% dan 85% yang lebih besar dibandingkan dengan algoritma SVM 85% dan 82%. Pada gambar 5 dapat dilihat nilai rata-rata presisi dan recall dengan metode training k-fold cross validation nilai rata-rata presisi algoritma KNN sebesar 83% yang lebih besar dibandingkan algoritma SVM sebesar 80,5% dan untuk nilai recall algoritma KNN sebesar 80% yang lebih kecil dibandingkan algoritma SVM sebesar 81%.

#### 4. KESIMPULAN

Penelitian ini mengevaluasi metode algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) pada dataset *Lower Back Pain Symptoms* yang terdiri dari 310 data. Berdasarkan hasil penelitian pada algoritma KNN memberikan hasil akurasi, presisi, dan recall terbaik pada metode persentase split dengan parameter  $k=10$  dengan nilai di atas 85%. Begitu juga pada algoritma SVM hasil terbaik diberikan oleh metode persentase split dengan nilai akurasi serta nilai rata-rata presisi dan recall di atas 80%. Artinya algoritma KNN bekerja lebih baik dalam mengklasifikasi nyeri punggung bawah dibandingkan dengan algoritma SVM.

#### REFERENSI

- [1] R. Sahara and T. Y. Pristya, "Faktor Risiko yang Berhubungan dengan Kejadian Low Back Pain (LBP) pada Pekerja: Literature Review," *J. Ilm. Kesehat.*, vol. 19, no. 3, pp. 92–99, 2020, [Online]. Available: <https://journals.stikom.ac.id/index.php/jikes/article/download/585/499/>.
- [2] R. N. Waykole and A. D. Thakare, "Classification of Low Back Pain Using Deep Learning Neural Network Model," *J. Emerg. Technol. Innov. Res.*, vol. 10, no. 2, pp. 91–97, 2018.
- [3] N. P. Kumbea, A. Asrifuddin, and O. J. Sumampouw, "Keluhan Nyeri Punggung Bawah Pada Nelayan," *Indones. J. Public Heal. Community Med.*, vol. 2, no. 1, pp. 21–26, 2021.
- [4] A. A. Mortara, M. Permatasari, A. Desiani, Y. Andriani, and M. Arhami, "Perbandingan Algoritma C4.5 dan Adaptive Boosting dalam Klasifikasi Penyakit Alzheimer Comparison of C4.5 and Adaptive Boosting Algorithms in Alzheimer's Disease Classification," *J. Teknol. dan Inf.*, vol. 13, no. 2, pp. 196–207, 2023, doi: 10.34010/jati.v13i2.
- [5] M. A. Paramesti, A. F. Prawiningrum, and D. H. Akhmad, "Lower Back Pain Classification Using Machine Learning," in *Asia Pacific Conference on Research in Industrial and Systems Engineering (APCoRISE) Lower*, 2019, pp. 1–6.
- [6] T. A. Assegie, "Optimal Tree Depth in Decision Tree Classifiers for Predicting Heart Failure Mortality," *Heal. Front.*, vol. 1, pp. 58–66, 2023.
- [7] N. Neneng, K. Adi, and R. Isnanto, "Support Vector Machine Untuk Klasifikasi Citra Jenis Daging Berdasarkan Tekstur Menggunakan Ekstraksi Ciri Gray Level Co-Occurrence Matrices (GLCM)," *J. Sist. Inf. Bisnis*, vol. 6, no. 1, p. 1, 2016, doi: 10.21456/vol6iss1pp1-10.
- [8] I. C. R. Drajana, "Metode Support Vector Machine Dan Forward Selection Prediksi Pembayaran Pembelian Bahan Baku Kopra," *Ilk. J. Ilm.*, vol. 9, no. 2, pp. 116–123, 2017, doi: 10.33096/ilkom.v9i2.134.116-123.
- [9] I. Ayu, A. Sita, and A. Wahyu, "Klasifikasi Indeks Pembangunan Manusia dengan Metode K-Nearest Neighbor dan Support Vector Machine di Pulau Jawa," vol. 15, pp. 8–21, 2019.
- [10] A. J. T., D. Yanosma, and K. Anggriani, "Implementasi Metode K-Nearest Neighbor (Knn) Dan Simple Additive Weighting (Saw) Dalam Pengambilan Keputusan Seleksi Penerimaan Anggota Paskibraka," *Pseudocode*, vol. 3, no. 2, pp. 98–112, 2017, doi: 10.33369/pseudocode.3.2.98-112.
- [11] Y. Yahya and W. Puspita Hidayanti, "Penerapan Algoritma K-Nearest Neighbor Untuk Klasifikasi Efektivitas Penjualan Vape (Rokok Elektrik) pada 'Lombok Vape On,'" *Infotek J. Inform. dan Teknol.*, vol. 3, no. 2, pp. 104–114, 2020, doi: 10.29408/jit.v3i2.2279.
- [12] S. Rahmatullah, S. Wahyuni, M. F. Chaining, and F. C. Method, "Penerapan Data Mining Untuk Prediksi Penjualan Produk Terlaris Menggunakan Metode K-Nearest Neighbor," vol. 3, no. 2, pp. 75–86, 2020, doi: 10.47065/bits.v3i4.1408.
- [13] A. Desiani, "Perbandingan Implementasi Algoritma Naive Bayes dan K-Nearest Neighbor pada Klasifikasi Penyakit Hati," *J. Sist. Inf. dan Sist. Komput.*, vol. 7, no. 2, pp. 104–110, 2022.

- [14] P. M. Nirmala Dharmapatni and N. L. P. Merawati, "Penerapan Algoritma Support Vector Machine Dalam Sentimen Analisis Terkait Kenaikan Tarif BPJS Kesehatan," *J. Bumigora Inf. Technol.*, vol. 2, no. 2, pp. 105–112, 2020, doi: 10.30812/bite.v2i2.904.
- [15] A. Pratama, R. C. Wihandika, and D. E. Ratnawati, "Implementasi algoritme support vector machine (SVM) untuk prediksi ketepatan waktu kelulusan mahasiswa," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 2, no. April, pp. 1704–1708, 2018.
- [16] D. A. Anggoro, "Comparison of Accuracy Level of Support Vector Machine (SVM) and K-Nearest Neighbors (KNN) Algorithms in Predicting Heart Disease," *Int. J. Emerg. Trends Eng. Res.*, vol. 8, no. 5, pp. 1689–1694, 2020, doi: 10.30534/ijeter/2020/32852020.
- [17] N. Nurajijah and D. Riana, "Algoritma Naïve Bayes, Decision Tree, dan SVM untuk Klasifikasi Persetujuan Pembiayaan Nasabah Koperasi Syariah," *J. Teknol. dan Sist. Komput.*, vol. 7, no. 2, pp. 77–82, 2019, doi: 10.14710/jtsiskom.7.2.2019.77-82.
- [18] R. Munawarah, O. Soesanto, and M. R. Faisal, "Penerapan Metode Support Vector Machine Pada Diagnosa Hepatitis," *Kumpul. J. Ilmu Komput.*, vol. 04, no. 01, pp. 103–113, 2016.
- [19] I. W. Saputro and B. W. Sari, "Uji Performa Algoritma Naïve Bayes untuk Prediksi Masa Studi Mahasiswa," *Creat. Inf. Technol. J.*, vol. 6, no. 1, p. 1, 2020, doi: 10.24076/citec.2019v6i1.178.