

Hibridisasi Metode Fuzzy K-Nearest Neighbor Dengan Metode Modified Particle Swarm Optimization Pada Pengklasifikasian Penyakit Tanaman Kedelai

Rimbun Siringoringo

Universitas Methodist Indonesia
ringorbnsrg@gmail.com

Resianta Perangin-angin

Universitas Methodist Indonesia
Resianta88@gmail.com

Abstract — Fuzzy k-Nearest Neighbor (Fk-NN) merupakan salah satu metode klasifikasi yang powerful. Kehadiran konsep fuzzy pada metode ini berhasil meningkatkan kinerjanya pada hampir semua masalah klasifikasi. Kelemahan utama FkNN adalah sulitnya menentukan parameter-parameternya. Parameter tersebut adalah jumlah tetangga (k) dan kekuatan fuzzy (m). Kedua parameter tersebut sangat sensitif. Hal tersebut menyulitkan penentuan nilai m dan k , sehingga mengakibatkan Fk-NN sulit dikontrol karena tidak ada teori-teori atau panduan yang dapat menyimpulkan berapa seharusnya nilai m dan k yang tepat. Penelitian ini menghadirkan Particle Swarm Optimization (PSO) untuk menentukan nilai k dan m yang terbaik. Model yang diusulkan pada penelitian ini diuji pada Dataset Soybean. Data uji tersebut telah distandardisasi oleh UCI Machine Learning Repository yang banyak diterapkan pada masalah klasifikasi. Penerapan MPSO pada penentuan parameter-parameter Fk-NN diharapkan mampu meningkatkan nilai performa klasifikasi. Berdasarkan eksperimen yang telah dilakukan mengindikasikan bahwa model yang ditawarkan pada penelitian ini menghasilkan performa klasifikasi yang lebih baik dibandingkan dengan model Fk-NN saja. Pada bagian akhir dilakukan perbandingan superioritas model penelitian dengan model klasifikasi yang lain seperti IBK dan Decision Tree. Model penelitian ini memiliki tingkat performa yang lebih baik.

Kata kunci : *fuzzy k-nearest neighbor, modified particle swarm optimization, Soybean dataset*

I. PENDAHULUAN

Metode *k-Nearest Neighbor* (k-NN) merupakan metode machine learning yang terpopuler, sederhana dan mudah diimplementasikan. Selain keunggulan tersebut, k-NN memiliki dua kelemahan. Pertama, keberhasilan metode ini tergantung pada jumlah tetangga atau neighbor (k) yang diterapkan, sehingga untuk menghasilkan tingkat akurasi yang tinggi, para peneliti harus mencoba nilai k dengan jumlah yang bervariasi, tentu saja hal ini tidak efektif karena hal tersebut dilakukan secara manual. Hal tersebut dapat tercermin pada penelitian yang dilakukan [3], dengan menerapkan k yang bervariasi, tingkat akurasi terbaik diperoleh pada $k=3$, sementara [5] memperoleh akurasi terbaik pada $k=13$. Kedua, selain ketergantungan pada nilai k , hubungan antara setiap data dengan kelas bersifat kaku (crisp) dimana setiap data hanya memiliki hubungan dengan satu kelas secara eksklusif, sedangkan pada kelas yang lain tidak memiliki hubungan sama sekali.

Berbagai upaya telah dilakukan untuk menghindari sifat kekakuan k-NN. Salah satu upaya untuk meminimalisasi kekakuan tersebut adalah dengan

menggabungkan prinsip logika fuzzy ke dalam k-NN. Penggabungan tersebut menghasilkan metode baru yang dikenal dengan Fuzzy k-Nearest Neighbor atau Fk-NN [11]. Pada Fk-NN hubungan antara data dengan kelas tidak bersifat kaku, setiap kelas dan data memiliki hubungan keanggotaan atau membership dengan tingkatan tertentu. Kekuatan hubungan tersebut membutuhkan parameter fuzzy strength (m). Dibandingkan dengan k-NN, Fk-NN menghasilkan pencapaian tingkat akurasi yang lebih tinggi pada hampir semua masalah klasifikasi [12]. Hal tersebut juga menjadi alasan utama mengapa metode Fk-NN menjadi metode pilihan yang banyak diminati pada berbagai penelitian khususnya.

Faktor kekuatan fuzzy (m) dan faktor jumlah tetangga (k) merupakan faktor-faktor penentu yang bersifat fundamental pada Fk-NN, artinya nilai m dan k berdampak langsung pada akurasi yang dihasilkan oleh model tersebut. Penentuan nilai m dan k seringkali tidak mudah dan sulit dikontrol karena tidak ada teori atau panduan yang menyimpulkan berapa seharusnya nilai m dan k yang tepat [12]. Untuk menjawab permasalahan tersebut dipandang perlu adanya metode lain yang

dapat membantu Fk-NN menemukan nilai m dan k tersebut. Pada penelitian ini penulis menawarkan sebuah pendekatan solusi optimasi parameter (parameter optimization) dalam rangka pemberian nilai m dan k yang adaptif.

Metode Particle Swarm Optimization (PSO) merupakan metode yang penulis terapkan karena beberapa pertimbangan. Pertimbangan pertama, jika dibandingkan dengan algoritma yang sejenis, misalnya Genetic Algorithm (GA), PSO tergolong lebih sederhana karena tidak memiliki banyak prosedur seperti prosedur selection, mutation maupun crossover yang ada pada GA. Pertimbangan kedua, metode PSO telah terbukti dapat mengoptimalkan parameter metode machine learning yang lain. Hal tersebut ditunjukkan melalui berbagai penelitian berikut. PSO sangat cocok dikombinasikan dengan SVM [13], PSO dan Jaringan Syaraf Tiruan [14], PSO dengan Self Organizing Map atau SOM [15]. Hasil yang diperoleh pada model-model tersebut mengindikasikan perbaikan akurasi melalui penerapan bahwa PSO dalam mengoptimasi parameter.

Pada penelitian ini MPSO (Modified Particle Swarm Optimization) yang merupakan varian lain dari PSO digunakan untuk mengoptimasi parameter Fk-NN. Penelitian ini membangun sebuah model untuk mengevaluasi pemberian kredit berbasis klasifikasi Fk-NN dan MPSO, dengan kata lain, pengoptimalan parameter Fk-NN oleh MPSO diharapkan meningkatkan akurasi klasifikasi.

II. PEMBAHASAN

A. Disain Ekperimen

Penelitian ini dilakukan dalam beberapa tahap, meliputi tahap pra-proses data, pemilihan parameter k , m yang optimal menggunakan MPSO dan FKNN. Model klasifikasi FkNN-MPSO ditampilkan pada gambar 1. Prosedur penyelesaian masalah di atas dapat dijelaskan dalam bentuk langkah-langkah berikut ini :

Langkah 1 : Inisialisasi nilai k dan m . Dalam penerapan Fk-NN nilai parameter k dan m harus diinisialisasi terlebih dahulu.

Langkah 2 : Bangkitkan Partikel awal secara random. Nilai k dan m dibangkitkan secara paralel sebanyak 8 partikel.

Langkah 3 : Nilai k dan m di atas digunakan untuk

melatih (*training*) data latih.

Langkah 4 : Setiap partikel dievaluasi berdasarkan nilai *fitness*. Fungsi *fitness* (*fitness function*) merupakan fungsi yang digunakan untuk menghitung *fitness* atau tingkat kebaikan suatu individu untuk bertahan hidup. Pada penelitian ini, fungsi *fitness* disajikan pada persamaan (1)

$$fitness = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (1)$$

Langkah 5 : Update posisi dan kecepatan partikel. Prosedur PSO mengharuskan posisi dan kecepatan setiap partikel.

Langkah 6 : Pada tahap ini kembali dilakukan pengujian model FK-NN untuk menemukan partikel terbaik. Partikel terbaik yang dievaluasi berdasarkan nilai *fitness* pada langkah 4 diterapkan untuk melatih data latih kembali dan dihitung nilai *fitness* nya

Langkah 7 : Perbaharui *personal optimal fitness* (*pf*) dan *personal optimal position* (*pbest*). Sejauh ini semua partikel pada iterasi awal (pertama) telah dievaluasi, iterasi pertama menghasilkan sebuah partikel terbaiknya (*pbest*).

Langkah 8 : Pada tahap ini dievaluasi apakah seluruh iterasi telah selesai dikerjakan. Jika belum akan dilanjutkan pada **langkah 3**, jika telah selesai maka dilanjutkan pada **langkah 10**.

Langkah 9 : Setiap iterasi memiliki partikel terbaik. Pada tahap ini partikel terbaik dari semua iterasi akan dievaluasi untuk menentukan partikel terbaik *global optimal position* (*gbest*).

Langkah 10 : Jika nilai *gbest* pada **langkah 9** merupakan nilai yang diharapkan maka pada saat tersebut telah diperoleh nilai k dan m yang optimal dan dilanjutkan pada **langkah 11**. Jika nilai *gbest* belum sesuai dengan kriteria yang diharapkan maka kembali dibangkitkan populasi baru dengan kembali pada **langkah 4**.

Langkah 11 : Lakukan pengujian klasifikasi data dengan memasukkan data uji.

Langkah 12 : Lakukan *cross validation* untuk mengetahui apakah hasil klasifikasi pada **langkah 11** telah memiliki akurasi paling baik. Jika ya maka seluruh proses selesai, jika tidak maka lakukan evaluasi nilai K. Sampai diperoleh nilai akurasi terkecil

Dataset yang digunakan pada penelitian ini adalah *Soybean*, yang bersumber dari UCI *Machine Learning Repository*. Dataset ini dipilih karena pertimbangan yaitu dataset berdimensi tinggi.

Tabel 1 Deskripsi Data Set

Dataset	Soybean
Tipe atribut	Kategorial dan Numerik
Jumlah Atribut	36
Jumlah instance	683
Missing value	Yes
Jumlah class	19

Dataset *soybean* merupakan dataset dengan karakteristik *missing value*. Hal tersebut ditandai dengan munculnya nilai-nilai atribut yang kosong atau *missing* (diwakili oleh karakter "?"). Jika tidak ditangani, *missing value* dapat mengurangi akurasi hasil klasifikasi. Berikut ini adalah sampel data dengan komponen *missing value*.

Tabel 2 Data dengan Missing Value

Data Ke-	Nilai atribut
679	0,?,?,?,?,?,?,?,?,?,1,0,2,2,?,1,?,?,?,?,?,?,?,?,?,?,?,?,?,?,17
680	0,1,?,0,?,0,0,?,?,?,1,1,0,2,2,0,1,?,1,?,?,?,?,?,?,?,?,3,?,?,?,?,?,1,18
681	2,1,?,0,?,0,0,?,?,?,1,1,0,2,2,0,1,?,1,?,?,?,?,?,?,?,?,3,?,?,?,?,?,1,18
682	0,1,?,0,?,1,3,?,?,?,1,2,1,1,0,1,?,1,?,?,?,?,?,?,?,?,3,?,?,?,?,?,1,18
683	2,1,?,0,?,1,3,?,?,?,1,1,2,1,1,0,1,?,1,?,?,?,?,?,?,?,?,3,?,?,?,?,?,1,18

Pada penelitian ini, *missing value* ditangani atau dihilangkan melalui metode *mean*. Hasil penanganan *missing value* ditampilkan pada sampel data berikut ini.

Tabel 3 Data setelah missing value removal

Data Ke-	Nilai atribut
679	0,0,2,1,0,2,2,1,0,1,0,1,0,2,2,0,1,0,1,0,0,0,0,0,0,0,0,0,0,0,0,17
680	0,1,2,0,0,0,0,1,0,1,1,1,0,2,2,0,1,0,1,0,0,0,0,0,0,0,0,3,0,0,0,0,0,1,18
681	2,1,2,0,0,0,0,1,0,1,1,1,0,2,2,0,1,0,1,0,0,0,0,0,0,0,0,3,0,0,0,0,0,1,18
682	0,1,2,0,0,1,3,1,0,1,1,1,2,1,1,0,1,0,1,0,0,0,0,0,0,0,0,3,0,0,0,0,0,1,18
683	2,1,2,0,0,1,3,1,0,1,1,1,2,1,1,0,1,0,1,0,0,0,0,0,0,0,0,3,0,0,0,0,0,1,18

Dataset *Soybean* termasuk dalam kategori dataset berdimensi tinggi atau *high dimensional dataset*, oleh

karena itu seleksi fitur masih relevan untuk diterapkan pada dataset tersebut. Seleksi fitur didasarkan pada metode Cfs-Subset (*Correlation feture selection-Subset*). Dengan menerapkan seleksi tribut, jumlah atribut dataset *soybean* berhasil direduksi dari 36 atribut (tabel IV) menjadi 24 atribut (tabel V).

Tabel 4 Daftar atribut awal

Atribut
date, plant-stand, precip temp, hail, crop-hist, area-damaged, severity, seed-tmt, germination, plant-growth, leaves, leafspots-halo, leafspots-marg, leafspot-size, leaf-shread, leaf-malf, leaf-mild, stem, lodging, stem-cankers, canker-lesion, fruiting-bodies, external-decay, mycelium, int-discolor, sclerotia, fruit-pods, fruit-spots, seed, mold-growth, seed-discolor, seed-size, shriveling, roots, class

Tabel 5 Daftar atribut akhir

Atribut
date, plant-stand, precip temp, area-damaged, severity, plant-growth, leaves, leafspots-halo, leaf spots-marg, leafspot-size, leaf-malf, leaf-mild, stem, stem-cankers, canker-lesion, fruiting-bodies, int-discolor, fruit-pods, fruit-spots, seed, seed-discolor, roots, class

Metode seleksi data *training* dan data *testing* yang diterapkan pada penelitian ini adalah metode *k-fold cross validation* dengan nilai $k=10$. Dataset *Soybean* terdiri atas 1000 instance (*record*). Dengan menerapkan skema *10-fold cross validation*, masing-masing *fold* terdiri atas 100 data. Pada tabel VI berikut ditampilkan skema partisi data tersebut.

Tabel 6 Partisi data set

Fold	File partisi (.dat)	Jumlah data
1	soybean-10-1	69
2	soybean-10-2	69
3	soybean-10-3	69
4	soybean-10-4	69
5	soybean-10-5	69
6	soybean-10-6	69
7	soybean-10-7	69
8	soybean-10-8	69
9	soybean-10-9	69
10	soybean-10-10	69

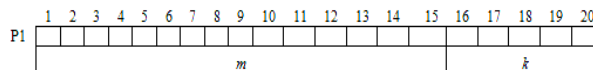
Algoritma MPSO (*Modified Particle Swarm Optimization*) merupakan metode yang *parametric*, dimana penerapannya memerlukan pengaturan parameter. Pada tabel VII berikut diberikan daftar parameter MPSO yang diterapkan.

Tabel 7 Setting parameter MPSO

No	Parameter	Nilai
1	Jumlah partikel (P)	10, 20 dan 40

2	Jumlah <i>swarm</i> (S)	100
3	<i>Velocity</i> maksimum (V_{max})	2
4	Faktor konstriksi (C_1)	2,3
5	Faktor konstriksi (C_2)	2,3

Skema pengkodean (*encoding*) partikel dilakukan dengan membangkitkan bilangan *random* dengan rentang 0 sampai 10 dengan panjang 20 bilangan. Dari 20 bilangan tersebut, bilangan 1 sampai 15 digunakan untuk pembangkitan parameter m , bilangan 16 sampai 20 diterapkan untuk pembangkitan parameter k . Pada gambar 2 ditampilkan skema pengkodean partikel.



Gambar 1 Skema pengkodean partikel

B. Hasil Penelitian

Algoritma PSO diawali dengan proses inisialisasi *swarm*. Pada tabel VIII ditampilkan skema inisialisasi *swarm*. Segmentasi dari tabel tersebut terdiri atas 3 bagian yaitu komposisi partikel, parameter (k,m) dan nilai *fitness*. *Swarm* terdiri atas 10 partikel (P1-P10). Partikel-1 Terdiri atas 20 bilangan *random* (9 0 12 13 1 9 2 4 13 8 3 7 14 14 11 7 4 5 6 3 6). Nilai k,m yang dibangkitkan pada partikel tersebut adalah 21.14, 9.32. Dengan nilai k,m tersebut diperoleh nilai *fitness* 0.65 atau 65%. Pada tabel IX ditampilkan sampel hasil klasifikasi data uji dengan menerapkan nilai k,m yang diperoleh tersebut.

Tabel 8 Inisialisasi SWARM Data set

Partikel	Nilai (k,m)	<i>fitness</i>
Swarm 1		
P1 : 9 0 12 13 19 2 4 13 8 3 7 14 14 11 7 4 5 6 3 6	21.14, 9.32	0.65
P2 : 8 0 2 5 14 11 10 4 12 0 14 9 6 6 7 11 8 6 7 11 9	24.21, 8.38	0.72
P3 : 1 6 12 4 8 8 4 0 11 3 2 12 4 12 12 14 2 5 5 6 14	30.96, 8.83	0.72
P4 : 13 6 11 10 11 12 0 7 6 3 10 11 11 9 8 14 14 14 9 9	21.14, 7.03	0.74
1	33.00, 9.15	0.74
P5 : 14 7 3 8 9 4 12 7 1 6 8 9 0 1 6 14 2 2 4 9 13	19.12, 10.72	0.74
P6 : 3 0 2 5 0 7 6 10 14 2 13 14 2 0 2 13 12 14 13 7 6	27.55, 8.45	0.74
P7 : 1 4 0 5 8 6 13 6 9 9 3 13 14 8 5 14 8 11 3 10 4	26.81, 9.59	0.74
P8 : 13 4 3 0 12 1 9 5 6 12 11 6 9 5 0 8 2 12 9 4 10	26.81, 11.86	0.74
P9 : 4 13 0 10 10 8 3 4 8 6 2 1 13 0 0 2 8 8 6 4 11	29.15, 8.53	0.74
P10 : 11 10 10 5 1 10 13 6 14 12 3 3 5 0 12 2 1 14 4 8 7		

Tabel 9 Hasil klasifikasi data uji

Data Uji ke	Prediksi	Aktual	Hasil
1	1	1	Benar
2	1	1	Benar
3	1	1	Benar
4	1	1	Benar
5	2	2	Benar
6	2	2	Benar

7	2	3	Salah
8	2	3	Salah
9	3	3	Benar
10	2	3	Salah
11	3	3	Benar
12	3	4	Salah
13	4	4	Benar
14	4	4	Benar
15	4	4	Benar
20	7	7	Benar
25	11	11	Benar
30	13	13	Benar
35	3	3	Benar
40	13	6	Salah
45	8	7	Salah
50	11	11	Benar
55	13	13	Benar
60	3	14	Salah
65	15	15	Benar
69	17	17	Benar

Pada *swarm* selanjutnya, masing-masing partikel akan diperbaharui sampai memperoleh nilai *fitness* target yaitu 1.00 (100%). Jika nilai ini tidak dapat dicapai maka nilai tertinggi akan diambil sebagai solusi. Pada tabel X ditampilkan hasil *update* atau pembaharuan partikel, yaitu pembaharuan nilai k,m serta perbaikan nilai *fitness*. Pada tabel tersebut diperoleh nilai *fitness* terbaik 0,84 (84%) dan nilai k,m (3.40, 1.01)

Tabel 10 Update partikel SWARM-59 (FINAL)

Partikel	<i>k,m</i>	<i>fitness</i>
SWARM :59		
P1 : 63 54 66 67 55 63 56 58 67 62 57 61 68 68 65 61 58 59 60 57 60	3.40, 1.01	0.84
P2 : 57 49 51 54 63 60 59 53 61 49 63 58 55 55 56 60 57 55 56 60 58	3.49, 1.11	0.84
P3 : 49 54 60 52 56 56 52 48 59 51 50 60 52 60 60 62 50 53 53 54 62	3.67, 1.13	0.84
P4 : 64 57 62 61 62 63 51 58 57 54 61 62 62 60 59 65 65 65 60 60 52	3.31, 1.04	0.84
P5 : 62 55 51 56 57 52 60 55 49 54 56 57 48 49 54 62 50 50 52 57 61	3.70, 1.14	0.84
P6 : 49 46 48 51 46 53 52 56 60 48 59 60 48 46 48 59 58 60 59 53 52	3.54, 1.21	0.84
P7 : 48 51 47 52 55 53 60 53 56 56 50 60 61 55 52 61 55 58 50 57 51	3.69, 1.15	0.84
P8 : 66 57 56 53 65 54 62 58 59 65 64 59 62 58 53 61 55 65 62 57 63	3.31, 1.05	0.84
P9 : 58 67 54 64 64 62 57 58 62 60 56 55 67 54 54 56 62 62 60 58 65	3.25, 1.05	0.84
P10 : 59 58 58 53 49 58 61 54 62 60 51 51 53 48 60 50 49 62 52 56 55	3.65, 1.13	0.84

Tabel 11 Global Best

S	G_{Best}	<i>k,m</i>	<i>fitness.</i>
59	63 54 66 67 55 63 56 58 67 62 57 61 68 68 65 61 58 59 60 57 60	3.40, 1.01	0,84

Global Best atau G_{Best} adalah partikel dengan nilai *fitness* paling besar dari semua *swarm* yang ada. Pada tabel XI di atas ditampilkan G_{Best} . Tabel tersebut berisi informasi tentang *swarm*, partikel dan nilai *fitness* (*Global Best fitness*). Posisi G_{Best} dicapai pada *swarm* ke-59 (S59) dengan nilai (*k,m*) adalah (3.40, 1.01) dan *fitness* sebesar 0.84 (84%).

Untuk menguji apakah terdapat pengaruh penerapan jumlah partikel terhadap peningkatan nilai *fitness* (dalam hal ini adalah *accuracy*), maka penelitian ini menerapkan jumlah partikel yang bervariasi yaitu 10 partikel, 20 partikel dan 40 partikel. Hasil pengujian diampilkkan pada tabel XII, XIII dan XIV berikut ini. Dari ketiga pengujian tersebut dapa diindikasikan bahwa peningkatan jumlah partikel tidak berpengaruh secara signifikan terhadap perbaikan *fitness*.

Tabel 13 Hasil pengujian 20 Partikel

No. Swarm	<i>k,m</i>	<i>fitness</i>
1	34.13, 7.73	0,74
2	26.11, 8.53	0,74
5	26.11, 8.53	0,75
4	23.09, 7.50	0,77
5	20.70, 6.70	0,78
43	4.92, 1.55	0,80
46	4.58, 1.44	0,81
59	3.59, 1.13	0,83
63	3.35, 1.05	0,62
64	3.30, 1.04	0,52
69	3.19, 1.00	0,49
77	3.00, 0.94	0,10

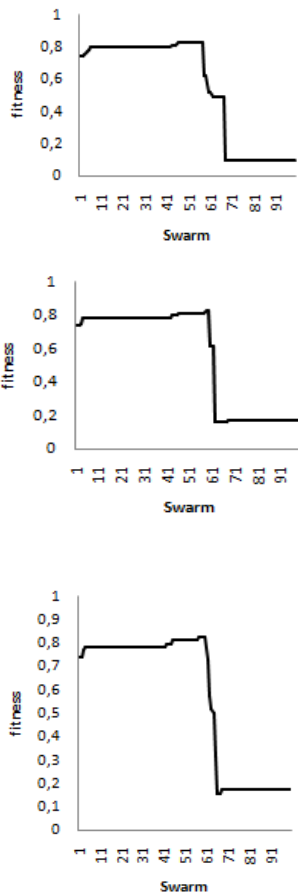
Tabel 12 Hasil pengujian 10 Partikel

No. Swarm	<i>k,m</i>	<i>fitness</i>
1	21.14, 7.03	0,74
2	34.13, 6.98	0,74
3	29.15, 6.2	0,75
4	26.81, 7.73	0,78
5	22.57, 5.23	0,78
6	20.28, 4.82	0,78
43	4.56, 1.33	0,80
46	4.27, 1.25	0,81
58	3.46, 1.03	0,83
60	3.34, 0.99	0,62
62	3.29, 0.98	0,16
68	3.29, 0.98	0,17

Tabel 14 Hasil pengujian 40 Partikel

No. Swarm	k, m	$fitness$
1	19.88, 7.73	0,75
2	30.96, 9.50	0,74
4	23.64, 7.28	0,77
5	21.14, 6.52	0,78
43	4.82, 1.50	0,80
46	4.60, 1.43	0,81
58	3.67, 1.15	0,83
62	3.42, 1.07	0,62
64	3.36, 1.05	0,52
65	3.31, 1.03	0,48
66	3.31, 1.03	0,16
69	3.31, 1.03	0,17

Global Seeking adalah proses penelusuran G_{Best} dimulai dari *swarm* pertama sampai *swarm* terakhir. Pada penelitian ini G_{Best} merupakan *fitness* dengan nilai paling besar, sehingga proses penelusuran akan membentuk grafik menaik sebagaimana ditampilkan pada gambar 2.



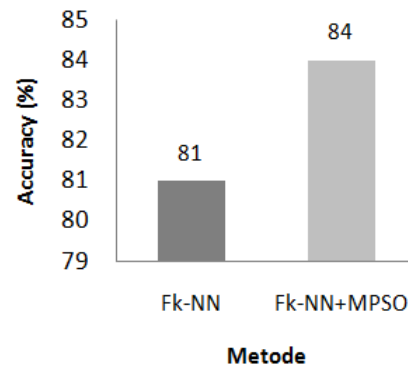
Gambar 2 Grafik *Global Seeking* : a) 10 partikel, b) 20 partikel, c) 40 partikel

C. Pengujian Performa model klasifikasi Fk-NN + MPSO

Penerapan metode MPSO pada algoritma klasifikasi Fk-NN dapat meningkatkan performa dari metode Fk-NN pada klasifikasi penyakit tanaman berdasarkan dataset soybean. Dengan Fk-NN diperoleh tingkat akurasi klasifikasi sebesar 81%, dengan model Fk-NN + MPSO diperoleh performa akurasi klasifikasi sebesar 84%.

Tabel 15 Perbandingan Performa model FK-NN dengan model FK-NN + MPSO

Performa (0 sd 1)	Model	
	Fk-NN	Fk-NN + MPSO (Penelitian ini)
Accuracy	81%	84%



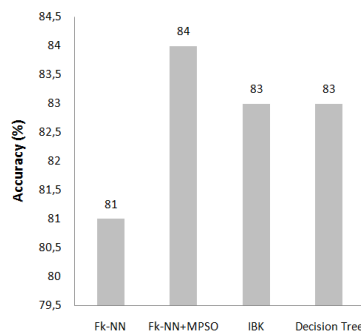
Gambar 3. Grafik Perbandingan Performa Fk-NN dengan Model Fk-NN + MPSO

D. Perbandingan superioritas Model

Untuk memvalidasi superioritas dari model Fk-NN + MPSO dalam memprediksi penyakit tanaman kedelai berbasis dataset *Soybean*, hasil yang diperoleh dengan metode Fk-NN + MPSO dibandingkan dengan metode klasifikasi (*classifier*) lain. Hasil Pengujian menggunakan metode *IBK*, *Decision Tree* disajikan pada tabel berikut. Dalam bentuk grafik, perbandingan tersebut ditampilkan pada gambar 5.

Tabel 17 Perbandingan Superioritas Model

Performa (0 sd. 1)	Metode			
	Fk-NN	IBK	Decision Tree	Fk-NN +MPSO (Penelitian ini)
Accuracy	81%	83%	83%	84%



Gambar 5 Grafik perbandingan superioritas model

III. KESIMPULAN

Penelitian ini menawarkan sebuah model klasifikasi *Fuzzy k-Nearest Neighbor* (Fk-NN) berbasis Algoritma *Modified Particle Swarm Optimization* (MPSO). Penerapan MPSO pada Fk-NN bertujuan untuk meningkatkan performa klasifikasi melalui optimasi parameter ketetangaan (k) dan kekuatan *fuzzy* (m). Penerapan MPSO dapat mengeliminasi aspek subjektivitas dalam penentuan parameter k dan m . Hasil penelitian membuktikan bahwa penerapan MPSO dalam proses klasifikasi berhasil meningkatkan akurasi Fk-NN sehingga berdampak pada peningkatan performa klasifikasi kelayakan kredit. Model Fk-NN dan MPSO dapat bekerja pada data berdimensi tinggi seperti dataset *germancredit*. Penelitian ini juga membandingkan superioritas model Fk-NN + MPSO dengan model klasifikasi lain yaitu *Naive Bayes* dan *Decision Tree*. Hasil pengujian memberikan hasil bahwa model Fk-NN + MPSO memberikan performa yang lebih unggul dibanding ke dua model tersebut.

REFERENSI

- [1] Lee, M-C. *Enterprise Credit Risk Evaluation models: A Review of Current Research Trend*. International Journal of Computer Applications, 44(11) : 0975 – 8887. 2012
- [2] Ghatasheh, A. *Business Analytics using Random Forest Trees for Credit Risk Prediction: A Comparison Study*. International Journal of Advanced Science and Technology, 72 (2014) : 19-30. 2014.
- [3] Abdel Mouna, A. K. *Bank Credit Risk Analysis with K-Nearest Neighbor classifier : case of Tunisian Banks*. Accounting and Management Information Systems, 14 (1) : 79-106. 2005
- [4] Rahman, M.M., Ahmed, S. & Shuvo, M.H. *Nearest Neighbor Classifier Method for Making Loan Decision in Commercial Bank*. IJ. Intelligent Systems and Applications, 4 (8) : 60-68. 2014
- [5] Kurama, O., Luukka, P. & Collan, K. *Credit Analysis Using a Combination of Fuzzy Robust PCA and a Classification Algorithm*. Advances in Intelligent Systems and Computing-Springer, 3(15) : 19-29. 2015.
- [6] Mortezaipour, R. & Afzali, M. *Assessment of Customer Credit through Combined Clustering of Artificial Neural Networks, Genetics Algorithm and Bayesian Probabilities*. International Journal of Computer Science and Information Security, 11(12) : 1-5. 2013.
- [7] CHEN, Q., XUE, H.F. & YAN, L. CREDIT RISK ASSESSMENT BASED ON POTENTIAL SUPPORT VECTOR MACHINE. *INTERNATIONAL CONFERENCE ON NATURAL COMPUTATION (ICNC)* : pp. 1-25. 2011.
- [8] Karimi, A. *Credit Risk Modeling for Commercial Banks*. International Journal of Academic Research in Accounting, Finance and Management Sciences, 4 (3) : 1-6. 2014.
- [9] Kamaloo, E. & Abadeh, M.S. *Credit Risk Prediction Using Fuzzy Immune Learning*. Advances in Fuzzy Systems-Hindawi, 3 (2014) : 1-12 . 201
- [10] Takyar, S.M.T., Nashtaei, R.A. & Chirani, E. *The comparison of credit Risk between Artificial Neural Network and Logistic regression Models in Tose-Taavon Bank in Guilan*. International Journal of Applied Operational Research, 5(1) : 63-72. 2014.
- [11] Rosyid, H., Prasetyo, E. & Agustin, S. *Perbaikan Akurasi Fuzzy K-Nearest Neighbor In Every Class Menggunakan Fungsi Kernel*. *Prosiding Seminar Nasional Teknologi Informasi dan Multimedia 2013* : pp. 13-18. 2013.
- [12] Danenas, P. & Garsva, G. *Credit risk evaluation modeling using evolutionary linear SVM classifiers and sliding window approach*. Proceeding of International Conference on Computational Science-ICCS : pp. 1324 – 1333. 2012.
- [13] Li, S., Zhu, Y., Xu, C. & Zhou, Z. *Study of Personal Credit Evaluation Method Based on PSO-RBF Neural Network Mode*. American Journal of Industrial and Business Management, 3(2013) : 429-434. 2013.
- [14] O'Neill, M. & Brabazon, A. *Self-Organizing Swarm (soswarm) for Financial Credit-Risk Assessment*. Proceeding on 2008 IEEE Congress on Evolutionary Computation : pp. 3087 – 3093. 2008.
- [15] Keller, M. James, Michael R Gray, James A. Givens. *A Fuzzy K-Nearest Neighbor*. IEEE Transactions On Sistem and Cybernetics, 15(4) : 1-8. 1985.
- [16] Jacobcoca, M., Maca, P. & Pech, P. *A comparison of selected modifications of the particle swarm optimization algorithm*. Journal of Applied Mathematics., 14(2014) : 10-15. 2014.
- [17] Guo, H. & He, J. *A modified particle swarm optimization algorithm*. Journal of Computer Science 10(2) : 341-346. 2013.
- [18] Yang, C.H., Hsiao, C-H. & Chuang, L-Y. *Linearly decreasing weight particle swarm optimization with accelerated strategy for data clustering*. International Journal of Computer Science, 37(3) : 3-9. 2010
- [19] Han, J., Kamber, M. & Pei, J. *Data mining techniques and concepts*. Morgan Kaufman publisher. Watham : USA. 2012
- [20] Ramya, R., S. *Analysis of feature selection techniques in credit risk assessmen*. *Proceedings of International Conference on Advanced Computing and Communication Systems (ICACCS - 2015)* : pp. 1-6. 2015