

IMPLEMENTASI ALGORITMA *PARTICLE SWARM OPTIMIZATION*(PSO) DAN *K-NEAREST NEIGHBOR*(K-NN) DALAM MEMPREDIKSI KEBERHASILAN ANAK SMK MENDAPATKAN KERJA

Indra Lina Putra

Politeknik Balekambang Jepara, Email: indra.lina.putra95@gmail.com

ABSTRAK

K-NN merupakan suatu metode supervised yang bertujuan mendapatkan pola baru suatu data dengan menghubungkan pola data sebelumnya dengan yang baru untuk mengklasifikasikan data ke dalam beberapa macam kelas. Kelebihan algoritma K-NN diantaranya sederhana, efektif diterapkan pada data dengan jumlah besar, tangguh terhadap suatu data latih yang noise serta mudah diaplikasikan. Selain kelebihan yang dimiliki K-NN juga terdapat kelemahan yaitu kurang optimal dalam menentukan nilai K. Pada penelitian sebelumnya, penentuan nilai k dilakukan dengan metode try and error, dimana dalam penentuan K nya dilakukan berulang-ulang dan dicoba sampai mendapatkan hasil kinerja yang terbaik. Untuk memperbaiki kinerja K-NN dilakukan perbaikan dengan metode optimasi yaitu PSO. Metode PSO memiliki beberapa kelebihan yaitu sedikit parameter, mudah diterapkan, konvergensi yang cepat, dan sederhana sehingga PSO banyak diterapkan pada optimasi fungsi, optimasi metode konvensional dan klasifikasi pola. Dari eksperimen penelitian yang dilakukan pada metode K-NN dengan menggunakan K=1,2,3,4,5,6,7,8,9,10 dengan perhitungan 10 folds cross validasi didapatkan nilai akurasi tertinggi pada K=7 yaitu sebesar 93.50%, yang dicari dengan metode try and error, selanjutnya pada metode PSOK-NN telah dihitung dan berhenti pada iterasi ke-2 dan didapatkan nilai K optimalnya adalah 7. Jadi dapat disimpulkan bahwa hasil dari metode PSOK-NN dapat dibuktikan bahwa nilai tersebut adalah yang optimal karena pada eksperiment metode K-NN juga di dapatkan nilai akurasi tertingginya adalah pada K=7.

Keywords: *klasifikasi, K-Nearest Neighbor (K-NN), Particle Swarm Optimization (PSO), Penentuan nilai K pada K-NN.*

1. PENDAHULUAN

Pendidikan merupakan usaha sadar yang terencana untuk mewujudkan suasana belajar dan proses pembelajaran agar peserta didik secara aktif mengembangkan potensi dirinya untuk memiliki kekuatan spiritual keagamaan, pengendalian diri, kepribadian, kecerdasan, akhlak mulia dan ketrampilan yang diperlukan dirinya, masyarakat, bangsa dan negara [1].

Salah satu jenjang pendidikan yang jadi pilihan akhir-akhir saat ini adalah Sekolah Menengah Kejuruan(SMK). SMK merupakan salah satu bentuk satuan pendidikan formal yang menyelenggarakan pendidikan kejuruan pada jenjang pendidikan menengah sebagai lanjutan

dari SMP, MTs, atau bentuk lain yang sederajat. Sekolah di jenjang pendidikan dan jenis kejuruan dapat bernama Sekolah Menengah Kejuruan (SMK) atau Madrasah Aliyah Kejuruan (MAK), atau bentuk lain yang sederajat (Undang-undang Sisdiknas Nomor 20 Tahun 2003). SMK memiliki banyak program keahlian. Program keahlian yang dilaksanakan di SMK menyesuaikan dengan kebutuhan dunia kerja yang ada. Pendidikan kejuruan adalah pendidikan menengah yang mempersiapkan peserta didik terutama agar siap bekerja dalam bidang tertentu.

Pada instansi SMK banyak sekali data lulusan siswa-siswi dimana data tersebut belum diolah secara baik, Untuk mengolah data yang kemudian menjadi informasi yang bermanfaat

maka diperlukan teknik data mining. Data mining merupakan sebuah teknologi yang dapat memproses data dalam volume besar yang digunakan oleh perusahaan untuk mengubah data mentah menjadi informasi yang berguna untuk membuat suatu keputusan bisnis yang sangat penting. Pada dasarnya data mining mempunyai 7 fungsi yaitu *Description*, *Classification*, *Clustering*, *Association*, *Sequencing*, *Forecasting*, dan *Prediction*. Dalam fungsi *Classification* merupakan suatu pekerjaan menilai objek data untuk memasukkannya ke dalam kelas tertentu dari sejumlah kelas yang tersedia [2].

Dari beberapa metode klasifikasi, Salah satunya Algoritma K-Nearest Neighbor (K-NN). K-NN merupakan sebuah metode untuk melakukan klasifikasi terhadap obyek baru berdasarkan (K) tetangga terdekatnya. K-NN termasuk algoritma *supervised learning*, dimana hasil dari *query instance* yang baru, diklasifikasikan berdasarkan mayoritas dari kategori pada K-NN. Kelas yang paling banyak muncul yang akan menjadi kelas hasil klasifikasi [3].

Metode K-NN memiliki kelebihan yaitu efektif diterapkan pada data jumlah yang besar dan tangguh terhadap suatu data latih yang *noise* yang merupakan sebuah data yang memiliki *range* nilai paling jauh dibanding data lainnya [4]. Selain kelebihan yang dimiliki, K-NN juga mempunyai kekurangan yaitu kurangnya optimal dalam menentukan nilai *k* karena menggunakan metode *try and error* [5]. Jadi metode *try and error* pada penentuan nilai K pada K-NN merupakan suatu pengelolaan data yang dilakukan berulang-ulang dengan menentukan parameter K sampai hingga didapatkan hasil yang baik, waktu yang diperlukan dalam pengelolaan perhitungan pencarian K optimal tersebut juga membutuhkan waktu yang cukup lama. Sehingga penentuan K optimal dengan menggunakan metode *try and error* menjadi kelemahan pada metode K-NN.

Untuk memperbaiki kekurangan metode tersebut, diperlukan sebuah solusi perbaikan untuk mengoptimalkan kinerja K-NN. Salah satu solusi yang bisa diterapkan adalah dengan melakukan optimasi dengan menggunakan metode *Particle Swarm Optimization* (PSO). Di metode PSO setiap solusi ada dalam ruang

pencarian dipandang sebagai partikel yang mana setiap partikel memiliki nilai *fitness* untuk dioptimalkan, dan memiliki kecepatan untuk perpindahan partikel [6]. Metode PSO memiliki beberapa kelebihan yaitu sedikit parameter, mudah diterapkan, konvergensi yang cepat, dan sederhana sehingga PSO banyak diterapkan pada optimasi fungsi, optimasi metode konvensional dan klasifikasi pola [7].

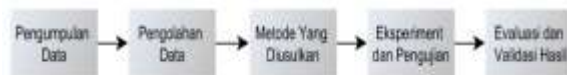
Adapun penelitian tentang prediksi yang menggunakan algoritma K-Nearest Neighbor berbasis *Particle Swarm Optimization*, yaitu peneliti yang dilakukan oleh Nursetia wati pada tahun 2021, yang berjudul prediksi kelulusan mahasiswa menggunakan K-Nearest Neighbor berbasis *Particle Swarm Optimization* Dengan menggunakan teknik *Fold Cross Validation* pada algoritma K-Nearest Neighbor. Dari penelitian tersebut menghasilkan akurasi tertinggi 88,58 pada nilai $k = 14$. Pengujian selanjutnya menggunakan algoritma *Particle Swarm Optimization* untuk mendapatkan akurasi tertinggi pada populasi size = 10 dengan akurasi 89,14% [8].

Selanjutnya, penelitian yang dilakukan oleh Waris Yunus pada tahun 2018, yang berjudul Algoritma K-Nearest Neighbor Berbasis *Particle Swarm Optimization* Untuk Prediksi Penyakit Ginjal Kronik. setelah dilakukan pengujian diperoleh hasil bahwa dengan optimasi menggunakan PSO yang diterapkan pada metode K-NN, nilai akurasi, presisi, dapat meningkat cukup signifikan sehingga metode ini dapat diimplementasikan dalam mengklasifikasi dan memprediksi penyakit ginjal kronik. Dari hasil penelitian ini, diperoleh model terbaik pada K-Nearest Neighbor pada parameter $K = 1$, dengan tingkat akurasi tertinggi yaitu 78.75 %, sedangkan dengan menambahkan metode PSO didapatkan hasil dengan tingkat akurasi 97.25 %. Dari Tingkat akurasi yang diperoleh, metode K-NN+PSO terbukti lebih baik digunakan untuk memprediksi penyakit ginjal kronik [4].

Dari uraian diatas maka tujuan penelitian ini adalah untuk meningkatkan kinerja algoritma K-NN dengan menerapkan metode PSO untuk mengatasi permasalahan K-NN dalam menentukan nilai K yang dilakukan secara *try and error*, yang kemudian digunakan untuk memprediksi keberhasilan anak SMK dalam mendapatkan kerja atau tidak.

2. METODE PENELITIAN

Dalam penelitian ini terdapat lima langkah diantaranya yaitu pengumpulan data, pengolahan data, metode yang diusulkan, eksperimen dan pengujian metode, evaluasi dan validasi hasil. Berikut gambar langkah-langkah metode penelitiannya.



Gambar 2.1 Langkah-langkah metode penelitian

1. Pengumpulan data

Pada penelitian ini, teknik pengumpulan data yang di gunakan adalah teknik pengamatan/*observasi* dan teknik dokumentasi, di mana data didapatkan dengan cara pengamatan dan peristiwa yang sudah berlalu. Data yang di gunakan pada penelitian ini adalah data dari SMK PGRI 2 Kudus yaitu data lulusan tahun pelajaran 2019/2020. Di SMK PGRI 2 Kudus terdapat beberapa jurusan yaitu Akutansi dan keuangan lembaga(AKL), Bisnis Daring dan Pemasaran(BDP), Tata Boga(TB), Usaha Perjalanan Wisata(UPW), dan Teknik komputer dan jaringan(TKJ). Dari data yang telah di kumpulkan dari ke lima jurusan tersebut pada tahun pelajaran 2019/2020 sebanyak 200 anak. Pada dataset yang diambil terdapat 10 atribut/variable yang di gunakan. Variabel tersebut di bagi menjadi dua kategori yaitu variabel pemprediksi yaitu yang dijadikan sebagai penentu prediksi keberhasilan anak mendapatkan kerja atau tidak dan variabel tujuan yaitu variabel yang di jadikan sebagai hasil prediksi berhasil mendapatkan kerja atau tidak. Variabel pemprediksi terdiri dari nama, nilai UN, keahlian/ketrampilan, sikap/kepribadian, keaktifan, prestasi, komunikatif, penampilan/kerapian, kerjasama DU/DI. Sedangkan variabel tujuannya adalah berhasil/tidak.

2. Pengolahan data

Setelah melakukan pengumpulan data, tahap selanjutnya yaitu pengolahan data. Pada tahap ini juga meliputi beberapa tahap, berikut tahapan pengolahan data:

a. Tahap pembersihan data

Pada tahap pembersihan data, akan di jelaskan di mana jika ada record/objek yang tidak sesuai dengan kriteria maka akan di hapus. Pada penelitian ini, dataset yang diambil dari SMK PGRI 2 Kudus tidak ada data yang variabel/atributnya kosong, jadi tidak ada data yang dihapus, maka untuk jumlah data yang di gunakan tetap sejumlah 200 data.

b. Tahap menentukan atribut

Pada tahap menentukan atribut, akan di jelaskan untuk memperoleh dataset dengan jumlah atribut yang lebih sedikit tetapi bersifat informative, jadi dimana jika ada atribut yang tidak memberikan bobot pada hasil akhir maka akan dihilangkan.

c. Tahap integrasi data

Pada tahap integrasi data akan di jelaskan untuk meningkatkan akurasi dan efesiensi algoritma. Data yang digunakan dalam penelitian ini sifatnya kategorikal. Untuk pengujian dataset akan diujikan dengan menggunakan *RapidMiner*.

3. Metode yang di usulkan

Pada Penelitian ini mengusulkan metode PSO untuk mengatasi permasalahan algoritma *k*-NN dalam menentukan nilai *k*. Metode usulan ini diberi nama *particle swarm optimization k-Nearest Neighbor* (PSOk-NN). Berikut langkah-langkah metode yang diusulkan:

a. Inisialisasi parameter PSO

b. Tentukan jumlah partikel dan nilai masing-masing partikelnya secara random.

c. Proses algoritma(*k*-NN). Pada tahap ini merupakan pencarian nilai *error rate* fungsi tujuan untuk setiap partikel yang di tentukan secara random pada langkah sebelumnya.

d. Mencari nilai setiap *fitness* pada setiap partikel.

e. Mencari nilai P_{best} dan G_{best} .

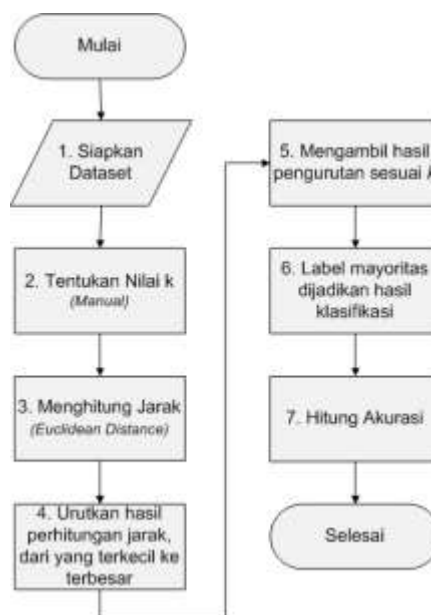
- f. Lihat hasilnya, apakah kondisi saat ini sudah terpenuhi/ konvergen atau belum, dilihat pada hasil nilai P_{best} dan G_{best} nya, jika belum maka akan dilanjutkan iterasi selanjutnya, untuk membangkitkan solusi baru dengan PSO, tetapi jika sudah terpenuhi atau konvergen maka proses selesai dan nilai K optimasi PSO telah di temukan.
 - g. Jika belum konvergen, maka tentukan kecepatan partikel menggunakan (fungsi 2.4)
 - h. Update posisi partikel dengan cara $X_j(i) = V_j(i) + X_j(i-1)$
 - i. Mencari nilai *fitness* lagi pada setiap partikelnya.
 - j. Mencari nilai P_{best} dan G_{best} .
 - k. Lihat hasilnya, apakah kondisi saat ini sudah terpenuhi/ konvergen atau belum, jika belum maka akan dilanjutkan iterasi selanjutnya, untuk membangkitkan solusi baru dengan PSO, tetapi jika sudah terpenuhi atau konvergen maka proses selesai dan nilai K optimasi PSO telah di temukan.
4. Eksperiment dan Pengujian Metode
Pada tahap eksperiment dan pengujian ini akan di jelaskan tentang eksperiment yang di lakukan. Pada penelitian ini menggunakan dataset dari sekolah SMK PGRI 2 Kudus tahun pelajaran 2019/2020, kemudian dataset tersebut akan diuji dengan metode yang telah di usulkan,
 5. Evaluasi Hasil Eksperimen
Evaluasi hasil eksperimen menggunakan evaluasi nilai akurasi. Setelah akurasi metode yang diperbandingkan diperoleh, selanjutnya dilakukan *T-test* untuk mengetahui perbedaan yang signifikan diantara metode yang diperbandingkan yaitu metode K-NN dan metode PSOK-NN. Dari hasil *T-test* tersebut dapat diketahui metode terbaik dari metode yang diperbandingkan.

3. HASIL DAN PEMBAHASAN

a. Pada bagian ini akan dijelaskan eksperimen kedua metode yaitu metode K-NN dan metode PSOK-NN, dimana pada penelitian

ini akan digunakan $K=1,2,3,4,5,6,7,8,9,10$ dengan perhitungan 10 *folds* cross validasi. Selanjutnya untuk metode PSOK-NN akan mencari K optimal yang digunakan untuk memperbaiki kelemahan metode K-NN dalam penentuan nilai K yang di tentukan dengan metode *try and error*. Selanjutnya akan dilihat pada pengukuran kinerja akurasi pada masing-masing metode dengan ketentuan K yang digunakan pada metode tersebut.

Pada eksperimen metode K-NN akan dijelaskan eksperimen langkah-langkah proses perhitungan metode K-NN, sampai dengan ditemukan hasil evaluasi nilai akurasi, dalam contoh eksperimen metode K-NN, K yang digunakan adalah $K=1,2,3,4,5,6,7,8,9,10$ tetapi dalam contoh eksperimen ini penulis hanya menuliskan jumlah $K=5$ dengan perhitungan 1 *folds* cross validasi saja. Tetapi secara keseluruhan hasil nilai akurasi dengan $K=1,2,3,4,5,6,7,8,9,10$ dengan perhitungan 10 *folds* cross validasi akan di tampilkan pada proses perhitungan K-NN dibawah ini. Berikut langkah-langkah perhitungan metode K-NN serta langkah-langkah proses metode K-NN dengan menggunakan $K=5$.



Gambar 3.1 Langkah-langkah metode K-NN.

1. Siapkan dataset.

Dataset diambil dari SMK PGRI 2 Kudus lulusan tahun pelajaran 2019/2020, sejumlah

sebanyak 200 data, dengan rincian 9 atribut yang terdiri dari 8 atribut nilai dan 1 atribut kategori. bisa lihat di https://bit.ly/dataset_penelitian

2. Tentukan nilai K, pada proses eksperimen ini diambil contoh dengan jumlah nilai K=5;
3. Menghitung jarak data uji dengan data latih dengan rumus *Euclidean Distance*

$$d_i = \sqrt{\sum_{i=1}^p (x_{2i} - x_{1i})^2}$$

Dataset lulusan sekolah SMK PGRI 2 Kudus sebanyak 200 data dibagi menjadi dua, yaitu data uji dan data pelatihan dengan perbandingan 1:9, data uji terdiri dari 20 data dan data pelatihan terdiri dari 180 data. Bisa lihat di https://bit.ly/dataset_penelitian

Sebagai contoh perhitungan data uji yang pertama adalah nilai UN = 69, keahlian = 85, sikap = 90, keaktifan = 80, prestasi = 0, komunikatif = 90, penampilan = 80, kerjasama DU/DI = 80. Dan data pelatihan yang pertama yaitu nilai UN = 70, keahlian = 85, sikap = 85, keaktifan = 80, prestasi = 0, komunikatif = 90, penampilan = 85, kerjasama DU/DI = 90. Berikut contoh perhitungan data uji dengan data pelatihan dapat dihitung dengan:

$$d_i = \sqrt{(69-70)^2 + (85-85)^2 + (90-85)^2 + (80-80)^2 + (0-0)^2 + (90-90)^2 + (80-85)^2 + (80-90)^2} = 12.288$$

Selanjutnya lakukan perhitungan jarak data uji dengan semua data pelatihan. Berikut hasil perhitungan jarak data uji dengan semua data pelatihan.

Tabel 3.1 Hasil perhitungan data uji pertama dengan semua data pelatihan

No.	Jarak
1.	12.228
2.	14.177
3.	19.131
4.	81.142
5.	81.142
6.	12.884
...	...
...	...
179.	8.717
180.	17.436

4. Mengurutkan hasil perhitungan jarak data uji dengan data Pelatihan dari kecil ke besar.

Hasil perhitungan pada langkah ke-3 diurutkan dari jarak terkecil ke besar. Berikut hasil pengurutan perhitungan data uji pertama dengan semua data pelatihan metode K-NN dari jarak terkecil ke besar.

Tabel 3.2 Hasil pengurutan perhitungan data uji pertama dengan semua data pelatihan metode K-NN

No.	Jarak
71.	0.00
76.	0.00
78.	0.00
135.	7.141
89.	7.348
47.	8.124
...	...
...	...
18.	82.638
67.	92.011

5. Ambil hasil pengurutan pada langkah ke-4 dari nilai terkecil sesuai nilai K=5.

Pada langkah ini hasil pengurutan perhitungan jarak data uji dengan data pelatihan diambil dari nilai terkecil sesuai nilai K, sehingga jarak terdekat data uji terhadap data pelatihan sesuai dengan nilai K diperoleh. Berikut diperlihatkan 5 jarak terdekat data uji terhadap data pelatihan.

Tabel 3.3 Jarak terdekat sesuai nilai K Metode K-NN

No.	Jarak	Label
71.	0.00	BERHASIL
76.	0.00	BERHASIL
78.	0.00	BERHASIL
135.	7.141	BERHASIL
89.	7.348	BERHASIL

6. Label mayoritas dijadikan hasil klasifikasi

Label hasil klasifikasi diperoleh dari label mayoritas sesuai nilai K. Dari Tabel 4.6 diperoleh data klasifikasi kategori BERHASIL = 5, dan klasifikasi kategori TIDAK = 0, sehingga hasil klasifikasi dengan K = 5 adalah BERHASIL. Berikut ditampilkan label aktual dan label prediksi hasil klasifikasi data uji metode K-NN.

Tabel 3.4 Label aktual dan label prediksi hasil klasifikasi semua data uji metode K-NN

NO.	Label Aktual	Label Prediksi
1.	BERHASIL	BERHASIL
2.	BERHASIL	BERHASIL
3.	BERHASIL	BERHASIL
4.	BERHASIL	BERHASIL
5.	BERHASIL	BERHASIL
6.	BERHASIL	BERHASIL
7.	BERHASIL	BERHASIL
8.	BERHASIL	BERHASIL
9.	BERHASIL	BERHASIL
10.	BERHASIL	BERHASIL
11.	BERHASIL	BERHASIL
12.	BERHASIL	BERHASIL
13.	BERHASIL	BERHASIL
14.	BERHASIL	BERHASIL
15.	BERHASIL	BERHASIL
16.	BERHASIL	BERHASIL
17.	TIDAK	BERHASIL
18.	BERHASIL	BERHASIL
19.	BERHASIL	BERHASIL
20.	BERHASIL	BERHASIL

7. Hitung akurasi metode K-NN

Dari Tabel 3.4 didapatkan *True positive*(TP) = 19, *True negative*(TN) = 0, *False positive*(FP) = 1, *False negative*(FN) = 0. Terdapat 20 data uji, dengan 2 label yaitu BERHASIL dan TIDAK. Berikut ditampilkan hasil klasifikasi Metode K-NN

Tabel 3.5 Nilai hasil klasifikasi metode K-NN.

Prediksi/Aktual	Berhasil Kerja	Tidak Berhasil	Total
Berhasil Kerja	19 (TP)	1 (FP)	20 (P)
Tidak Berhasil	0 (FN)	0 (TN)	0 (N)

$$\text{Rumus } accuracy = \frac{TP+TN}{P+N}, \text{ maka}$$

$$\text{akurasinya} = \frac{TP+TN}{P+N} = \frac{19+0}{20+0} = 0.95$$

Pada penelitian ini menggunakan dataset privat yang diambil dari SMK PGRI 2 Kudus lulusan tahun ajaran 2019/2020, dengan dataset sebanyak 200 data, dengan data tersebut dengan K=5, menghasilkan akurasi sebesar 95.00%. Hasil tersebut merupakan pada *folds* ke-1, setelah dihitung sampai dengan *folds* ke-10 dan setelah dirata-rata hasil akurasi dengan K=5 sebesar

89.00%. Berikut ditampilkan hasil nilai akurasi k= 1 sampai 10 dengan 10 *folds cross* validasi.

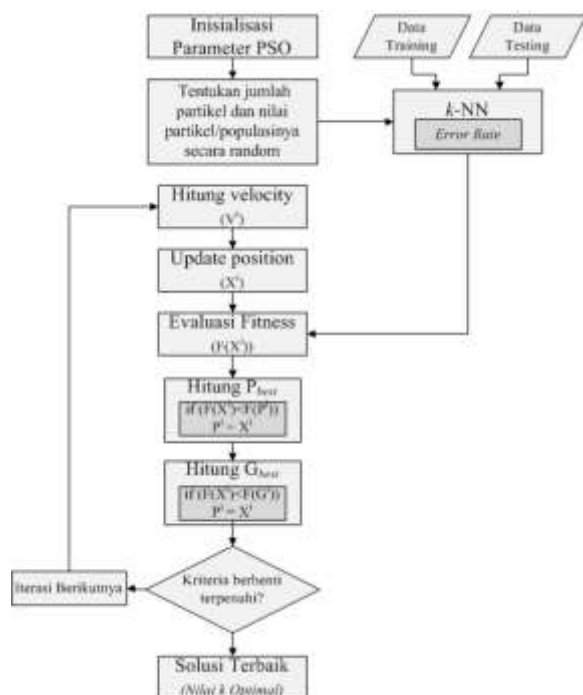
Tabel 3.6 Hasil nilai akurasi k= 1 sampai 10 dengan 10 *folds cross* validasi.

ALGORITMA	AKURASI
K-NN (K=1)	92.00%
K-NN (K =2)	92.00%
K-NN (K =3)	91.50%
K-NN (K =4)	87.00%
K-NN (K =5)	89.00%
K-NN (K =6)	91.50%
K-NN (K =7)	93.50%
K-NN (K =8)	91.00%
K-NN (K =9)	90.50%
K-NN (K =10)	91.10%

Pada tabel 3.6 dapat disimpulkan bahwa pada K= 7 adalah nilai terbaik dari 10 K yang digunakan, yaitu didapat nilai akurasi sebesar 93.50% dengan perhitungan menggunakan 10 *folds cross* validasi.

b. Eksperimen metode PSOK-NN

Pada eksperimen metode PSOK-NN akan dijelaskan langkah-langkah proses perhitungannya sampai dengan ditemukan hasil P_{best} dan G_{best} yang konvergen yang nilainya akan dijadikan K optimal. Berikut langkah-langkah perhitungan metode PSOK-NN.



Gambar 3.2 Langkah-langkah metode PSOK-NN

Pada langkah-langkah metode PSOK-NN ditampilkan tabel bantu yang merupakan penjelasan beberapa simbol beserta keterangan yang digunakan dalam rumus di metode PSOK-NN. Berikut simbol-simbol yang digunakan dalam perhitungan metode PSOK-NN.

Tabel 3.7 Simbol-simbol yang digunakan dalam proses perhitungan metode PSOK-NN

Simbol	Keterangan
w	Bobot inersia 0-1
c_1, c_2	Learning rates
r_1, r_2	Bilangan random yang - berdistribusi dalam interval 0 – 1
f	Fitness
i	Iterasi
j	Nomer partikel
x	Partikel
v	Kecepatan
n	Jumlah partikel

Berikut langkah-langkah perhitungan manual metode PSOK-NN:

1. Inisialisasi parameter PSO

Diketahui parameter PSO yaitu $w= 0.3$, $c_1= 2$, $c_2= 2$, $r_1= 0.5$, $r_2= 0.5$, yang didapatkan secara random.

2. Tentukan jumlah partikel atau yang disimbolkan dengan n , serta tentukan nilainya pada masing-masing posisi partikel atau nilai K nya. Pada percobaan penelitian ini dipilih jumlah $n=25$. Serta tentukan nilai pada masing-masing posisi partikel atau nilai K nya, yang ditentukan secara random. Pada tabel 3.8 ditampilkan jumlah partikel serta nilai masing-masing posisi partikel yang digunakan.

Tabel 3.8 Tabel jumlah partikel serta nilainya dimasing-masing partikel.

No	Partikel	Nilai posisi partikel
1.	$X_1(0)$	35
2.	$X_2(0)$	43
3.	$X_3(0)$	47
4.	$X_4(0)$	45
5.	$X_5(0)$	15
6.	$X_6(0)$	21
7.		
8.		
24.	$X_{24}(0)$	1
25.	$X_{25}(0)$	11

3. Setelah menentukan jumlah n dan nilai masing-masing partikel pada K nya selanjutnya yaitu mencari *error rate* pada setiap partikel. Pencarian *error rate* menggunakan metode K-NN. Berikut hasil pencarian *error rate* di setiap partikelnya.

Tabel 3.9 Nilai *error rate* di setiap partikel

NO.	K	Error Rate %
1.	35	11
2.	43	13
3.	47	11.5
4.	45	13
5.	15	8
6.	21	8
7.	5	9
8.	3	7
9.	23	7
10.	7	5
11.	25	7.5

12.	39	10.5
13.	41	12
14.	49	11.5
15.	17	7.5
16.	13	8
17.	37	13
18.	29	8.5
19.	31	8.5
20.	9	6.5
21.	33	9.5
22.	19	7.5
23.	27	8
24.	1	6.5
25.	11	6.5

4. Evaluasi nilai *fitness* fungsi tujuan untuk setiap partikel $X_j(i)$ untuk $j=1,2,3,4, \dots, 25$, untuk langkah pada iterasi ke-0 ini belum ada perhitungannya, sehingga *error* masih 100%. Berikut ditampilkan nilai fungsi disetiap partikel.

Tabel 3.10 Hasil nilai *fitness* evaluasi disetiap partikel pada iterasi ke-0

No	Partikel	Nilai posisi partikel
1.	$f_{x_1}(0)$	100
2.	$f_{x_2}(0)$	100
3.	$f_{x_3}(0)$	100
4.	$f_{x_4}(0)$	100
5.	$f_{x_5}(0)$	100
6.	$f_{x_6}(0)$	100
7.		
8.		
24.	$f_{x_{24}}(0)$	100
25.	$f_{x_{25}}(0)$	100

5. Tentukan P_{best} dan G_{best} , Untuk mendapatkan nilai posisi partikel pada P_{best} digunakan rumus jika $f_{x_j(i)} < f_{x_j(i-1)}$ maka $P_{best} = x_1(i)$. dan untuk mendapatkan nilai G_{best} , yaitu dengan nilai fungsi tujuan paling kecil/minimum diantara semua partikel untuk semua iterasi sebelumnya, atau disimbolkan dengan $f_{x_j(i)}$. tetapi pada tahap ini P_{best} dan G_{best} belum ada hasilnya, karna belum ada perhitungan sebelumnya. Berikut ditampilkan P_{best} dan G_{best} yang belum diketahui hasilnya.

Tabel 3.11 Hasil P_{best} dan G_{best} pada iterasi ke-0

No	Partikel	Nilai posisi partikel	Nilai G_{best}
1.	$P_{best} x_1$	-	
2.	$P_{best} x_2$	-	
3.	$P_{best} x_3$	-	
4.	$P_{best} x_4$	-	
5.	$P_{best} x_5$	-	-
6.	$P_{best} x_6$	-	
7.			
8.			
24.	$P_{best} x_{24}$	-	
25.	$P_{best} x_{25}$	-	

6. Tentukan kecepatan awal partikel pada iterasi ke-0, yaitu $V_1(0)=V_2(0)=V_3(0)=V_4(0)=V_5(0)=V_6(0) \dots V_{24}(0)=V_{25}(0)=0$. Pada langkah ini kecepatan awal partikel hasilnya masih 0. Karena belum ada perhitungan sebelumnya. Berikut ditampilkan hasil kecepatan awal di setiap masing-masing partikel.

Tabel 3.12 Hasil kecepatan awal partikel pada iterasi ke-0

No	Partikel	Nilai posisi partikel
1.	$V_1(0)$	0
2.	$V_2(0)$	0
3.	$V_3(0)$	0
4.	$V_4(0)$	0
5.	$V_5(0)$	0
6.	$V_6(0)$	0
7.		
8.		
24.	$V_{24}(0)$	0
25.	$V_{25}(0)$	0

7. Pada langkah ini merupakan langkah awal pada iterasi selanjutnya yaitu iterasi ke-1. Hitung masing-masing nilai posisi partikel pada iterasi ke-1, untuk mendapatkan nilai posisi partikel pada iterasi ke-1 digunakan $X_j(i) = X_j(i-1) + V_j(i)$. Partikel ke-1 iterasi ke-1 disimbolkan dengan $X_1(1)$ Sebagai contoh mencari nilai posisi partikel ke-1 iterasi ke-1 yaitu $X_1(1) = 35 + 0 = 35$. Nilai 35 didapat dari nilai posisi partikel ke-1 iterasi ke-0 yang disimbolkan dengan $X_j(i-1)$ dan nilai 0 didapat dari nilai kecepatan partikel ke-1 iterasi ke-0 yang disimbolkan dengan $V_j(i)$. Selanjutnya lakukan perhitungan sampai partikel ke-25.

Berikut ditampilkan semua hasil nilai posisi partikel pada iterasi ke-1.

Tabel 3.13 Hasil nilai posisi partikel pada iterasi ke-1

No	Partikel	Nilai posisi partikel
1.	$X_1(1)$	35
2.	$X_2(1)$	43
3.	$X_3(1)$	47
4.	$X_4(1)$	45
5.	$X_5(1)$	15
6.	$X_6(1)$	21
7.		
8.		
24.	$X_{24}(1)$	1
25.	$X_{25}(1)$	11

8. Evaluasi nilai fungsi tujuan untuk setiap partikel $X_j(i)$ untuk $j=1,2,3,4, \dots, 25$. untuk mendapatkan nilai posisi partikel di setiap *fitness* partikel maka digunakan bantuan *rapidminer* yang digunakan untuk mencari *error rate* disetiap partikelnya. Langkah ini merupakan langkah dari sebagian perhitungan mencari akurasi pada metode K-NN. Pada penelitian ini digunakan sebanyak 200 data, lalu pada *rapidminer* di design untuk perhitungan pencarian akurasi pada metode K-NN. Sebagai contoh mencari nilai posisi partikel pada *fitness* partikel ke-1. Lalu data tersebut dimasukan serta ditentukan K nya, contoh pada $X_j(i)$ nilai K yang digunakan adalah 35, didapat dari $X_1(1)$ partikel ke-1 iterasi ke-1. Setelah di dapatkan hasil akurasinya maka didapat juga hasil pencarian *fitness* pada partikel ke-1 iterasi ke-1. akurasi 100% di kurangi hasil akurasi perhitungan pada *rapidminer* sesuai dengan jumlah K nya tersebut, jadi misal mencari *fitness* partikel ke-1 yaitu 89%, maka $100-89=11$. Jadi pada *fitness* partikel ke-1 iterasi ke-1 didapatkan nilai partikel sebesar 11%. Selanjutnya lakukan perhitungan mencari nilai *fitness* sampai ke-25. Berikut ditampilkan semua hasil nilai posisi partikel pada semua *fitness* partikel.

Tabel 3.14 Hasil nilai evaluasi setiap partikel pada iterasi ke-1

No	Partikel	Nilai posisi partikel
1.	$f_{x_1}(1)$	11
2.	$f_{x_2}(1)$	13
3.	$f_{x_3}(1)$	11.5
4.	$f_{x_4}(1)$	13
5.	$f_{x_5}(1)$	8
6.	$f_{x_6}(1)$	8
7.		
8.		
24.	$f_{x_{24}}(1)$	6.5
25.	$f_{x_{25}}(1)$	6.5

9. Mencari nilai P_{best} dan G_{best} . Untuk mendapatkan nilai posisi partikel pada P_{best} digunakan rumus, jika $f_{x_j(i)} < f_{x_j(i-1)}$ maka $P_{best} = x_1(i)$. Sebagai contoh perhitungan pada P_{best} $x_1 =$ jika $11 < 100$, maka 35. Selanjutnya hitung sampai P_{best} ke-25. Untuk mendapatkan nilai G_{best} , yaitu dengan nilai fungsi tujuan paling kecil/minimum diantara semua partikel untuk semua iterasi sebelumnya atau disimbolkan dengan $f_{x_j(i)}$. Berikut ditampilkan seluruh nilai masing-masing partikel P_{best} dan nilai G_{best} .

Tabel 3.15 Hasil P_{best} dan nilai G_{best} pada iterasi ke-1

No	Partikel	Nilai posisi partikel	Nilai G_{best}
1.	$P_{best} x_1$	35	7
2.	$P_{best} x_2$	43	
3.	$P_{best} x_3$	47	
4.	$P_{best} x_4$	45	
5.	$P_{best} x_5$	15	
6.	$P_{best} x_6$	21	
7.			7
8.			
24.	$P_{best} x_{24}$	1	7
25.	$P_{best} x_{25}$	11	

10. Hitung kecepatan $V_j(i)$, untuk mendapatkan nilai kecepatan pada masing-masing partikel digunakan perhitungan kecepatan partikel ke-1 iterasi ke-1, $V_j(i) = 0.3 \times 0 + 2 \times 0.5 \times (35-35) + 2 \times 0.5 \times (7-35) = -28$. Selanjutnya lakukan perhitungan untuk mencari nilai semua masing-masing partikel. Berikut ditampilkan semua hasil nilai perhitungan mencari kecepatan partikel pada iterasi ke-1.

Tabel 3.16 Hasil kecepatan partikel $V_j(i)$ pada iterasi ke-1

No	Partikel	Nilai posisi partikel
1.	$V_1(1)$	-28
2.	$V_2(1)$	-36
3.	$V_3(1)$	-40
4.	$V_4(1)$	-38
5.	$V_5(1)$	-8
6.	$V_6(1)$	-14
7.		
8.		
24.	$V_{24}(1)$	6
25.	$V_{25}(1)$	-4

11. Pada perhitungan P_{best} dan G_{best} pada iterasi 1 belum konvergen atau nilai antara P_{best} dan G_{best} belum sama, maka akan dilanjutkan ke iterasi ke-2. Untuk mencari nilai posisi partikel pada iterasi ke-2 cara perhitungannya masih sama yaitu dengan $X_j(i) = X_j(i-1) + V_j(i)$. Partikel ke-1 iterasi ke-2 disimbolkan dengan $X_1(2)$. Sebagai contoh mencari nilai posisi partikel ke-1 iterasi ke-2. Jadi $X_1(2) = 35 + (-28) = 7$. Nilai 35 didapat dari nilai posisi partikel ke-1 iterasi ke-1 yang disimbolkan dengan $X_j(i-1)$ dan nilai -28 didapat dari nilai kecepatan partikel ke-1 iterasi ke-1 yang disimbolkan dengan $V_j(i-1)$. Selanjutnya lakukan perhitungan sampai partikel ke-25. Berikut ditampilkan semua hasil nilai posisi partikel pada iterasi ke-2.

Tabel 3.17 Hasil nilai posisi partikel pada iterasi ke-2

No	Partikel	Nilai posisi partikel
1.	$X_1(2)$	7
2.	$X_2(2)$	7
3.	$X_3(2)$	7
4.	$X_4(2)$	7
5.	$X_5(2)$	7
6.	$X_6(2)$	7
7.		
8.		
24.	$X_{24}(2)$	7
25.	$X_{25}(2)$	7

12. Evaluasi nilai *fitness* fungsi tujuan untuk setiap partikel $X_j(i)$ untuk $j=1,2,3,4, \dots, 25$, pada iterasi ke-2 caranya sama seperti pada iterasi ke-1. Sebagai contoh mencari nilai posisi partikel pada *fitness* partikel ke-1 iterasi ke-2. Pada partikel sebelumnya didapatkan nilai partikel yaitu 7, sehingga K yang digunakan adalah 7, selanjutnya dicari dengan rapidminer dengan $k=7$ menghasilkan nilai 95%, maka $100-95=5$, Jadi pada *fitness* partikel ke-1 iterasi ke-2 didapatkan nilai partikel sebesar 5%. Selanjutnya lakukan perhitungan mencari nilai *fitness* sampai ke-25. Berikut ditampilkan semua hasil nilai posisi partikel pada semua *fitness* partikel.

Tabel 3.18 Hasil nilai *fitness* evaluasi disetiap partikel pada iterasi ke-2

No	Partikel	Nilai posisi partikel
1.	$fx_1(2)$	5
2.	$fx_2(2)$	5
3.	$fx_3(2)$	5
4.	$fx_4(2)$	5
5.	$fx_5(2)$	5
6.	$fx_6(2)$	5
7.		
8.		
24.	$fx_{24}(2)$	5
25.	$fx_{25}(2)$	5

13. Tentukan P_{best} dan G_{best} . Untuk mendapatkan nilai posisi partikel pada P_{best} digunakan rumus jika $fx_j(i) < fx_j(i-1)$ maka $P_{best} = x_1(i)$. Sebagai contoh perhitungan pada P_{best} $x_1 =$ jika $5 < 11$, maka 7, selanjutnya hitung sampai P_{best} ke-25. Untuk mendapatkan nilai G_{best} , yaitu dengan nilai fungsi tujuan paling kecil/minimum diantara semua partikel untuk semua iterasi sebelumnya atau disimbolkan dengan $fx_j(i)$. Berikut ditampilkan seluruh nilai masing-masing partikel P_{best} dan nilai G_{best} .

Tabel 3.19 Hasil P_{best} dan nilai G_{best} pada iterasi ke-2

No	Partikel	Nilai posisi partikel	Nilai G_{best}
1.	$P_{best} x_1$	7	

2.	$P_{best} x_2$	7	
3.	$P_{best} x_3$	7	
4.	$P_{best} x_4$	7	
5.	$P_{best} x_5$	7	7
6.	$P_{best} x_6$	7	
7.			
8.			
24.	$P_{best} x_{24}$	7	
25.	$P_{best} x_{25}$	7	

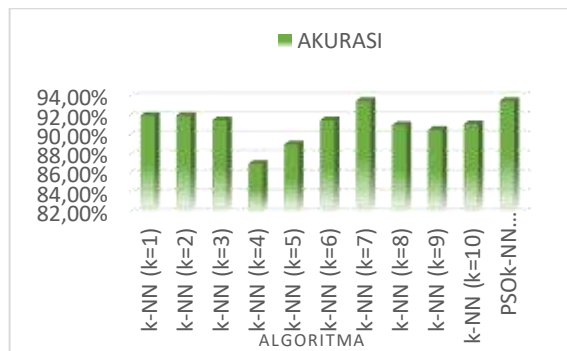
Pada perhitungan eksperimen pada metode PSOK-NN telah berhenti pada iterasi ke-2 karena nilai P_{best} dan G_{best} yang sudah sama atau konvergen maka perhitungan PSOK-NN berhenti, dengan jumlah K optimal yaitu sebesar 7. Jadi dalam eksperimen metode PSOK-NN ini dengan menggunakan dataset sejumlah 200 data yang diambil dari lulusan SMK PGRI 2 Kudus pada tahun pelajaran 2019/2020 telah di temukan nilai optimalnya yaitu sebesar 7. Nilai K optimal yang ditemukan pada metode PSOK-NN selanjutnya akan dicari nilai akurasi.

Dari nilai hasil akurasi metode K-NN pada K=1,2,3,4,5,6,7,8,9,10 dengan perhitungan 10 *folds* cross validasi disetiap K nya sekaligus membuktikan bahwa metode optimasi PSOK-NN dalam pencarian nilai optimasinya adalah 7. Berikut ditampilkan hasil nilai akurasi metode K-NN dari K=1,2,3,4,5,6,7,8,9,10 dengan perhitungan 10 *folds* cross validasi serta ditampilkan nilai optimasi pada metode PSOK-NN.

Tabel 3.20 Hasil nilai akurasi kinerja metode K-NN dan PSOK-NN.

ALGORITMA	AKURASI
K-NN (K=1)	92.00%
K-NN (K =2)	92.00%
K-NN (K =3)	91.50%
K-NN (K =4)	87.00%
K-NN (K =5)	89.00%
K-NN (K =6)	91.50%
K-NN (K =7)	93.50%
K-NN (K =8)	91.00%

K-NN (K =9)	90.50%
K-NN (K =10)	91.10%
PSOK-NN (K=7)	93.50%



Gambar 3.3 Grafik hasil nilai akurasi kinerja metode K-NN dan PSOK-NN.

Pada gambar 3.2 ditampilkan grafik perbandingan hasil nilai akurasi pada metode K-NN dengan K=1,2,3,4,5,6,7,8,9,10 dan metode PSOK-NN dengan K optimasi sebesar K=7, yang dapat disimpulkan bahwa adanya perbedaan nilai akurasi terutama pada K=7 dimana akurasi dengan K tersebut merupakan nilai tertinggi yaitu sebesar 93.50%. Jadi secara keseluruhan dapat disimpulkan bahwa eksperimen pada metode PSOK-NN adalah benar K optimalnya adalah 7.

4. KESIMPULAN

Algoritma K-NN merupakan suatu metode *supervised* yang bertujuan mendapatkan pola baru suatu data dengan menghubungkan pola data sebelumnya dengan yang baru untuk mengklasifikasikan data ke dalam beberapa macam kelas berdasarkan atribut yang ada. Metode K-NN memiliki kelebihan yaitu efektif diterapkan pada data dengan jumlah yang besar, tangguh terhadap suatu data latih yang *noise* yang merupakan sebuah data yang memiliki *range* nilai paling jauh dibanding data lainnya. Selain kelebihan yang dimiliki pada K-NN juga memiliki kekurangan yaitu kurangnya optimal dalam menentukan jumlah nilai K, Selama ini dalam penentuan jumlah nilai K nya ditentukan dengan metode *try and error*, sehingga jika dalam penentuan jumlah K nya kurang tepat dapat

mengakibatkan hasil kinerja yang kurang maksimal.

Untuk memperbaiki kinerja algoritma K-NN dalam penentuan jumlah nilai K diusulkan metode optimasi PSO. Karena metode PSO merupakan sebagai partikel yang mana setiap partikel memiliki nilai *fitness* untuk dioptimalkan, serta memiliki kecepatan untuk perpindahan partikel. Metode PSO juga sedikit parameter, mudah diterapkan, konvergensi yang cepat, dan sederhana. Sehingga PSO banyak diterapkan sebagai metode optimasi. Sehingga pada penelitian ini diusulkan metode PSOK-NN untuk memperbaiki kinerja algoritma K-NN

Pada eksperimen ini peneliti menggunakan dataset sebanyak 200 data yang diambil dari lulusan SMK PGRI 2 Kudus pada tahun ajaran 2019/2020, pada perhitungan K-NN dalam eksperimen yang dituliskan pada bagian hasil dan pembahasan menggunakan nilai K sebanyak 10 yaitu K= 1,2,3,4,5,6,7,8,9,10 dengan perhitungan menggunakan 10 *folds* cross validasi dengan nilai tertinggi pada K= 7 yaitu sebesar 93.50%. selanjutnya untuk perhitungan metode PSOK-NN berhenti pada iterasi ke-2 dengan didapatkan nilai optimalnya sebesar 7. Pada eksperimen ini sama-sama didapatkan nilai K sebesar 7. Perbedaan pada kedua metode tersebut adalah efektif dan kecepatan dalam pencarian nilai K nya. Jadi metode PSOK-NN lebih cepat serta efektif waktu dalam pencarian nilai K nya. Jadi Secara keseluruhan algoritma PSOK-NN dapat memperbaiki permasalahan pada algoritma K-NN yaitu dapat memperbaiki penentuan jumlah nilai K yang dahulu dengan metode *try and error* dengan digantikan dengan optimasi metode PSO-KNN.

5. REFERENSI

- [1] UNDANG-UNDANG REPUBLIK INDONESIA NOMOR 20 TAHUN 2003 TENTANG SISTEM PENDIDIKAN NASIONAL. 2003.
- [2] Mustika *et al.*, 2019. *Data Mining dan Aplikasinya*. Widina Bhakti Persada Bandung.
- [3] L. A. R. Hakim, A. A. Rizal, and D. Ratnasari. 2019. Aplikasi Prediksi Kelulusan Mahasiswa Berbasis K-Nearest Neighbor (K-NN). *JTIM J. Teknol. Inf. dan Multimed.* vol. 1, no. 1, pp. 30–36.
- [4] W. Yunus. 2018. Algoritma K-Nearest Neighbor Berbasis Particle Swarm Optimization Untuk Prediksi Penyakit Ginjal Kronik. *J. Tek. Elektro CosPhi*, vol. 2, no. 2, pp. 51–55.
- [5] S. Mutrofin, A. Izzah, A. Kurniawardhani, and M. Masrur. 2014. Optimasi Teknik Klasifikasi Modified K Nearest Neighbor Menggunakan Algoritma Genetika. *J. GAMMA*, vol. 10, no. 1, pp. 130–134.
- [6] J. Kennedy and R. Eberhart. 2021. Particle Swarm Optimisation. *Stud. Comput. Intell.*, vol. 927, pp. 5–13.
- [7] S. S. Rao. 2009. *Engineering Optimization - Theory and Practice*. canada.
- [8] N. Wati. 2021. Prediksi Kelulusan Mahasiswa Menggunakan K-Nearest Neighbor Berbasis Particle Swarm optimization. *JTII*, vol. 6, no. 2, pp. 230–238.