

# Optimasi Metode Klasifikasi dengan *Particle Swarm Optimization (PSO)* dan Perbandingan Split Data untuk Prediksi Penyakit Diabetes

Elly Muningsih<sup>1\*</sup>, Sutrisno<sup>2</sup>, Vembria Rose Handayani<sup>3</sup>, Cindyra Galuhwardani<sup>4</sup>  
<sup>1,3,4</sup>Sistem Informasi, Universitas Bina Sarana Informatika, Indonesia  
<sup>2</sup>Teknologi Komputer, Universitas Bina Sarana Informatika, Indonesia  
<sup>1\*</sup>[elly.emh@bsi.ac.id](mailto:elly.emh@bsi.ac.id), <sup>2</sup>[sutrisno.stz@bsi.ac.id](mailto:sutrisno.stz@bsi.ac.id), <sup>3</sup>[vembria.vrh@bsi.ac.id](mailto:vembria.vrh@bsi.ac.id),  
<sup>4</sup>[cindyra.cyg@bsi.ac.id](mailto:cindyra.cyg@bsi.ac.id)

**Abstrak:** Diabetes merupakan penyakit kronis yang ditandai oleh gangguan metabolik dan peningkatan kadar gula darah melebihi batas normal. Prediksi dini diabetes penting dilakukan untuk membantu identifikasi risiko secara lebih cepat serta mendukung pengambilan keputusan dalam pencegahan dan penanganan awal. Penelitian ini bertujuan membandingkan performa metode *Decision Tree (DT)*, *Iterative Dichotomiser 3 (ID3)* dan *Support Vector Machine (SVM)* dalam prediksi diabetes dengan dan tanpa optimasi bobot fitur menggunakan *Particle Swarm Optimization (PSO)*. Dataset dibagi ke dalam tiga skenario, yaitu 70:30, 80:20, dan 90:10, untuk menganalisis pengaruh proporsi data *training* dan *testing*. Evaluasi dilakukan menggunakan *10-fold cross validation* pada data *training* serta pengujian pada data *testing* dengan metrik akurasi dan *Area Under Curve (AUC)*. Hasil penelitian menunjukkan bahwa PSO memberikan pengaruh berbeda pada setiap metode. Pada SVM, PSO meningkatkan akurasi *testing* secara konsisten pada seluruh skenario, yaitu dari 85,26% menjadi 89,10%, dari 75,00% menjadi 90,38%, dan dari 80,77% menjadi 94,23%. Pada DT, model tanpa PSO sudah menunjukkan performa tinggi, tetapi DT dengan PSO menghasilkan nilai tertinggi pada split 90:10 dengan akurasi 100% dan AUC 1,000. Pada ID3, PSO cenderung meningkatkan *AUC cross validation*, meskipun tidak selalu meningkatkan akurasi. Secara umum, PSO efektif meningkatkan performa model tertentu, khususnya SVM, namun efektivitasnya dipengaruhi oleh algoritma dan skenario pembagian data. Temuan ini dapat menjadi dasar pengembangan sistem prediksi diabetes yang lebih selektif dalam memilih fitur, metode klasifikasi, dan strategi evaluasi model akhir dalam konteks penerapan diagnosis awal berbasis data secara lebih akurat, stabil, dan terukur klinis.

**Kata kunci:** Diabetes; *Decision Tree*; *Particle Swarm Optimization (PSO)*; *Cross Validation*; Metode Klasifikasi;

**Abstract:** Diabetes is a chronic disease characterized by metabolic disorders and elevated blood glucose levels beyond the normal range. Early prediction of diabetes is important to support faster risk identification and assist decision-making in prevention and early treatment. This study aims to compare the performance of Decision Tree (DT), Iterative Dichotomiser 3 (ID3), and Support Vector Machine (SVM) methods in diabetes prediction, both with and without feature weight optimization using Particle Swarm Optimization (PSO). The

dataset was divided into three scenarios, namely 70:30, 80:20, and 90:10, to analyze the effect of training and testing data proportions. Evaluation was conducted using 10-fold cross-validation on the training data and testing on the testing data, with accuracy and Area Under the Curve (AUC) as evaluation metrics. The results show that PSO produced different effects on each method. In SVM, PSO consistently improved testing accuracy across all scenarios, from 85.26% to 89.10%, from 75.00% to 90.38%, and from 80.77% to 94.23%. In DT, the model without PSO already demonstrated high performance; however, DT with PSO achieved the highest result in the 90:10 split, with 100% accuracy and an AUC of 1.000. In ID3, PSO tended to improve cross-validation AUC, although it did not always increase accuracy. Overall, PSO is effective in improving the performance of certain models, particularly SVM; however, its effectiveness is influenced by the algorithm and data-splitting scenario. These findings can serve as a basis for developing a more selective diabetes prediction system in terms of feature selection, classification methods, and final model evaluation strategies for more accurate, stable, and clinically measurable data-driven early diagnosis.

**Keywords:** Diabetes; Decision Tree; Particle Swarm Optimization (PSO); Cross-Validation; Classification Methods;

## 1. PENDAHULUAN

*Diabetes Mellitus (DM)* merupakan penyakit metabolik kronis yang ditandai dengan meningkatnya kadar glukosa darah atau hiperglikemia akibat gangguan sekresi insulin, kerja insulin, atau kombinasi keduanya [1], [2]. Kondisi ini terjadi ketika tubuh tidak mampu menghasilkan insulin dalam jumlah yang cukup atau tidak dapat memanfaatkan insulin secara optimal, sehingga glukosa tidak dapat digunakan secara efektif oleh sel tubuh dan tetap berada dalam aliran darah. Diabetes menjadi salah satu masalah kesehatan yang terus meningkat prevalensinya, baik secara global maupun di Indonesia, sehingga diperlukan perhatian serius terhadap upaya deteksi dini dan pencegahan komplikasi [2]. Selain itu, diabetes juga dapat menimbulkan komplikasi serius seperti penyakit jantung, gagal ginjal, dan kerusakan saraf apabila tidak terdeteksi dan ditangani sejak dini [3]. Diabetes umumnya diklasifikasikan menjadi diabetes tipe 1 dan tipe 2, di mana diabetes tipe 1 berkaitan dengan gangguan produksi insulin oleh pankreas, sedangkan diabetes tipe 2 umumnya disebabkan oleh resistensi insulin atau rendahnya sensitivitas sel tubuh terhadap insulin [1]. Beberapa faktor klinis yang sering berkaitan dengan diabetes antara lain kadar glukosa darah, indeks massa tubuh atau *Body Mass Index (BMI)*, tekanan darah, kadar insulin, usia, dan riwayat keluarga terhadap diabetes. Faktor-faktor tersebut banyak digunakan sebagai atribut dalam dataset prediksi diabetes karena memiliki hubungan dengan risiko seseorang mengalami diabetes [3], [4]. Oleh karena itu, pemanfaatan data kesehatan dalam prediksi diabetes menjadi penting untuk membantu proses identifikasi risiko secara lebih awal dan mendukung pengambilan keputusan berbasis data dalam bidang kesehatan.

Beberapa penelitian menunjukkan bahwa metode klasifikasi telah banyak digunakan untuk prediksi penyakit diabetes, diantaranya metode *Naïve Bayes* dilaporkan sedikit lebih unggul dibanding KNN pada dataset diabetes wanita dengan akurasi 78,35% dibanding 77,98%, sedangkan penelitian lain menunjukkan KNN memperoleh akurasi tertinggi 99% pada 2.000 data, meskipun pada pengujian 30 data *Naïve Bayes* lebih tinggi yaitu 66% dibanding KNN 53% [1], [5]. Selain itu, *Naïve Bayes* juga digunakan secara khusus untuk memprediksi penderita diabetes berdasarkan 502 data rekam medis, dengan beberapa skenario pengujian dan akurasi tertinggi yang dilaporkan mencapai 95,92% [6]. Pada metode *Decision Tree*, penelitian sebelumnya menunjukkan bahwa algoritma ini dapat digunakan untuk klasifikasi diabetes, namun masih memiliki keterbatasan pada nilai *recall*

kelas positif diabetes, dengan *precision* 0,78, *recall* 0,45, dan *F1-score* 0,57 [7]. Sementara itu, *Random Forest* juga dilaporkan memiliki kinerja baik pada data klinis lokal dengan akurasi 87%, *precision* 84%, *recall* 82%, dan *F1-score* 83%, serta menunjukkan bahwa glukosa darah, HbA1c, dan BMI menjadi fitur yang berpengaruh dalam prediksi diabetes [3].

Temuan tersebut menunjukkan bahwa pemilihan fitur yang relevan memiliki peran penting dalam meningkatkan kinerja model klasifikasi. Sejalan dengan hal tersebut, *Particle Swarm Optimization (PSO)* banyak digunakan untuk meningkatkan kinerja metode klasifikasi melalui optimasi parameter maupun seleksi fitur pada kasus prediksi penyakit diabetes. Penelitian tersebut antara lain membandingkan *Naive Bayes*, *C4.5*, dan *Random Forest* menunjukkan bahwa *Random Forest* menghasilkan akurasi tertinggi sebesar 95,15% tanpa optimasi, sedangkan penggunaan *PSO* mampu meningkatkan performa model yang sebelumnya kurang optimal, terutama pada *Naive Bayes* dan *C4.5* [8]. Pada metode *Decision Tree*, kombinasi *preprocessing*, *SMOTE*, dan *PSO* mampu meningkatkan akurasi model dari 95,21% menjadi 96,52%, serta meningkatkan *precision* dan *F1-score*, meskipun nilai *recall* mengalami sedikit penurunan [9]. Pada metode *Support Vector Machine (SVM)*, integrasi *PSO* dan *SMOTE* mampu meningkatkan akurasi dari 82,72% menjadi 83,95%, terutama dalam membantu optimasi parameter dan meningkatkan prediksi pada kelas minoritas [4]. Temuan serupa juga diperoleh pada penelitian *SVM* lainnya, yaitu *SVM* yang dioptimasi menggunakan *PSO* menghasilkan akurasi rata-rata 83,60%, sedikit lebih tinggi dibandingkan *SVM* konvensional sebesar 83,39% berdasarkan evaluasi *5-fold cross validation* [10]. Selain pada domain kesehatan, *PSO* juga terbukti efektif dalam optimasi model klasifikasi lain, seperti *Random Forest* untuk prediksi prestasi akademik mahasiswa, dengan peningkatan akurasi dari 86,7% menjadi 89,4% serta pemanfaatan *feature importance* untuk mengetahui kontribusi setiap fitur terhadap hasil klasifikasi [11].

Sedikit berbeda dari penelitian sebelumnya, penelitian ini bertujuan untuk membandingkan kinerja metode klasifikasi *Decision Tree (DT)*, *Iterative Dichotomiser 3 (ID3)* dan *Support Vector Machine (SVM)* dalam prediksi penyakit DM, baik tanpa optimasi maupun dengan optimasi *Particle Swarm Optimization (PSO)* dimana pengujian dilakukan menggunakan tiga skenario pembagian data *training* dan *testing*, yaitu 70:30, 80:20, dan 90:10, untuk mengetahui pengaruh komposisi data terhadap hasil klasifikasi. Evaluasi model dilakukan melalui *cross validation* dan pengujian pada data testing dengan menggunakan nilai akurasi dan AUC sebagai indikator kinerja. Selain itu, proses seleksi fitur dilakukan menggunakan *Select by Weight* dengan parameter *weight relation greater* dan *weight* 0,3. Dataset yang digunakan berasal dari *UCI Machine Learning Repository* sebanyak 520 record, terdiri dari 16 atribut dan satu label *Class* yang menunjukkan risiko Diabetes Mellitus. Seluruh proses pengolahan dan pengujian data dilakukan menggunakan software *AI Altair Studio*.

## 2. METODE PENELITIAN

### Data Preparation

*Data Preparation* atau persiapan data yaitu menyiapkan data yang akan digunakan dalam proses pengolahan. Sebelum dilakukan *Data Preparation* dilakukan terlebih dahulu proses *Business Understanding* dan *Data Understanding*. Pengetahuan tentang dataset dilakukan dalam tahap ini. Dataset yang digunakan dalam penelitian ini adalah Diabetes Symptom Dataset yang diperoleh dari *UCI Machine Learning Repository*. Dataset ini berisi data laporan gejala terkait Diabetes Mellitus (DM) sebanyak 520 orang, yang dikumpulkan melalui kuesioner langsung dari pasien di Sylhet Diabetes Hospital, Sylhet, Bangladesh. Dataset memiliki 16 atribut, yaitu *Age*, *Sex*, *Polyuria*, *Polydipsia*, *sudden weight loss*, *weakness*, *Polyphagia*, *Genital thrush*, *visual blurring*, *Itching*, *Irritability*, *delayed healing*, *partial paresis*, *muscle stiffness*, *Alopecia*, dan *Obesity*, serta satu atribut target yaitu *Class*.

Atribut *Class* digunakan untuk menunjukkan status risiko DM, dengan nilai *Positive* untuk pasien yang berisiko atau terindikasi DM dan *Negative* untuk pasien yang tidak berisiko DM. Distribusi kelas pada dataset terdiri dari 320 data *Positive* dan 200 data *Negative* yang menunjukkan adanya ketidakseimbangan kelas dalam kategori ringan dengan rasio sekitar 1,6:1. Pada atribut *Age*, nilai dikategorikan menjadi lima kelompok, yaitu 1 = 20–35 tahun, 2 = 36–45 tahun, 3 = 46–55 tahun, 4 = 56–65 tahun, dan 5 = di atas 65 tahun. Atribut *Sex* diberi nilai 1 untuk laki-laki dan 2 untuk perempuan, sedangkan atribut gejala lainnya memiliki nilai kategorikal *Yes* dan *No*.

## Metode Klasifikasi dan Modelling

### a. Metode Klasifikasi

Metode klasifikasi yang akan digunakan dalam penelitian ini adalah metode *Decision Tree (DT)*, *ID3* dan *Support Vector Machine (SVM)*. Penjelasan mengenai masing-masing metode dijelaskan dibawah ini.

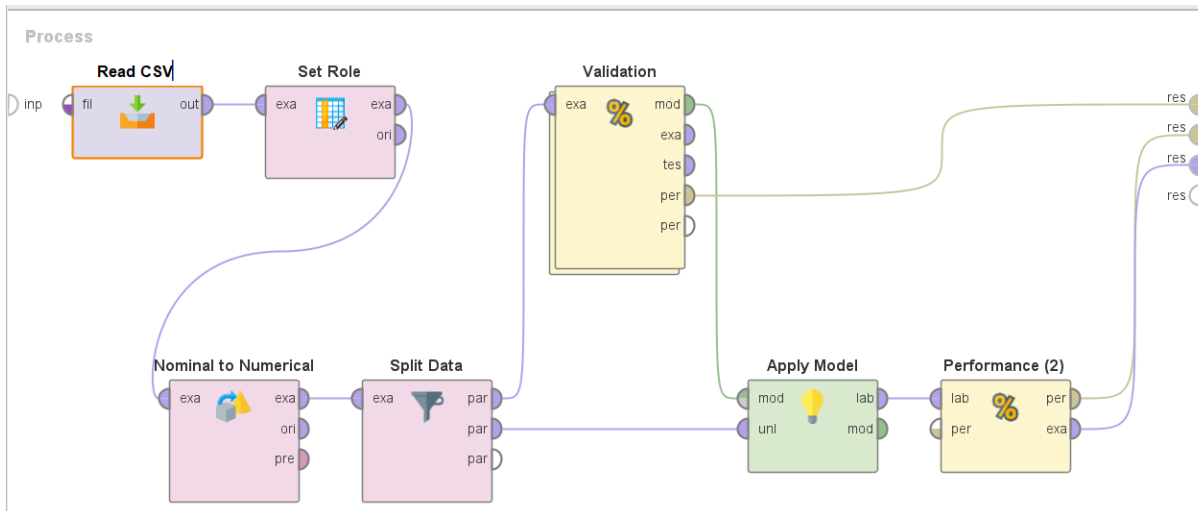
**Decision Tree (DT)** adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi data dan analisis prediktif yang digunakan pada *data mining*, *knowledge discovery*, *machine learning* dan masalah *artificial intelligence* [12]. DT adalah algoritma yang mempartisi individu-individu ke dalam kelompok-kelompok yang serupa berdasarkan variabel yang akan diprediksi dan hasil dari algoritma menghasilkan pohon yang mengungkapkan hierarki hubungan antarvariabel [13].

**Iterative Dichotomiser 3 (ID3)** ditemukan oleh Ross Quinlan dimana algoritmanya untuk menghasilkan pohon keputusan dari dataset dan biasanya digunakan pada *machine learning* dan domain *Natural Language Processing (NLP)* [14]. Tujuan dari algoritma ID3 adalah untuk memperoleh informasi kelas terbesar tentang sub-set data saat membuat keputusan pada simpul internal dan dijelaskan bahwa inti dari algoritma ini adalah menghitung perolehan informasi sebagai kriteria pemilihan atribut uji untuk tingkat hierarki node non-daun dalam pohon keputusan [15]

**Support Vector Machine (SVM)** adalah algoritma pengaturan yang diperkenalkan oleh Vapnik pada tahun 1995 dimana basis algoritmanya menggunakan presisi untuk menggeneralisasi kesalahan yaitu dengan cara membuat "*hyperplane*" dan membagi data ke dalam kelas-kelas sehingga semua sampel yang termasuk dalam satu kelas akan dikategorikan pada satu sisi dan di sisi lainnya [16]. SVM dapat meningkatkan akurasi model melalui optimalisasi pemisahan ruang menggunakan fungsi kernel, tetapi salah satu kelemahan SVM adalah membutuhkan banyak percobaan untuk membangun model SVM yang optimal dibandingkan dengan teknik pembelajaran mesin lainnya [17].

### b. Modelling

*Modelling* atau pemodelan dilakukan menggunakan *Altair AI Studio*, yaitu *platform* berbasis *visual workflow* yang merupakan penerus dari *RapidMiner* yang resmi menjadi bagian dari *Altair* pada tahun 2022. *Tools* ini dipilih karena mendukung proses *machine learning* secara sistematis dan mudah direplikasi, mulai dari *preprocessing*, pembagian data, validasi, penerapan model, evaluasi performa, hingga optimasi PSO. Operator yang digunakan meliputi *Nominal to Numerical*, *Split Data*, *Cross Validation*, *Apply Model*, dan *Performance* untuk memperoleh nilai akurasi dan AUC. Pemodelan menggunakan *Altair AI Studio* untuk metode klasifikasi tanpa PSO ditampilkan pada Gambar 1.



Gambar 1. Modelling

Pada operator *Split Data*, pengolahan dilakukan tiga kali untuk masing-masing metode klasifikasi yang dibandingkan. *Split Data* yang digunakan adalah 70:30, 80:20 dan 90:10. Angka pertama menunjukkan data *training* dan angka kedua menunjukkan data *testing* atau data validasi.

### Evaluation Metode Klasifikasi

Metode Klasifikasi pada penelitian ini dievaluasi menggunakan *10-fold cross validation*, *confusion matrix*, akurasi, dan *Area Under Curve (AUC)*. Evaluasi dilakukan pada data *training* melalui *10-fold cross validation* untuk memperoleh rata-rata performa model, kemudian dilanjutkan dengan pengujian pada data *testing* berdasarkan skenario split data 70:30, 80:20, dan 90:10. Evaluasi ini digunakan untuk membandingkan performa metode DT, ID3, dan SVM, baik tanpa optimasi maupun dengan optimasi bobot fitur menggunakan PSO.

#### a. Confusion Matrix.

*Confusion matrix* digunakan untuk mengetahui kesesuaian antara kelas aktual dan kelas hasil prediksi. Tabel ini menghasilkan nilai *True Positive (TP)*, *True Negative (TN)*, *False Positive (FP)*, dan *False Negative (FN)* yang selanjutnya digunakan untuk menghitung akurasi. Akurasi menunjukkan proporsi prediksi yang benar terhadap seluruh data uji. Namun, akurasi tidak selalu cukup menggambarkan performa model, terutama jika data tidak seimbang, sehingga diperlukan metrik tambahan seperti AUC [18], [19]

#### b. Receiver Operating Characteristic (ROC) dan Area Under Curve (AUC)

*Receiver Operating Characteristic (ROC)* merupakan kurva yang menunjukkan hubungan antara *True Positive Rate (TPR)* dan *False Positive Rate (FPR)* pada berbagai ambang klasifikasi. Sementara itu, AUC menunjukkan luas area di bawah kurva ROC yang menggambarkan kemampuan model dalam membedakan kelas positif dan negatif. Semakin mendekati nilai 1, semakin baik performa klasifikasi model. Dalam penelitian ini, AUC digunakan bersama akurasi untuk mengevaluasi performa model secara lebih komprehensif [18], [19].

## 3. HASIL DAN PEMBAHASAN

### Nilai Akurasi dan AUC

Dari pengolahan data yang dilakukan dengan menggunakan tiga skenario perbandingan data *training* dan *testing*, yaitu 70:30, 80:20, dan 90:10, diperoleh hasil

evaluasi dari masing-masing metode klasifikasi, yaitu DT, SVM, dan ID3. Setiap metode diuji tanpa optimasi dan dengan optimasi PSO. Hasil evaluasi yang ditampilkan meliputi akurasi *cross validation* (CV), AUC CV, akurasi *testing*, dan AUC *testing*. Hasil pengujian untuk split data 70:30 disajikan pada Tabel 1, split data 80:20 pada Tabel 2, dan split data 90:10 pada Tabel 3.

**Tabel 1.** Perbandingan Split Data 70:30

Metode	Akurasi CV Tanpa PSO	Akurasi CV + PSO	AUC CV Tanpa PSO	AUC CV + PSO	Akurasi Testing Tanpa PSO	Akurasi Testing + PSO	AUC Testing Tanpa PSO	AUC Testing + PSO
DT	94.82% +/- 4.50%	93.16% +/- 4.27%	0.948 +/- 0.042	0.891 +/- 0.143	96.79%	92.31%	0.968	0.909
SVM	87.63% +/- 5.11%	88.71% +/- 6.84%	0.967 +/- 0.028	0.971 +/- 0.026	85.26%	89.10%	0.975	0.949
ID3	94.54% +/- 4.04%	91.21% +/- 2.18%	0.845 +/- 0.187	0.929 +/- 0.041	94.87%	96.15%	0.962	0.962

Dari Tabel 1 diketahui penggunaan PSO belum memberikan peningkatan yang konsisten pada semua metode. Pada DT, PSO justru menurunkan akurasi dan AUC, baik pada *cross validation* maupun *testing*. Pada SVM, PSO meningkatkan akurasi, tetapi AUC *testing* menurun. Pada ID3, PSO meningkatkan accuracy testing dari 94.87% menjadi 96.15% dan AUC CV dari 0.845 menjadi 0.929, meskipun akurasi CV menurun. Secara keseluruhan, model dengan performa terbaik untuk akurasi *testing* adalah DT tanpa PSO dengan nilai 96.79%, sedangkan AUC *testing* tertinggi diperoleh SVM tanpa PSO dengan nilai 0.975.

**Tabel 2.** Perbandingan Split Data 80:20

Metode	Akurasi CV Tanpa PSO	Akurasi CV + PSO	AUC CV Tanpa PSO	AUC CV + PSO	Akurasi Testing Tanpa PSO	Akurasi Testing + PSO	AUC Testing Tanpa PSO	AUC Testing + PSO
DT	94.45% +/- 3.46%	93.01% +/- 4.81%	0.953 +/- 0.041	0.949 +/- 0.039	97.12%	95.19%	0.981	0.973
SVM	87.53% +/- 3.95%	88.94% +/- 5.45%	0.968 +/- 0.016	0.952 +/- 0.025	75.00%	90.38%	0.907	0.964
ID3	94.47% +/- 2.28%	93.28% +/- 3.35%	0.770 +/- 0.234	0.938 +/- 0.042	95.19%	95.19%	0.984	0.962

Penggunaan PSO dari Tabel 2 diketahui belum memberikan peningkatan yang konsisten pada semua metode. Pada DT, PSO justru menurunkan akurasi dan AUC, baik pada CV maupun *testing*. Pada SVM, PSO memberikan peningkatan paling besar pada data *testing*, yaitu akurasi naik dari 75.00% menjadi 90.38% dan AUC naik dari 0.907 menjadi 0.964. Pada ID3, PSO meningkatkan AUC CV, tetapi tidak meningkatkan akurasi *testing* dan AUC *testing* justru menurun. Secara keseluruhan, performa terbaik berdasarkan akurasi *testing* diperoleh oleh DT tanpa PSO sebesar 97.12%, sedangkan AUC *testing* tertinggi diperoleh

oleh ID3 tanpa PSO sebesar 0.984. Jadi, pada split 80:20, PSO paling bermanfaat untuk SVM, tetapi tidak selalu meningkatkan performa metode lainnya.

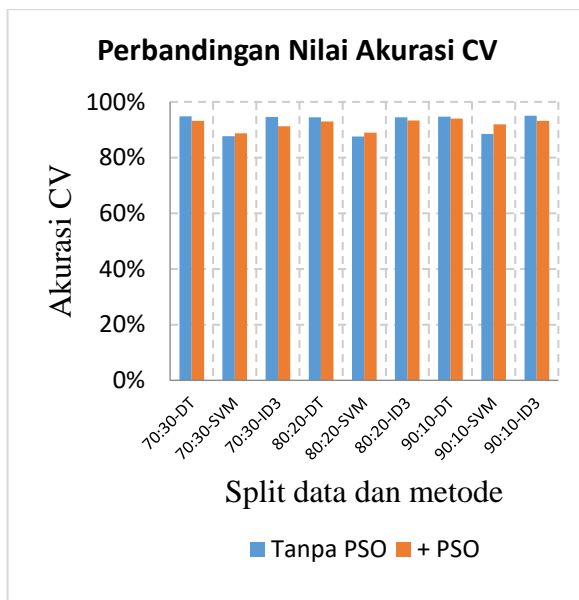
**Tabel 3.** Perbandingan Split Data 90:10

Metode	Akurasi CV Tanpa PSO	Akurasi CV + PSO	AUC CV Tanpa PSO	AUC CV + PSO	Akurasi Testing Tanpa PSO	Akurasi Testing + PSO	AUC Testing Tanpa PSO	AUC Testing + PSO
DT	94.67%	94.01%	0.938	0.953	96.15%	100.00%	0.965	1.000
	+/-	+/-	+/-	+/-				
	3.92%	2.64%	0.056	0.035				
SVM	88.47%	91.89%	0.972	0.958	80.77%	94.23%	0.972	0.975
	+/-	+/-	+/-	+/-				
	4.82%	3.97%	0.017	0.021				
ID3	95.08%	93.15%	0.858	0.924	96.15%	94.23%	0.943	0.965
	+/-	+/-	+/-	+/-				
	3.91%	3.02%	0.193	0.049				

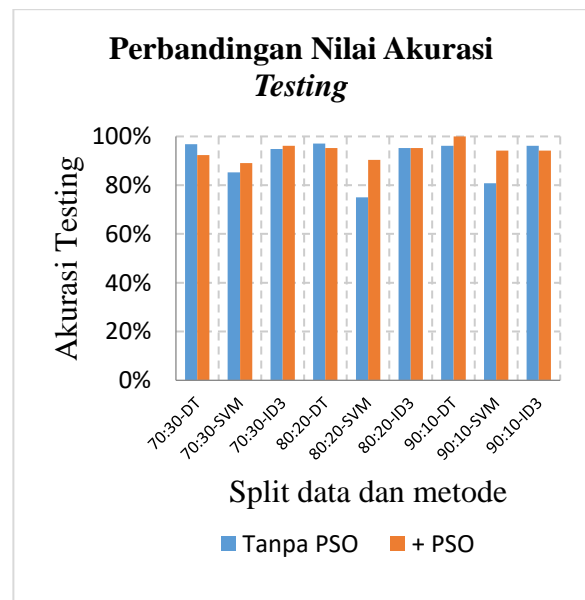
Penggunaan PSO di Tabel 3 ini memberikan peningkatan terbaik pada metode DT, dengan akurasi *testing* naik dari 96.15% menjadi 100.00% dan AUC *testing* naik dari 0.965 menjadi 1.000. Pada SVM, PSO juga meningkatkan akurasi *testing* dari 80.77% menjadi 94.23% dan AUC *testing* dari 0.972 menjadi 0.975. Namun, pada ID3, PSO menurunkan akurasi *testing* dari 96.15% menjadi 94.23%, meskipun AUC *testing* meningkat.

### Perbandingan Nilai Akurasi

Perbandingan nilai akurasi CV dan *testing* dengan split data untuk metode klasifikasi yang digunakan dalam penelitian ini ditampilkan dalam Gambar 2 dan Gambar 3.



**Gambar 2.** Perbandingan Akurasi CV



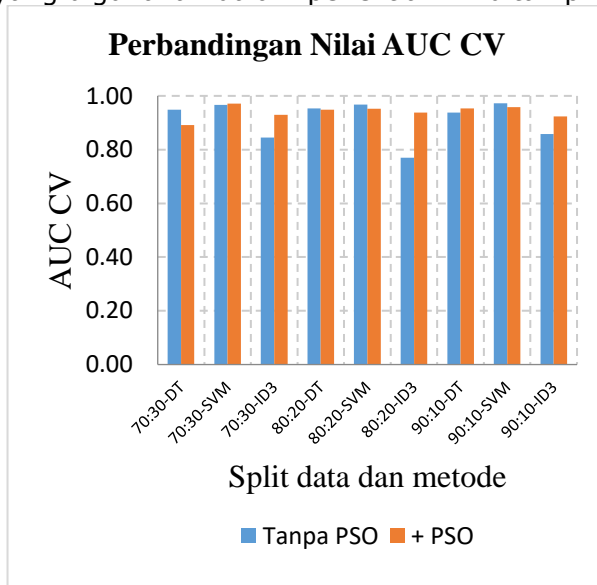
**Gambar 3.** Perbandingan Akurasi Testing

Berdasarkan grafik Gambar 3 diketahui bahwa penggunaan PSO tidak selalu meningkatkan akurasi pada semua metode. Pada grafik akurasi CV, PSO cenderung meningkatkan performa SVM, tetapi menurunkan akurasi pada DT dan ID3 di beberapa split data. Hal ini

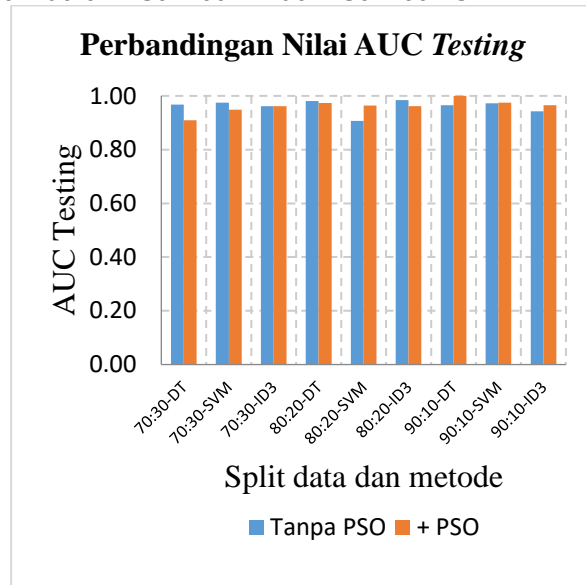
menunjukkan bahwa pengaruh PSO pada proses validasi masih bergantung pada metode klasifikasi yang digunakan. Sedangkan pada grafik Gambar 3 diketahui bahwa peningkatan paling konsisten terlihat pada SVM + PSO, terutama pada split data 80:20 dan 90:10. Metode DT + PSO menghasilkan performa tertinggi pada split 90:10 dengan akurasi mencapai 100%, tetapi pada split 70:30 dan 80:20 justru lebih rendah dibandingkan DT tanpa PSO. Sementara itu, performa ID3 + PSO cenderung tidak stabil. Dengan demikian, PSO paling efektif meningkatkan akurasi pada metode SVM, sedangkan performa terbaik secara keseluruhan diperoleh pada DT + PSO dengan split data 90:10.

### Perbandingan Nilai AUC

Perbandingan nilai AUC CV dan AUC *testing* dengan split data untuk metode klasifikasi yang digunakan dalam penelitian ini ditampilkan dalam Gambar 4 dan Gambar 5.



Gambar 4. Perbandingan Nilai AUC CV



Gambar 5. Perbandingan AUC Testing

Berdasarkan grafik perbandingan nilai AUC CV dan AUC *testing* pada Gambar 4 dan Gambar 5, penggunaan PSO tidak memberikan peningkatan yang konsisten pada semua metode. Pada grafik AUC CV, PSO mampu meningkatkan nilai AUC pada ID3 di semua split data, terutama pada split 80:20, tetapi pada DT dan SVM peningkatannya tidak selalu terjadi. Pada grafik AUC *testing*, nilai AUC sebagian besar sudah berada pada kategori tinggi, yaitu mendekati 1,00. PSO memberikan peningkatan yang cukup baik pada SVM untuk split 80:20 dan 90:10, serta menghasilkan nilai terbaik pada DT split 90:10 dengan AUC mencapai 1,00. Namun, pada beberapa metode seperti DT split 70:30 dan 80:20, PSO justru menurunkan nilai AUC.

### Bobot Fitur dengan PSO

Optimasi fitur pada penelitian ini dilakukan menggunakan *Particle Swarm Optimization* (PSO) untuk menghasilkan bobot pada setiap fitur. Selanjutnya, proses pemilihan fitur dilakukan menggunakan operator *Select by Weight* dengan parameter *weight relation greater* dan nilai ambang *weight* 0,3. Artinya, hanya fitur yang memiliki bobot lebih besar dari 0,3 yang digunakan dalam proses klasifikasi. Berdasarkan hasil seleksi, diperoleh sepuluh fitur terpilih. Dari fitur tersebut, *Polyuria*, *Polydipsia*, *Irritability*, dan *Age* memperoleh bobot maksimum sebesar 1,000, sehingga menunjukkan kontribusi paling dominan terhadap proses klasifikasi. Sementara itu, fitur lainnya tetap digunakan

karena memiliki bobot di atas batas minimum yang telah ditentukan. Hasil fitur terpilih berdasarkan bobot PSO ditampilkan pada Tabel 4.

**Tabel 4.** Fitur terpilih dengan bobot PSO

No	Fitur Terpilih	Bobot Fitur
1	<i>Polyuria, Polydipsia, Irritability, Age</i>	1,000
2	<i>Genital thrush</i>	0,841
3	<i>Visual blurring</i>	0,649
4	<i>Itching</i>	0,481
5	<i>Polyphagia</i>	0,446
6	<i>Alopecia</i>	0,413
7	<i>Weakness</i>	0,403

#### 4. KESIMPULAN

Berdasarkan hasil pengujian, penggunaan PSO tidak selalu meningkatkan performa semua metode klasifikasi. Peningkatan paling konsisten terjadi pada SVM, sedangkan performa terbaik diperoleh pada DT + PSO dengan split data 90:10, yaitu *accuracy testing* 100% dan *AUC testing* 1,000. Selain itu, PSO juga digunakan untuk seleksi fitur dengan operator *Select by Weight* dengan parameter *weight relation greater* dan *weight* 0,3, sehingga diperoleh sepuluh fitur terpilih. Fitur *Polyuria, Polydipsia, Irritability, dan Age* memiliki bobot tertinggi sebesar 1,000. Secara keseluruhan, PSO dapat membantu meningkatkan performa model dan memilih fitur yang relevan, tetapi efektivitasnya bergantung pada metode klasifikasi dan pembagian data yang digunakan. Pada penelitian selanjutnya, pengujian dapat dikembangkan dengan menggunakan variasi parameter yang berbeda serta membandingkan model klasifikasi lainnya, sehingga dapat diketahui pengaruhnya terhadap peningkatan akurasi dan kinerja model secara lebih komprehensif.

#### 5. REFERENCES

- [1] J. Lemantara and T. Lusiani, "Analisis Prediksi Penyakit Diabetes Pada Wanita Menggunakan Metode Naïve Bayes Dan K-Nearest Neighbor," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.4911.
- [2] E. Muningsih, F. Rizki, and K. D. Asiffa, "Diabetes Prediction System 'Diapres' Berbasis Optimasi Parameter Pada Metode Decision Tree," *J. Teknoinfo*, vol. 18, no. 1, pp. 305–315, 2024.
- [3] E. da C. Pereira and W. Andriyani, "Prediksi Diabetes Menggunakan Machine Learning Diabetes Prediction Using Machine Learning," *Jiko (Jurnal Inform. Dankomputer)*, vol. 9, no. 3, pp. 639–649, 2025, doi: 10.26798/jiko.v9i3.2104.
- [4] S. Daniswara, Z. I. Arishandy, and E. Y. Puspaningrum, "Jurnal Ilmiah Klasifikasi Penyakit Diabetes Menggunakan Particle Swarm Optimaze Pada Algoritma Support Vector Machine," *J. Ilmiah, Teknol. Inf. dan Robot.*, vol. 7, no. 2, pp. 20–28, 2025, doi: 10.33005/jifti.v7i2.163.
- [5] F. N. Ikhromr, I. Sugiyarto, U. Faddillah, and B. Sudarsono, "Implementasi Data Mining Untuk Memprediksi Penyakit Diabetes Menggunakan Algoritma Naives Bayes dan K-Nearest Neighbor," *INTECOMS J. Inf. Technol. Comput. Sci.*, vol. 6, no. 1, pp. 416–428, 2023, doi: 10.31539/intecom.v6i1.5916.
- [6] C. A. Rahayu, R. Hartono, and A. Sudiarjo, "PREDIKSI PENDERITA DIABETES MENGGUNAKAN METODE NAIVE BAYES," *JITET (Jurnal Inform. dan Tek. Elektro Terap.)*, vol. 11, pp. 261–266, 2023.
- [7] Nurussakinah and M. Faisal, "Klasifikasi Penyakit Diabetes Menggunakan Algoritma

- Decision Tree," *J. Inform.*, vol. 10, no. 2, pp. 143–149, 2023.
- [8] R. Maulana and Eliyani, "View of Diabetes Classification Algorithm Optimization Using Particle Swarm Optimization on Naïve Bayes, C4.5 and Random Forest.pdf," *J. SISFOKOM (Sistem Inf. dan Komputer)*, pp. 499–509, 2025.
- [9] S. Allam, P. Utomo, and D. Kurniawan, "Optimasi Model Particle Swarm Optimization (PSO) Menggunakan SMOTE Untuk Menentukan Penyakit Diabetes Mellitus," *Build. Informatics, Technol. Sci.*, vol. 6, no. 4, pp. 2659–2671, 2025, doi: 10.47065/bits.v6i4.7111.
- [10] A. A. G. A. Pranandita and I. M. Widiartha, "View of Optimasi Metode Support Vector Machine (SVM) Menggunakan Particle Swarm Optimization pada Permasalahan Klasifikasi Diabetes.pdf," *JNATIA, J. Nas. Teknol. Inf. dan Apl.*, vol. 3, pp. 879–888, 2025.
- [11] S. Fitriani, E. Budiman, M. Fadli, M. Surono, and H. Sulistiani, "Optimalisasi Metode Random Forest menggunakan Particle Swarm Optimization dalam Prediksi Prestasi Mahasiswa," *Pros. Semin. Nas. Teknol. Komput. dan Sains*, vol. 3, no. 1, pp. 406–415, 2025, [Online]. Available: <https://prosiding.seminars.id/sainteks>
- [12] E. E. Ogheneovo and P. A. Nlerum, "Iterative Dichotomizer 3 (ID3) Decision Tree: A Machine Learning Algorithm for Data Classification and Predictive Analysis," *Int. J. Adv. Eng. Res. Sci.*, vol. 7, no. 4, pp. 514–521, 2020, doi: 10.22161/ijaers.74.60.
- [13] Y. Meraihi, A. B. Gabis, S. Mirjalili, A. Ramdane-Cherif, and F. E. Alsaadi, *Machine Learning-Based Research for COVID-19 Detection, Diagnosis, and Prediction: A Survey*, vol. 3, no. 4. Springer Nature Singapore, 2022. doi: 10.1007/s42979-022-01184-z.
- [14] K. Adhatrao, A. Gaykar, A. Dhawan, R. Jha, and V. Honrao, "Predicting Students' Performance Using ID3 and C4.5 Classification Algorithms," *Int. J. Data Min. Knowl. Manag. Process*, vol. 3, no. 5, pp. 39–52, 2013, doi: 10.5121/ijdkp.2013.3504.
- [15] S. Yang, J. Z. Guo, and J. W. Jin, "An improved Id3 algorithm for medical data classification," *Comput. Electr. Eng.*, vol. 65, pp. 474–487, 2018, doi: 10.1016/j.compeleceng.2017.08.005.
- [16] M. Abdar, S. R. N. Kalhori, T. Sutikno, I. M. I. Subroto, and G. Arji, "Comparing performance of data mining algorithms in prediction heart diseases," *Int. J. Electr. Comput. Eng.*, vol. 5, no. 6, pp. 1569–1576, 2015, doi: 10.11591/ijece.v5i6.pp1569-1576.
- [17] J. W. Chung, W. J. Kim, S. Beom Choi, J. S. Park, and D. W. Kim, "Screening for pre-diabetes using support vector machine model," *2014 36th Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBC 2014*, vol. 2014, pp. 2472–2475, 2014, doi: 10.1109/EMBC.2014.6944123.
- [18] D. T. Wilujeng, M. Fatekurohman, and I. M. Tirta, "Analisis Resiko Kredit Perbankan Menggunakan Algoritma K-Nearest Neighbor dan Nearest Weighted K-Nearest Neighbor," *Indones. J. Appl. Stat.*, vol. 5, no. 2, pp. 142–148, 2023.
- [19] A. D. Putri, F. Sholekhah, and E. Dadynata, "The Application of C4.5 Decision Tree Algorithm for Predicting the Survival Rate of Thyroid Cancer Patients," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 4, no. October, pp. 1485–1495, 2024.