

EVALUASI KINERJA ALGORITMA K-NN MENGGUNAKAN K-FOLD CROSS VALIDATION PADA DATA DEBITUR KSP GALIH MANUNGGAL

Agung Nugroho¹, Agit Amrullah^{2*}

¹Sistem Informasi, Universitas Amikom Yogyakarta

²Informatika, Universitas Amikom Yogyakarta

email: agit@amikom.ac.id^{2*}

Abstrak: Bisnis simpan pinjam di era sat ini semakin banyak beredar di masyarakat, baik itu online maupun konvensional. Banyak juga perusahaan penyedia jasa simpan pinjam kepada para konsumen atau nasabahnya atau disebut sebagai debitur yang kurang mempertimbangkan kemampuan nasabahnya dalam mengembalikan uang pinjamannya. Sebenarnya banyak algoritma yang mampu dimanfaatkan dalam mengatasi atau memprediksi seorang debitur tersebut nantinya layak untuk diberi pinjaman atau tidak. Salah satunya adalah algoritma *k-nearest neighbour*, dimana algoritma ini nantinya akan melakukan klasifikasi variabel dasar penilaian seorang debitur. Sudah banyak juga algoritma k-nn diterapkan ke sistem pendukung keputusan memprediksi seorang debitur itu layak memperoleh pinjaman atau atau tidak disetujui bank maupun di koperasi simpan pinjam. Akan tetapi, akurasi algoritma k-nn hasilnya berbeda-beda di beberapa kasus yang sudah ada. Kali ini, pada penelitian ini algoritma k-nn akan dicaritahu seberapa besar akurasinya menggunakan *k-fold Cross Validation* jika diterapkan kedalam sample data Debitur di KSP Galih manunggal. Hasil dari *k-fold cross validation* nantinya bisa mengetahui rata-rata keberhasilan dari suatu sistem dengan cara melakukan perulangan dengan mengacak atribut masukan sehingga sistem tersebut teruji untuk beberapa atribut input yang acak. Hasil dari kinerja algoritma k-nearest neighbour yang digunakan pada pendukung keputusan penentuan debitur pada KSP Galih manunggal menggunakan metode *k-fold Cross Validation* memberikan hasil 93%. Dengan demikian implementasi algoritma k-nn memberikan rekomendasi akurasi yang tinggi.

Kata Kunci : algoritma, validasi, klasifikasi, koperasi, data mining

Abstract: *The savings and loan business in this current era is increasingly circulating in society, both online and offline or conventionally. There are also many companies that provide savings and loan services to their consumers or customers or are referred to as debtors who do not consider the ability of their customers to repay their loans. In fact, there are many algorithms that can be used to overcome or predict whether a debtor will be eligible to be given a loan or not. One of them is the k-nn algorithm, where this algorithm will later classify the basic variables of a debtor's assessment. There have also been many k-nn algorithms applied to decision support systems to predict whether a debtor is eligible for a loan or not at a bank or savings and loan cooperative. However, the accuracy of the k-nn algorithm results differ in several existing cases. This time, in this study, the k-nn algorithm will find out how much accuracy it has using the k-fold Cross Validation if it is applied to the Debtor data sample at KSP Galih manunggal. The results of the k-fold cross validation will later be able to find out the average success of a system by repeating it by randomizing the input attributes so that the system is tested for several random input attributes.. The results of the performance of the k-nearest neighbor algorithm used in decision support for determining debtors at KSP Galih manunggal using the k-fold Cross Validation method give a result of 93%. Thus the implementation of the k-nn algorithm provides high accuracy recommendations.*

Keywords : algorithms, testing, data mining, classification, validation

PENDAHULUAN

Menurut penelitian yang dilakukan oleh Agung Nugroho (2019), algoritma K-NN memiliki tingkat akurasi yang tinggi dalam memprediksi kemampuan calon kreditur untuk membayar pinjaman [1]. Namun, penelitian tersebut dilakukan pada data debitur dari bank, sehingga perlu dilakukan penelitian lebih lanjut pada data debitur dari koperasi simpan pinjam seperti KSP Galih Manunggal. Oleh karena itu, penelitian ini bertujuan untuk mengevaluasi kinerja algoritma K-NN pada data debitur KSP Galih Manunggal menggunakan teknik *K-Fold Cross Validation* untuk memastikan hasil prediksi yang akurat dan dapat diterapkan pada data debitur yang baru.

Koperasi Simpan Pinjam (KSP) Galih Manunggal beralamatkan di JL. Raya kaliabu, Salaman, Kaliabu, Ngampel, Kaliabu, Kec. Salaman,

Kabupaten Magelang bergerak dibidang simpan pinjam. Dalam menjalankan bisnisnya terutama pada proses peminjaman dana ke nasabah atau debiturnya KSP Galih Manunggal mempunyai variable-variabel penilaiannya sendiri. Adapun variable penilaian yang digunakan adalah penghasilan perbulan debitur, pekerjaan debitur, nilai jaminan debitur, jumlah tanggungan keluarga debitur, pendidikan terakhir debitur, usia debitur, dan status pernikahan. Dari variabel-variabel penilaian diatas KSP Galih manunggal mempertimbangkan keputusannya dalam memberikan pinjaman kepada debiturnya. Tentunya variabel-variabel yang digunakan mempunyai bobot prioritas yang berbeda-beda.

Ada penelitian terdahulu yang berjudul "Implementasi Algoritma *K-Nearest Neighbor* Dalam Memprediksi Potensi Calon Kreditur Di Ksp Galih Manunggal". Penelitian yang sudah dilakukan

bertujuan membantu KSP Galih Manunggal dalam mengambil keputusan menentukan debitur mana yang layak diberikan pinjaman berdasarkan hasil perhitungan *k*-nearest neighbour terhadap data riwayat debitur lama dan calon debitur berdasarkan nilai kedekatannya [1].

Penelitian ini merupakan lanjutan dari penelitian terdahulunya. Dimana pada penelitian sebelumnya hasil implementasi algoritma *k*-nearest neighbour yang dimanfaatkan untuk memprediksi potensi calon debitur belum menghitung akurasi dari kinerja algoritmanya. Pada penelitian ini akan berfokus mencari tahu seberapa akurat algoritma *k*-nearest neighbour dalam memberikan hasil klasifikasi terhadap dataset yang digunakan. Dengan menggunakan metode evaluasi *k-fold cross validation*, diharapkan mampu mengetahui akurasi algoritma tersebut.

K-fold Cross-Validation adalah proses dimana dataset dipecah menjadi sejumlah *K fold* dan digunakan untuk mengevaluasi kemampuan model saat diberikan data baru. *K* mengacu pada jumlah kelompok sampel data dibagi menjadi. Misalnya, jika ada nilai *k* adalah 5, maka validasi silang dilakukan 5 kali lipat. Setiap lipatan digunakan sebagai set pengujian pada satu titik dalam proses [2].

TINJAUAN PUSTAKA

Ada beberapa jurnal terdahulu yang dijadikan penulis sebagai rujukan dalam penelitian ini. Yang pertama ada penelitian dari Huzain Azisa dkk yang berjudul "Performa Klasifikasi K-NN dan Cross-validation pada Data Pasien Pengidap Penyakit Jantung". Penelitian tersebut membahas tentang penerapan algoritma *k*-NN dan metode crossvalidation pada dataset pasien cardiovascular dan non cardiovascular menggunakan 11 atribut (age, gender, height, dll.) dengan jumlah 1000 record. Proses yang dilakukan yaitu dengan membagi rasio simulasi dataset 20:80, 50:50, dan 80:20, dengan penerapan *k-fold*=10 dan klasifikasi menggunakan algoritma *k*-nn dengan nilai *k*=2 sampai *k*=900. Hasil akhir penelitian tersebut menghasilkan akurasi 82% pada rasio dataset 50:50, dengan presisi recall 82%, dan *f*-measure 80% pada nilai *k*=13. Pada rasio dataset 20:80 menghasilkan akurasi 87%, dengan rincian recall sebanyak 97%, dan *f*-measure 92% pada *k*=3. Pada rasio dataset 80:20 diperoleh akurasi 91%, recall 92%, dan *f*-measure 72% pada *k*=5 [3].

Penelitian terdahulu selanjut ada dari Linda Mardiana dkk yang berjudul Analisis Diskriminan Dengan *K Fold Cross Validation* Untuk Klasifikasi Kualitas Air Di Kota Pontianak. Pada penelitian tersebut menggunakan data primer sebanyak 42 sampel air dikota Pontianak dengan variabel dependen yang diperoleh dari hasil perhitungan indeks pencemaran yang dibagi menjadi 2 kategori yaitu tercemar dan tercemar sedang. Sedangkan variabel independennya adalah fluoride, kesadahan, nitrat, dan DO. Proses penelitian ini adalah dengan

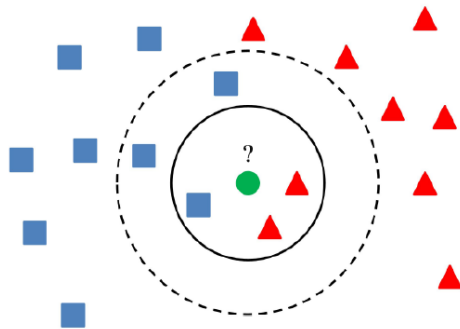
menerapkan metode *k-fold cross validation* terhadap variabel dependen untuk menguji nilai normal multivariat dan kesamaan matriks dari varian-kovarian, kemudian melakukan klasifikasi analisis diskriminan. Dari hasil perhitungan model yang digunakan diperoleh nilai $D2 = -3,500 + 0,087x2$ dengan nilai apparent error rate (APER) terendah 0,21 pada eksperimen kedua [4].

Penelitian terdahulu selanjutnya yang digunakan sebagai rujukan ada dari Ritham Tuntun dkk yang berjudul "Analisis Perbandingan Kinerja Algoritma Klasifikasi dengan Menggunakan Metode *K-Fold Cross Validation*". Penelitian tersebut membahas tentang akurasi dari 2 algoritma data mining klasifikasi yaitu algoritma *k*-nearest neighbour dan algoritma C4.5 dengan memanfaatkan metode *k-fold cross validation*. Data yang digunakan pada penelitian tersebut adalah 150 data public iris dengan membagi 3 klas yaitu iris-setosa, iris-versicolor, dan iris-virginica. Hasil dari evaluasi *k-fold cross validation* terhadap 2 algoritma yang digunakan adalah 95,33% presisi dengan nilai recall sebanyak 95,33% pada evaluasi algoritma *k*-NN. Sedangkan untuk algoritma C4.5 menghasilkan akurasi sebanyak 96%, dengan rincian presisi sebanyak 93,52%, dan recall sebanyak 94,44% [5].

Gambaran tentang perkembangan terbaru dalam penggunaan *k-fold cross validation* dan variasi teknik yang digunakan untuk meningkatkan akurasi, mengatasi masalah ketidakseimbangan kelas, dan mengoptimalkan evaluasi pada data dengan karakteristik khusus masih terus digali dan dimanfaatkan. Berikut ada beberapa jurnal yang digunakan penulis sebagai referensi untuk memecahkan permasalahan yang dihadapi. Penelitian yang dilakukan oleh Jaka Tirta Samudra Komparasi 3 Metode Algoritma Klasifikasi Data Mining Pada Prediksi Kenaikan Jabatan, Pada model penelitian ini menggunakan tiga model dari Naïve Bayes, *K*-Nearest Neighbor, dan Neural Network dengan mengambil dataset langsung dari hasil analisis, untuk itu dilakukan analisis pada setiap aspek untuk mengetahui hasil klasifikasi nilai yang digunakan pada evaluasi menggunakan validasi 5-Fold, 10-Fold, dan Cross Validatio 20-Fold sehingga diperoleh hasil identifikasi pada klasifikasi promosi dengan nilai akurasi tertinggi 76,6%, nilai F1 tertinggi 67,8%, nilai presisi tertinggi sebesar 65,9%, dan nilai recall tertinggi sebesar 76,6% [6].

Algoritma *K-Nearest Neighbour (k-nn)* adalah algoritma data mining klasifikasi yang menghitung jarak nilai kedekatan terpendek untuk mengklasifikasikan suatu dataset baru dengan data set lama untuk mengukur kemiripannya. Tahapan algoritma *k*-nn didapat dari prose pembelajaran (learning) pada data record terdahulu yang sudah dipilih melalui proses klasifikasi digunakan untuk mengklasifikasi data baru dengan hasil yang belum diketahui nilainya. Pada algoritma *k*-nn, data baru akan diklaifikasikan sesuai kedekatan atau kemiripan

terhadap data yang sudah mempunyai pola. Misal pola data sudah ditentukan sebanyak 5 data terdekat, nilai tersebut akan dinyatakan dengan k. Sebagai contoh nilai k=1 maka kasus tersebut hanya akan diklasifikasikan kedalam 1 data pola. Jika nilai k=5 maka data baru akan diklasifikasikan dengan 5 data pola yang mempunyai nilai jarak kemiripan terdekat. Ilustrasi nilai k=1, k=2, dan k=5 bisa digambarkan pada gambar 1 berikut ini[7][8].



Gambar 1. Ilustrasi k-nearest neighbour[7]

Ilustrasi algoritma k-nearest neighbour. Lingkaran hijau adalah sampel yang akan diklasifikasikan, kotak biru dan segitiga merah menggambarkan sampel dari dua kelas yang berbeda di dalam training set. Jika K adalah 3, maka 3 tetangga terdekat (digambarkan dengan lingkaran hitam dengan garis padat) akan menentukan kelas mana yang akan ditugaskan untuk sampel yang dianalisis. Jika K adalah 5, 5 tetangga terdekat akan diperhitungkan (digambarkan dengan lingkaran hitam dengan garis putus-putus)[9].

Secara umum untuk mendefinisikan jarak antara dua objek x dan y, digunakan rumus jarak Euclidean, seperti terlihat pada persamaan (1) berikut ini.

$$D_{xy} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2} \quad (1)[1]$$

Mining classification rule adalah tahapan menentukan suatu kasus kedalam kelas yang sudah ditentukan sebelumnya. Proses klasifikasi dimulai dengan membuat aturan klasifikasi menggunakan sejumlah data sebagai acuan. Dalam proses klasifikasi dibutuhkan data training atau data yang sudah diketahui kelas datanya untuk dijadikan sebagai model pemikiran[10][11]. Selanjutnya menentukan data baru untuk dilakukan pengujian dengan data yang sudah diketahui kelas datanya. Dari hasil pengujian ini akan didapat sebuah nilai yang bisa sebagai acuan prediksi suatu kasus. Proses klasifikasi k-nn bisa dilihat pada ilustrasi gambar berikut.

Proses klasifikasi sendiri terdiri dari beberapa tahapan yaitu ; seleksi data, cleaning data, pembagian kelas, pembobotan nilai atribut, dan dilanjutkan uji coba dataset menggunakan algoritma k-nearest neighbour. Seleksi data bertujuan untuk menentukan dataset mana saja yang akan digunakan sebagai data

sample.[3] Menggunakan rumus probability sampling. Persamaan rumus probability sampling menggunakan perhitungan Slovin, yang ditunjukkan pada persamaan (2) berikut ini;

$$n = \frac{N}{N \cdot e^2 + 1} \quad (2)[1]$$

N= total sample

N=Jumlah populasi data

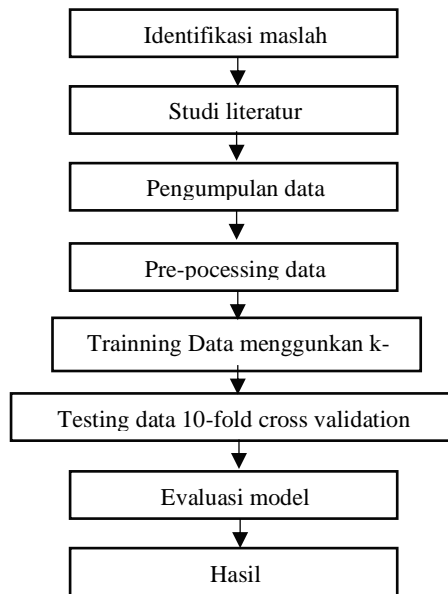
E=toleransi kesalahan(10%)

Cleaning Data dilakukan bertujuan untuk membuang data pada dataset lama yang tidak memberikan pengaruh terhadap hasil seleksi. [4] Proses cleaning data berpedoman pada aturan dan variabel syarat pada suatu kasus. Pembagian kelas bertujuan untuk menentukan data training atau dataset lama untuk ditentukan kelas mana yang cocok sesuai predictornya sesuai nilai komponen pada data predictor. Pembobotan Nilai Atribut dilakukan bertujuan untuk menentukan bobot dari setiap variabel penentu. Pembobotan ini menggunakan metode *Rank Order Centroid* (ROC). Metode ROC memberikan pembobotan pada atribut penilaian sesuai prioritas[12]. Persamaan pembobotan menggunakan metode ROC dirumuskan menggunakan perhitungan seperti pada persamaan (3) berikut ini;

$$W_k = 1/k \sum_{n=1}^k \binom{1}{i} \quad (3)[12]$$

Metode evaluasi *n-fold Cross Validation* digunakan untuk mengetahui nilai rata-rata dari kinerja suatu algoritma. Metode ini akan melakukan pengulangan dengan melakukan pengacakan terhadap atribut/variable penentu dari suatu algoritma untuk memperoleh hasil yang teruji. Metode ini melakukan pembagian terhadap sejumlah data sebanyak n-fold yang sudah ditentukan. Untuk memberikan hasil yang maksimal disarankan membagi dataset menjadi 10-fold cross validation dalam suatu model.[6][13]

Metode *Cross Validation* adalah suatu teknik evaluasi yang digunakan dalam pembangunan model untuk mengevaluasi kinerja model pada data yang tidak terlihat oleh model sebelumnya[14]. Adapun alur penelitian yang disusun dalam penelitian ini digambarkan dalam gambar 2 berikut:



Gambar 2. Alur Penelitian[5]

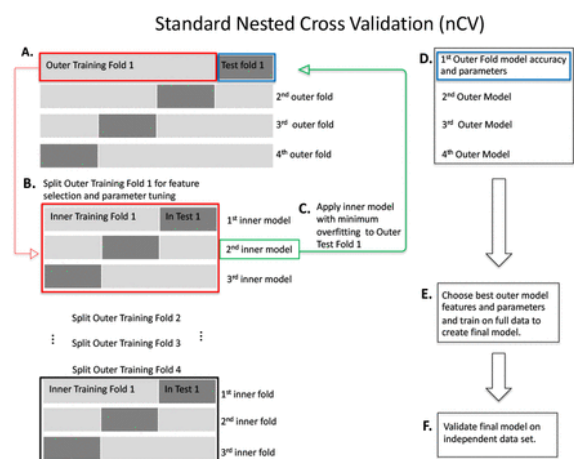
Langkah-langkah untuk melakukan metode *Cross Validation* pada implementasi algoritma *K-Nearest Neighbor (KNN)* untuk memprediksi potensi calon kreditur di KSP Galih Manunggal adalah sebagai berikut:

1. Identifikasi masalah, studi literatur, dan melakukan pengumpulan data untuk persiapan data pelatihan dan data validasi yang akan digunakan. Data pelatihan akan digunakan untuk melatih model KNN dan data validasi akan digunakan untuk mengevaluasi kinerja model.
2. Menentukan parameter yang akan digunakan dalam algoritma KNN, seperti jumlah tetangga terdekat (K), metrik jarak, dan lain-lain.
3. Membagi data pelatihan menjadi beberapa bagian dengan ukuran yang sama. Ukuran bagian yang dipilih dapat bervariasi tergantung pada jumlah total data yang dimiliki. Setelah itu, tentukan data pelatihan dan data validasi yang akan digunakan untuk setiap iterasi.
4. Melatih model KNN menggunakan data pelatihan yang telah dibagi pada setiap iterasi. Setelah itu, gunakan model yang telah dilatih untuk melakukan prediksi pada data validasi.
5. Evaluasi kinerja model pada data validasi dengan menggunakan metrik evaluasi disetiap iterasi.

6. Mengkalkulasi rata-rata kinerja model pada setiap iterasi. Ini akan memberikan estimasi kinerja model yang lebih stabil dan akurat.
7. Tuning parameter - Jika kinerja model masih kurang, maka lakukan tuning parameter seperti mengubah nilai K, atau menggunakan metrik jarak yang berbeda.
8. Setelah kinerja model dianggap cukup baik, gunakan model KNN pada data yang tidak terlihat sebelumnya untuk melakukan prediksi potensi calon kreditur di KSP Galih Manunggal.

METODE

k-fold cross validation adalah cara yang efektif untuk menggabungkan pemilihan fitur dan penyetelan parameter pembelajaran mesin untuk melatih model prediksi yang optimal. Dalam pendekatan *k-fold Cross Validation*, data dibagi menjadi k lipatan luar dan kemudian lipatan dalam dibuat di setiap set pelatihan luar untuk memilih fitur, menyetel parameter, dan melatih model. Menggunakan sarang bagian dalam membatasi kebocoran informasi antara lipatan luar yang dapat menyebabkan pemilihan fitur, dan akibatnya sarang bagian dalam mencegah perkiraan kesalahan *k-fold Cross Validation* lipatan luar yang bias untuk data independen. Namun, *k-fold Cross Validation* bersarang standar sangat intensif secara komputasi karena jumlah pengklasifikasi yang harus dilatih di sarang bagian dalam. Selain itu, kami menunjukkan bahwa *k-fold Cross Validation* bersarang dapat memilih kelebihan fitur yang tidak relevan, yang dapat memengaruhi interpretasi model secara biologis[2]. Berikut adalah alur proses *k-fold Cross Validation* ditunjukkan pada gambar 3 berikut ini.



Gambar 3. Proses *k-fold Cross Validation*[2]

Langkah-langkah *k-fold Cross Validation* :

1. Pisahkan data menjadi lipatan luar dari pasangan data pelatihan dan pengujian (4 lipatan luar dalam ilustrasi ini). Kemudian lakukan hal berikut untuk setiap lipatan

- latihan luar (ilustrasi dimulai dengan Lipatan Latihan Luar 1 (kotak merah, A).
2. Pisahkan lipatan pelatihan luar menjadi lipatan dalam untuk pemilihan fitur dan kemungkinan penyetelan hyperparameter dengan pencarian kisi.
 3. Gunakan model pelatihan dalam terbaik termasuk fitur dan parameter (model dalam ke-2, kotak hijau, untuk ilustrasi berdasarkan overfitting minimum (perbedaan antara akurasi pelatihan dan pengujian) di lipatan dalam untuk menguji pada lipatan uji luar (panah hijau ke kotak biru, Uji Lipat 1).
 4. Simpan model terbaik untuk lipatan luar ini termasuk fitur dan akurasi pengujian. Ulangi B-D untuk sisa lipatan luar.
 5. Memilih model luar terbaik dengan fitur-fiturnya berdasarkan minimum overfitting. Latih data lengkap untuk membuat model final.
 6. Validasi model akhir pada data independen. Metode yang telah dipublikasikan harus ditunjukkan dengan referensi yang sesuai pada bagian daftar pustaka. Apabila terdapat modifikasi yang relevan, maka hal tersebut juga harus dijelaskan[2].

HASIL DAN PEMBAHASAN

Hasil akhir penelitian terdahulu yang berjudul “Implentasi Algoritma K-Nearest Neighbor Dalam Memprediksi Potensi Calon Kreditur Di Ksp Galih Manunggal” diperoleh data sebagai berikut. Hasil transformasi data training disajikan pada table 1 berikut ini.

Tabel 1. Hasil Cleaning Data

Nama Atribut	Tingkat Kepentingan	Bobot
Penghasilan Perbulan	1	0,370
Pekerjaan	2	0,228
Nilai Jaminan	3	0,156
Tanggungan Keluarga	4	0,109
Pendidikan Terakhir	5	0,073
Usia	6	0,044
Status Pernikahan	7	0,020

Setelah diketahui nilai kedekatan antar atribut kasus baru dengan kasus lama satu per-satu. Kemudian dilakukan perhitungan yang pertama, mencari nilai keseluruhan kedekatan kasus tersebut yaitu kasus baru dan kasus lama menggunakan algoritma k-nn menghasilkan nilai 0,49743. Dengan proses perhitungan rinci pada table 2 sebagai berikut:

Tabel 2. Hasil Similarity k-nn Nasabah baru dan lama

Similarity
$\frac{[(0,45 \times 0,370) + (0,54 \times 0,228) + (0,45 \times 0,156) + (0,45 \times 0,109) + (0,52 \times 0,079) + (1 \times 0,044) + (0,18 \times 0,020)]}{(0,370 + 0,228 + 0,156 + 0,109 + 0,073 + 0,044 + 0,020)}$
$\frac{(0,1665 + 0,123 + 0,0702 + 0,04905 + 0,04108 + 0,044 + 0,0036)}{1}$
0,49743

Hasil proses perhitungan kedekatan data testing terhadap semua data training dapat dilihat pada tabel 3 berikut.

Tabel 3. Hasil Nilai Kedekatan Data Testing

Kode DTS	Kode DTR	A	B	C	D	E	F	G	H
TS001	TR001	0,5	0,65	0,5	1	0,75	0,35	1	1
TS001	TR002	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR003	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR004	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR005	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR006	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR007	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR008	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR009	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR010	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR011	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR012	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR013	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR014	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR015	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR016	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR017	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR018	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR019	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR020	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR021	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR022	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR023	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR024	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR025	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR026	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR027	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR028	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR029	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR030	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR031	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR032	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR033	0,5	0,15	0,5	0,5	0,75	0,35	1	0,5
TS001	TR034	0,5	0,15	0,5	0,5	0,75	0,5	0,5	0,75
TS001	TR035	0,75	0,35	1	0,5	0,15	0,5	0,5	0,75
TS001	TR036	0,5	0,65	0,5	1	0,75	0,35	1	1
TS001	TR037	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR038	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR039	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR040	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR041	0,15	0,5	0,5	0,75	0,35	0,5	0,15	0,5
TS001	TR042	0,5	0,15	0,5	0,5	0,75	0,35	0,5	1
TS001	TR043	0,5	0,65	0,5	1	0,75	0,35	0,15	1

Hasil Akhir Perhitungan Jarak Kedekatan Data Testing Dengan Semua Data Training disajikan dalam tabel 4 berikut ini.

Tabel 4. Data Testing Dengan Semua Data Training

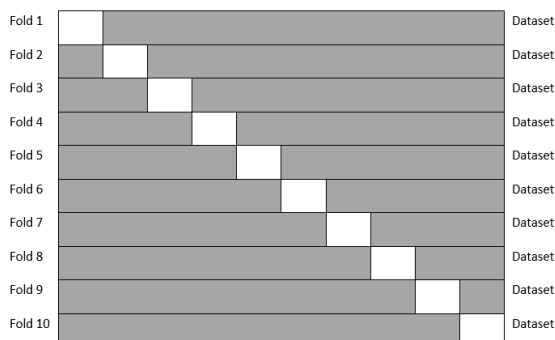
Kode Data Testing	Kode Data Training	Status	Kedekatan
TS001	TR034	Buruk	0,95
TS001	TR035	Buruk	0,92
TS001	TR036	Buruk	0,87
TS001	TR001	Buruk	0,83
TS001	TR038	Baik	0,73
TS001	TR039	Baik	0,71
TS001	TR040	Baik	0,66
TS001	TR041	Baik	0,62
TS001	TR042	Baik	0,59
TS001	TR043	Normal	0,55
TS001	TR012	Normal	0,54
TS001	TR013	Buruk	0,51
TS001	TR014	Normal	0,50
TS001	TR015	Buruk	0,49
TS001	TR016	Baik	0,48
TS001	TR017	Baik	0,46
TS001	TR018	Baik	0,44
TS001	TR019	Baik	0,43
TS001	TR020	Baik	0,41
TS001	TR021	Baik	0,38
TS001	TR022	Buruk	0,35
TS001	TR023	Buruk	0,34
TS001	TR024	Normal	0,31
TS001	TR025	Normal	0,29
TS001	TR026	Buruk	0,28
TS001	TR027	Baik	0,27
TS001	TR037	Baik	0,23
TS001	TR002	Baik	0,21
TS001	TR003	Baik	0,15
TS001	TR004	Baik	0,11
TS001	TR005	Baik	0,07
TS001	TR006	Baik	0,05
TS001	TR007	Normal	0,04
TS001	TR008	Normal	0,03
TS001	TR028	Buruk	0,02
TS001	TR029	Buruk	0,02
TS001	TR030	Normal	0,02
TS001	TR031	Baik	0,01
TS001	TR032	Baik	0,01
TS001	TR033	Baik	0,01

Pembagian Dataset dibagi menjadi 10 partisi diilustrasikan pada gambar 4 sebagai berikut :



Gambar 4. Ilustrasi 10-fold cross Validation

Dalam proses cross validation data akan dibagi dalam n buah partisi dengan ukuran yang sama D1,D2,D3 [15]. Dan selanjutnya proses testing dan training dilakukan sebanyak n kali, seperti yang terlihat seperti ilustrasi gambar 5 berikut ini.



Gamabr 5. 10 k-Fold Validation

Setiap iterasi validasi *k-fold cross validation* terhadap 10 dataset yang sudah dibagi sesuai aturan disajikan pada gambar 6, gambar 7, gmabr 8, gambar 9, gambar 10, gambar 11, gambar 12, gambar 13, gambar 14, dan gambar 15 dibawah ini.

iterasi ke-1

TR001	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	kecil	20-50	menikah	baik
TR002	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	Universitas	1-2 tahun	menengah	20-50	menikah	baik
TR003	xxxx	1x angsuran	= jumlah pinjaman	1-2 orang	SLTA	1-2 tahun	kecil	51-60	menikah	baik
TR004	xxxx	1x angsuran	> jumlah pinjaman	1 orang	SLTA	1-2 tahun	kecil	51-60	menikah	baik

Gambar 6. Hasil iterasi ke-1

iterasi ke-2

TR005	xxxx	2x angsuran	> jumlah pinjaman	1 orang	SLTA	3-5 tahun	menengah	20-50	menikah	baik
TR006	xxxx	2x angsuran	< jumlah pinjaman	1 orang	Universitas	1-2 tahun	menengah	20-50	menikah	normal
TR007	xxxx	3x angsuran	< jumlah pinjaman	2-3 orang	Universitas	>5 tahun	menengah	<20 / >60	menikah	normal
TR008	xxxx	2x angsuran	> jumlah pinjami	1 orang	SLTA	1-2 tahun	menengah	20-50	menikah	normal

Gambar 7. Hasil iterasi ke-2

iterasi ke-3

TR009	xxxx	1x angsuran	< jumlah pinjaman	tidak ada	SMP	<1 tahun	menengah	51-60	belum menikah	buruk
TR010	xxxx	1x angsuran	= jumlah pinjaman	2-3 orang	SD	<1 tahun	perorangan	51-60	menikah	buruk
TR011	xxxx	1x angsuran	< jumlah pinjaman	> 3 orang	SLTA	<1 tahun	kecil	51-60	menikah	Buruk
TR012	xxxx	1x angsuran	= jumlah pinjaman	> 3 orang	Universitas	1-2 tahun	menengah	51-60	janda / duda	Buruk

Gambar 8. Hasil iterasi ke-3

iterasi ke-4

TR013	xxxx	1x angsuran	= jumlah pinjaman	tidak ada	SMP	<1 tahun	kecil	51-60	belum menikah	Buruk
TR014	xxxx	1x angsuran	> jumlah pinjaman	tidak ada	SLTA	1-2 tahun	kecil	20-50	menikah	Baik
TR015	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	menengah	20-50	menikah	Baik
TR016	xxxx	1x angsuran	= jumlah pinjaman	1 orang	SLTA	1-2 tahun	kecil	20-50	menikah	Baik

Gambar 9. Hasil iterasi ke-4

iterasi ke-5

TR017	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	menengah	<20 / >60	menikah	Baik
TR018	xxxx	1x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	menengah	20-50	menikah	Baik
TR019	xxxx	1x angsuran	= jumlah pinjaman	1 orang	SLTA	<1 tahun	perorangan	20-50	menikah	baik
TR020	xxxx	1x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	menengah	20-50	menikah	baik

Gambar 10. Hasil iterasi ke-5

iterasi ke-6

TR021	xxxx	1x angsuran	= jumlah pinjaman	2-3 orang	Universitas	<1 tahun	perorangan	51-60	menikah	baik
TR022	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	Universitas	3-5 tahun	menengah	51-60	menikah	baik
TR023	xxxx	1x angsuran	= jumlah pinjaman	2-3 orang	SLTA	<1 tahun	kecil	20-50	menikah	baik
TR024	xxxx	1x angsuran	> jumlah pinjaman	1 orang	SLTA	1-2 tahun	kecil	51-60	menikah	baik

Gambar 11. Hasil iterasi ke-6

iterasi ke-7

TR025	xxxx	2x angsuran	> jumlah pinjaman	1 orang	SLTA	1-2 tahun	menengah	20-50	menikah	baik
TR026	xxxx	2x angsuran	< jumlah pinjaman	1 orang	Universitas	1-2 tahun	perorangan	20-50	menikah	baik
TR027	xxxx	3x angsuran	< jumlah pinjaman	2-3 orang	SLTA	3-5 tahun	besar	20-50	menikah	baik
TR028	xxxx	1x angsuran	> jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	perorangan	20-50	menikah	Normal

Gambar 12. Hasil iterasi ke-7

iterasi ke-8

TR029	xxxx	2x angsuran	> jumlah pinjaman	2-3 orang	Universitas	3-5 tahun	menengah	51-60	menikah	Normal
TR030	xxxx	1x angsuran	< jumlah pinjaman	2-3 orang	Universitas	<1 tahun	perorangan	20-50	menikah	baik
TR031	xxxx	1x angsuran	= jumlah pinjaman	>3 orang	SLTA	<1 tahun	kecil	20-50	janda / duda	Buruk
TR032	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	menengah	<20 / >60	menikah	Buruk

Gambar 13. Hasil iterasi ke-8

iterasi ke-9

TR033	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SD	<1 tahun	menengah	<20 / >60	menikah	Buruk
TR034	xxxx	3x angsuran	< jumlah pinjaman	>3 orang	SD	3-5 tahun	menengah	<20 / >60	janda / duda	Buruk
TR035	xxxx	1x angsuran	= jumlah pinjaman	tidak ada	SD	1-2 tahun	perorangan	<20 / >60	belum menikah	Buruk
TR036	xxxx	1x angsuran	= jumlah pinjaman	tidak ada	SLTA	1-2 tahun	menengah	51-60	belum menikah	Buruk

Gambar 14. Hasil iterasi ke-9

iterasi ke-10

TR037	xxxx	1x angsuran	= jumlah pinjaman	1 orang	SD	<1 tahun	perorangan	51-60	menikah	Buruk
TR038	xxxx	1x angsuran	< jumlah pinjaman	>3 orang	SMP	<1 tahun	kecil	51-60	belum menikah	Buruk
TR039	xxxx	2x angsuran	= jumlah pinjaman	>3 orang	SMP	1-2 tahun	menengah	51-60	janda / duda	Buruk
TR040	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	SLTA	1-2 tahun	kecil	20-50	menikah	Baik
TR041	xxxx	2x angsuran	= jumlah pinjaman	2-3 orang	Universitas	1-2 tahun	menengah	20-50	menikah	Baik
TR042	xxxx	2x angsuran	= jumlah pinjaman	1 orang	SLTA	3-5 tahun	menengah	<20 / >60	menikah	Buruk
TR043	xxxx	3x angsuran	= jumlah pinjaman	>3 orang	SLTA	3-5 tahun	menengah	20-50	menikah	Baik

Gambar 15. Hasil iterasi ke-10

Hasil dari implementasi 10-fold cross validation dengan jumlah data 43 set sebagai berikut:

$$\text{Akurasi} = \frac{40}{43} \times 100 = 93\%$$

Dari 10 iterasi yang dilakukan terhadap model dataset diperoleh hasil akurasi 93%.

KESIMPULAN DAN SARAN

Hasil dari kinerja algoritma k-nearest neighbour yang digunakan pada pendukung keputusan penentuan debitur pada KSP galih manunggal menggunakan metode k-fold Cross Validation memberikan hasil 93%. Dengan demikian implementasi algoritma k-nn memberikan rekomendasi akurasi yang tinggi.

DAFTAR PUSTAKA

[1] A. Nugroho, "Implementasi Algoritma K-Nearest Neighbor Dalam Memprediksi Potensi Calon Kreditur di KSP Galih Manunggal," *Dasi*, vol. 17, no. 2, pp. 1-6, 2016.

[2] S. Parvande, H. W. Yeh, M. P. Paulus, and B. A. McKinney, "Consensus features nested cross-validation," *Bioinformatics*, vol. 36, no. 10, pp. 3093-3098, 2020, doi: 10.1093/bioinformatics/btaa046.

[3] H. Azis, P. Purnawansyah, F. Fattah, and I. P. Putri, "Performa Klasifikasi K-NN dan Cross Validation Pada Data Pasien Pengidap Penyakit Jantung," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 81-86, 2020, doi: 10.33096/ilkom.v12i2.507.81-86.

[4] L. Mardiana, D. Kusnandar, and N. Satyahadewi, "Analisis Diskriminan Dengan K Fold Cross Validation Untuk Klasifikasi Kualitas Air Di Kota Pontianak," *Bimaster Bul. Ilm. Mat. Stat. dan Ter.*, vol. 11, no. 1, pp. 97-102, 2022, [Online]. Available: <https://jurnal.untan.ac.id/index.php/jbmstr/article/view/51608>

[5] R. Tuntun, K. Kusri, and K. Kusnawi, "Analisis Perbandingan Kinerja Algoritma Klasifikasi

dengan Menggunakan Metode K-Fold Cross Validation," *J. Media Inform. Budidarma*, vol. 6, no. 4, p. 2111, 2022, doi: 10.30865/mib.v6i4.4681.

[6] J. T. Samudra, B. H. Hayadi, and P. S. Ramadhan, "Komparasi 3 Metode Algoritma Klasifikasi Data Mining Pada Prediksi Kenaikan Jabatan," *J-SISKO TECH (Jurnal Teknol. Sist. Inf. dan Sist. Komput. TGD)*, vol. 5, no. 2, p. 127, 2022, doi: 10.53513/jsk.v5i2.5642.

[7] P. Putra, A. M. H. Pardede, and S. Syahputra, "Analisis Metode K-Nearest Neighbour (Knn) Dalam Klasifikasi Data Iris Bunga," *J. Tek. Inform. Kaputama*, vol. 6, no. 1, pp. 297-305, 2022.

[8] D. Fitriana, S. Dwiasnati, H. H. H, and K. A. Baihaqi, "Penerapan Metode Machine Learning untuk Prediksi Nasabah Potensial menggunakan Algoritma Klasifikasi Naïve Bayes," *Fakt. Exacta*, vol. 14, no. 2, p. 92, 2021, doi: 10.30998/faktorexacta.v14i2.9297.

[9] W. O. N. Kadir, B. Pramono, and Stiswaty, "Penerapan Data Mining Dengan Metode K-Nearest Neighbor (KNN) Untuk Mengelompokan Minat Konsumen Asuransi (PT. Jasaraharja Putera)," *J. Semant.*, vol. 5, no. 1, pp. 97-104, 2019.

[10] A. S. Aribowo, "Metode Data Mining Untuk Klasifikasi Kesetiaan Pelanggan Terhadap Merek Produk," *Semin. Nas. Sist. Inf. Indones.*, pp. 535-541, 2013.

[11] O. : Ahadiyah and N. Kholifah, "Classification Analysis of Credit Customers X Cooperatives Using Decision Tree C4.5 and Naïve Bayes," pp. 1-8, 2012.

[12] M. Mesran, T. M. Diansyah, and F. Fadlina, "Implemententasi Metode Rank Order Cendroid (ROC) dan Operational Competitiveness Rating Analysis (OCRA) dalam Penilaian Kinerja Dosen Komputer Menerapkan (Studi Kasus: STMIK Budi Darma)," *Pros. Semin. Nas. Ris. Inf. Sci.*, vol. 1, no. September, p. 822, 2019, doi: 10.30645/senaris.v1i0.89.

[13] S. A. dan A. Furqani, "Skripsi artikel skripsi," vol. IX, no. 8. pp. 1-15, 2018.

[14] Y. Widyaningsih, G. P. Arum, and K. Prawira, "Aplikasi K-Fold Cross Validation Dalam Penentuan Model Regresi Binomial Negatif Terbaik," *BAREKENG J. Ilmu Mat. dan Terap.*, vol. 15, no. 2, pp. 315-322, 2021, doi: 10.30598/barekengvol15iss2pp315-322.

[15] Z. R. Tembusai, H. Mawengkang, and M. Zarlis, "K-Nearest Neighbor with K-Fold Cross Validation and Analytic Hierarchy Process on Data Classification," *Int. J. Adv. Data Inf. Syst.*, vol. 2, no. 1, pp. 1-8, 2021, doi: 10.25008/ijadis.v2i1.1204.