

IMPLEMENTASI ALGORITMA K-NEAREST NEIGHBOR UNTUK KLASIFIKASI OMSET USAHA MIKRO DI KABUPATEN TIMOR TENGAH UTARA

Emerensiana Okan Manhitu¹, Yoseph Pius Kurniawan Kelen², Debora Chrisinta³
^{1,2,3}Universitas Timor

(Program Studi Teknologi Informasi Fakultas Pertanian Sains dan Kesehatan Universitas
Timor)

(Jl. Eltari KM. 15 Kefamenanu, Timor Tengah Utara, Nusa Tenggara Timur)

e-mail: 1mersimanhitu@gmail.com, 2yosepkelen@unimor.ac.id, 3deborachrisinta@unimor.ac.id

Abstrak

Penelitian ini berjudul "Implementasi Algoritma K-Nearest Neighbor Untuk Klasifikasi Omset Usaha Mikro Di Kabupateen Timor Tengah Utara" bertujuan untuk menerapkan Algoritma K-Nearest Neighbor dalam klasifikasi omset usaha mikro. K-Nearest Neighbor adalah metode klasifikasi terhadap sekumpulan data berdasarkan pembelajaran data yang sudah terklasifikasi sebelumnya. Penelitian terkait K-Nearest Neighbor telah banyak dilakukan, salah satunya menjelaskan bahwa metode klasifikasi K-Nearest Neighbor memiliki kelebihan di antaranya pelatihan cepat, sederhana, tahan pada data pelatihan yang memiliki gangguan(noise) dan lebih efektif apabila data latih yang berjumlah besar. Data yang digunakan meliputi karakteristik usaha mikro dan omset yang dihasilkan, yang diolah menggunakan bahasa pemrograman Python. Proses implementasi mencakup pengumpulan data, preprocessing, pemisahan dataset menjadi data pelatihan dan pengujian, serta evaluasi kinerja model menggunakan metrik akurasi. Dengan menerapkan data mining penerapan algoritma K-Nearest Neighbor untuk klasifikasi usaha masyarakat berdasarkan jenis izin usaha, algoritma K-Nearest Neighbor terbukti efektif dalam mengklasifikasikan jenis izin usaha berdasarkan data yang telah diinput. Hasil dari penelitian ini diharapkan dapat memberikan wawasan tentang faktor-faktor yang mempengaruhi omset usaha mikro dan mendukung pengambilan keputusan bagi pemangku kepentingan dalam meningkatkan daya saing usaha mikro di Kabupaten Timor Tengah Utara. Usaha kecil menengah memiliki keunggulan antara lain mudah dalam menciptakan lapangan kerja dibandingkan dengan sektor bisnis lainnya, memiliki kemampuan penyesuaian diri terhadap kondisi pasar yang terus menerus mengalami perubahan, dan memiliki peluang yang besar sehingga mampu memberikan kontribusi yang signifikan pada perdagangan dan ekspor.

Kata kunci: K-Nearest Neighbor, Klasifikasi, Omset, Usaha Mikro, Python.

Abstract

This research, entitled "Implementation of the K-Nearest Neighbor Algorithm for Classification of Micro-Business Turnover in North Central Timor Regency," aims to apply the K-Nearest Neighbor Algorithm in classifying micro-business turnover. KNN is a classification method for a set of data based on learning from previously classified data. Numerous studies related to KNN have been conducted, one of which explains that the K-Nearest Neighbor classification method has advantages including fast training, simplicity, resistance to noise in the training data, and greater effectiveness when the training data is large. The data used includes the characteristics of micro-businesses and the resulting turnover, processed using the Python programming language. The implementation process includes data collection, preprocessing, dataset splitting into training and testing data, and model performance evaluation using accuracy metrics. By applying data mining and the K-Nearest Neighbor algorithm for classifying community businesses based on the type of business license, the KNN algorithm has proven

effective in classifying types of business licenses based on the data inputted. The results of this study are expected to provide insights into the factors affecting micro-business turnover and support decision-making for stakeholders in improving the competitiveness of micro-businesses in North Central Timor Regency. Small and medium enterprises have advantages such as ease in creating jobs compared to other business sectors, the ability to adapt to constantly changing market conditions, and significant opportunities to contribute to trade and exports.

Keywords: *K-Nearest Neighbor, classification, revenue, micro business, Python.*

1. PENDAHULUAN

Berdasarkan kemajuan zaman, pertumbuhan dan pembangunan ekonomi sangat berperan penting dalam meningkatkan pendapatan ekonomi daerah termasuk dalam mensejahterakan rakyat. Peran usaha mikro, kecil dan menengah (UMKM) memiliki arti yang begitu penting bagi suatu daerah terutama sebagai salah satu penggerak pertumbuhan ekonomi daerah. Kegiatan usaha mikro, kecil dan menengah (UMKM) merupakan salah satu cara agar produk kreatif daerah dapat dikenal dan memberikan peluang bisnis bagi pelaku usaha di daerah. Selain itu, peran pelaku usaha mikro, kecil dan menengah (UMKM) dipandang sangat penting guna meningkatkan pendapatan perkapita maupun meningkatkan perekonomian suatu daerah. (Azizah et al. 2022)

Berdasarkan data Dinas Kementerian Koperasi Indonesia (2022) perkembangan UMKM di Indonesia pada tahun 2019 mencapai 65,46 juta unit dengan pangsa sebesar 99% dari total usaha di Indonesia. Jumlah tersebut naik 1,98% dibandingkan pada tahun 2018 yaitu sebesar 64,19 juta unit. Dengan total tersebut, pada tahun 2019 UMKM mampu menyerap tenaga kerja sebanyak 119,56 juta jiwa di seluruh Indonesia. Selain itu, menurut data dari Kementerian Koperasi dan Usaha Kecil dan Menengah pada Maret 2021, UMKM yang ada di seluruh Indonesia telah berkontribusi terhadap Produk Domestik Bruto (PDB) sebanyak 61,07%. (Aliyah 2022)

Usaha mikro sangat membantu perekonomian Indonesia dan dapat membantu meningkatkan kesejahteraan masyarakat. Usaha mikro di Indonesia, termasuk di Kabupaten Timor Tengah Utara memiliki peran penting dalam meningkatkan pertumbuhan ekonomi dan pengurangan kemiskinan. Namun, usaha mikro tersebut seringkali mengalami kesulitan dalam mengelola keuangan dan menghadapi tantangan dalam meningkatkan omset. Oleh sebab itu, untuk mengatasi permasalahan tersebut maka perlu dilakukan pengklasifikasian Omset Usaha Mikro di Kecamatan Kota Kefamenanu untuk membantu meningkatkan efisiensi dan akurasi dalam prediksi omset usaha mikro, serta memberikan kontribusi pada pengembangan strategi bisnis yang lebih efektif untuk usaha mikro. Salah satu cara untuk meningkatkan omset usaha mikro adalah dengan menggunakan analisis data yang lebih efektif dan efisien.

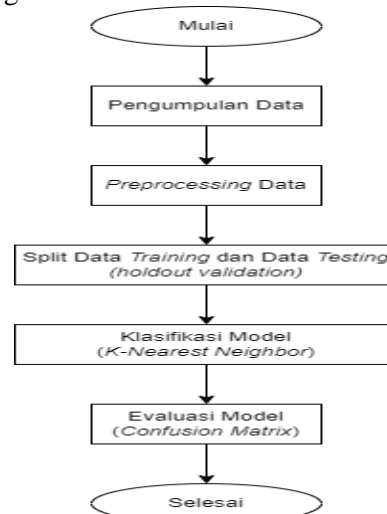
K-Nearest Neighbor (K-NN) merupakan teknik klasifikasi data berdasarkan pada klasifikasi data pada data latih yang memiliki kedekatan dengan data uji, dengan sejumlah k data yang terdekat. Teknik ini menghitung kedekatan kasus baru dengan kasus lama berdasarkan pencocokan bobot. Terdapat tiga kunci dalam algoritma ini yaitu sejumlah variabel, jarak atau kesamaan antar objek, dan ukuran dari k yaitu sejumlah data terdekat. Untuk mengklasifikasikan data yang belum diketahui kelasnya, jarak data baru ke data yang telah diketahui kelasnya dihitung, k -terdekatnya diidentifikasi, dan label kelas terdekat ini kemudian digunakan untuk menentukan label kelas dari objek tersebut. (Widaningsih 2022)

Pada penelitian ini akan dilakukan klasifikasi Omset Usaha Mikro dengan algoritma *K-Nearest Neighbor (K-NN)* dengan menambahkan metode *hold-out* pada proses data training dan testing. *Hold-Out* adalah salah satu dari jenis pengujian *Cross Validation* yang berfungsi untuk menilai kinerja proses sebuah metode algoritma dengan membagi dataset menjadi dua bagian yaitu data pelatihan dan data pengujian. Kelebihan *hold-out* dibandingkan dengan *cross validation* lainnya adalah sederhana dengan waktu komputasi yang cepat. Pada penelitian yang dilakukan oleh (Oktafiani, Hermawan, and Avianto 2023) *hold-out validation* lebih baik daripada skema *k-fold cross validation* untuk Random Forest (95.85%) dan SVM (98.89%). Keuntungan metode ini adalah beban komputasinya yang lebih rendah.

Algoritma K-NN sebelumnya sudah pernah digunakan untuk klasifikasi Status Stunting Balita. Penelitian tersebut dilakukan oleh (Prasetya et al. 2020) dihasilkan akurasi sebesar 98,89% pada pengujian 300 data balita. Penelitian dengan menggunakan metode K-NN juga dilakukan oleh (Wahyuni et al. 2022) penelitian tersebut menggunakan metode K-NN untuk klasifikasi jenis suara berdasarkan jangkauan vokal, parameter k yang digunakan dari 1 sampai 40. Hasil dari penelitian tersebut menghasilkan akurasi terbaik sebesar 88.88% dengan nilai k sebesar 8. Pada penelitian yang dilakukan oleh (Loka and Marsal 2023) membandingkan algoritma *K-Nearest Neighbors (K- NN)* dan Naïve Bayes untuk klasifikasi status gizi pada balita, dengan nilai akurasi tertinggi terdapat pada algoritma K-NN adalah 96,10%.

2. METODE PENELITIAN

Gambar berikut merupakan ilustrasi yang memperlihatkan tahapan penelitian yang dilakukan oleh peneliti adalah sebagai berikut :



Gambar 1. Tahapan Penelitian

Tahapan ini diawali dengan tahapan pengumpulan dataset. Data yang digunakan pada penelitian ini melalui observasi, studi literatur dan wawancara. Variabel penelitian yang digunakan adalah Jumlah pekerja, Modal, Aset dan Omset. Penentuan kebutuhan data penelitian adalah sebagai berikut: sampel data yang diambil yaitu 300 data. Data Usaha Mikro bersumber dari Dinas Koperasi Usaha Kecil dan Menengah Kabupaten Timor Tengah Utara dari tahun 2021-2023. Masuk pada tahap *Pre-Processing* dataset adalah proses yang mengubah data mentah ke dalam bentuk yang digunakan sebagai masukan Algoritma. Berikut adalah empat tahap *Pre-Processing* dataset: **Data Cleaning (Pembersihan Data)** Tahap *Pre-Processing* pertama adalah data cleaning. Ini merupakan proses identifikasi dan perbaikan masalah dalam data. Misalnya, kesalahan input, nilai hilang dan duplikasi. Tujuan utama dari tahap ini adalah untuk menciptakan dataset yang konsisten dan akurat. **Data Integration (Integrasi Data)** Kedua, tahap *Pre-Processing* data adalah data integration. Hal ini melibatkan penggabungan data dari berbagai sumber menjadi satu dataset yang kohesif. Tahapan ini penting dilakukan ketika informasi dikumpulkan dari berbagai database atau saat bekerja dengan dataset besar yang tersebar. **Data Transformation (Transformasi Data)** Tahap *Pre-Processing* data berikutnya adalah data transformation. Tahap ini adalah proses mengubah data ke dalam format atau struktur yang lebih sesuai untuk analisis. **Data Reduction (Pengurangan Data)** Tahap *Pre-Processing* data terakhir adalah data reduction. Ini merupakan proses pengurangan data untuk mengurangi volume data tanpa menghilangkan informasi penting. Hal ini bisa mempercepat pemrosesan dan analisis. Langkah-langkah imputasi *missing* data dengan metode K-NN adalah sebagai berikut:

1. Tentukan parameter K, K adalah jumlah observasi terdekat atau tetangga terdekat yang akan digunakan.

2. Menghitung jarak antara observasi yang mengandung *missing* data dengan observasi lengkap pada variabel ke-j yang tidak mengandung *missing* data dengan variabel j lainnya yang bersesuaian dengan rumus jarak *Euclidian*.

$$d(xa, xb) = \sqrt{\sum_{j=1}^m (x_{aj} - x_{bj})^2}. \quad (1)$$

Keterangan :

$d(xa, xb)$ = jarak observasi

m = dimensi data

j = variabel data

x_{aj} = nilai dari variabel ke-j pada setiap observasi *missing* data

x_{bj} = nilai dari variabel lainnya pada setiap observasi

3. Urutkan jarak berdasarkan observasi yang memiliki nilai jarak terbesar sampai observasi yang memiliki nilai jarak terkecil.

4. Menentukan K observasi terdekat berdasarkan nilai jarak terkecil.

5. Melakukan imputasi *missing* data dengan menghitung nilai *weight mean estimation* pada K observasi terdekat yang tidak mengandung nilai *missing* data dengan rumus (Larose, 2005)

$$X_j = \frac{\sum_k^K W_k V_k}{\sum_k^K W_k} \quad (2)$$

Keterangan :

X_j = estimasi rata-rata berbobot

V_k = nilai pada data lengkap pada variabel *missing* data

K = observasi terdekat yang digunakan

W_k = bobot observasi tetangga terdekat ke K dengan rumus

$$W_k = \frac{1}{d(W_{ak}, W_{bk})^2} \quad (3)$$

3. Hasil dan Pembahasan

3.1. Hasil Klasifikasi K-Nearest Neighbor dengan Python

Implementasi algoritma *K-Nearest Neighbor* tahap awal yang dilakukan adalah untuk memperbaiki masalah atau melakukan penyempurnaan atau pembersihan terhadap data yang akan di masukan ke dalam proses *modelling*.

Tabel 1. Data Usaha Mikro

No	Jumlah Pekerja	Modal	Aset	Omset
1	1	15,000,000	10,000,000	12.000.000
2	1	18,000,000	13,000,000	15.000.000
3	1	8,000,000	10,000,000	17.000.000
4	1	18,000,000	10,000,000	18.000.000
5	1	15,000,000	8,000,000	12.000.000
6	1	5,000,000	3,000,000	2.500.000
.....
223	9	200.000.000	300.000.000	360.000.000

3.1.1 Kategori Omset

Mengkategorikan omset menjadi rendah dan tinggi merupakan langkah penting bagi pelaku usaha mikro untuk memahami posisi mereka di pasar, merumuskan strategi yang tepat, dan meningkatkan kinerja bisnis secara keseluruhan. Hal ini juga membantu dalam pengambilan keputusan terkait pengelolaan keuangan, pemasaran, serta pengembangan sumber daya manusia. Berikut kategori omset berdasarkan rata-rata.

Tabel 2. Kategori Omset

No	Rata-Rata Omset	Kategori
1	<20.018.677	Rendah
2	>20.018.677	Tinggi

Tabel di atas menjelaskan bahwa usaha mikro dengan rata-rata omset kurang dari 20.018.677 dapat dikategorikan sebagai omset rendah, sedangkan yang memiliki omset lebih dari 20.018.677 dapat dianggap sebagai omset tinggi. Berikut adalah tabel kategori omset dari 223 data.

Tabel 3. Hasil Kategori Omset

No	Jumlah Pekerja	Modal	Aset	Omset
1	1	15,000,000	10,000,000	Rendah
2	1	18,000,000	13,000,000	Rendah
3	1	8,000,000	10,000,000	Rendah
4	1	18,000,000	10,000,000	Rendah
5	1	15,000,000	8,000,000	Rendah
6	1	5,000,000	3,000,000	Rendah
.....
223	9	200.000.000	300.000.000	Tinggi

Data omset dikategorikan kedalam kelas ‘Rendah’ dan ‘Tinggi’. Hal ini dilakukan agar membuat data lebih sederhana dari tabel sebelumnya dan lebih fokus pada data yang relevan yang akan digunakan pada proses *modelling*.

3.1.2 Menentukan Atribut Dan Target Dan Prediktor

Dataset yang telah disiapkan akan dibagi menjadi atribut dan target, Dimana kolom jumlah pekerja, modal dan aset dijadikan sebagai atribut (data x) dan kolom omset akan bertindak sebagai target (data y). Berikut adalah *coding* dan tampilan hasil untuk menampilkan data x dan data y :

```
x = df_cleaned[['Jumlah Pekerja', 'Modal', 'Aset']]
y = df_cleaned['Omset']
print("Data x:")
print(x)
print("\nData y:")
print(y)
```

```
Data x:
      Jumlah Pekerja      Modal      Aset
0              1  15000000  10000000
1              1  18000000  13000000
2              1   8000000  10000000
3              1  18000000  10000000
4              1  15000000   8000000
..           ...
218             2   250000   100000
219             2  1000000   500000
220             2  1000000  15000000
221             4  1000000   5000000
222             9 200000000 300000000

[223 rows x 3 columns]
```

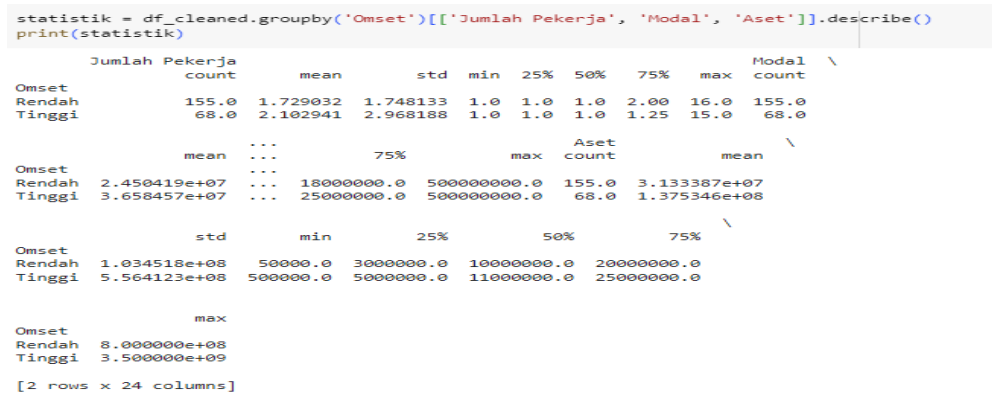
```
Data y:
0      Rendah
1      Rendah
2      Rendah
3      Rendah
4      Rendah
...
218     Rendah
219     Tinggi
220     Tinggi
221     Tinggi
222     Tinggi
Name: Omset, Length: 223, dtype: object
```

Gambar 2. Data Atribut dan Target

x dan y digunakan untuk memisahkan data atribut dan y sebagai target. Kolom “jumlah pekerja”, “modal”, dan “aset” sebagai atribut untuk melakukan analisis statistik dan visualisasi data dalam model. Sedangkan kolom “omset” sebagai target yang akan digunakan untuk diprediksi dalam model.

3.1.3 Menghitung Statistik Deskriptif Berdasarkan Kelas Omset

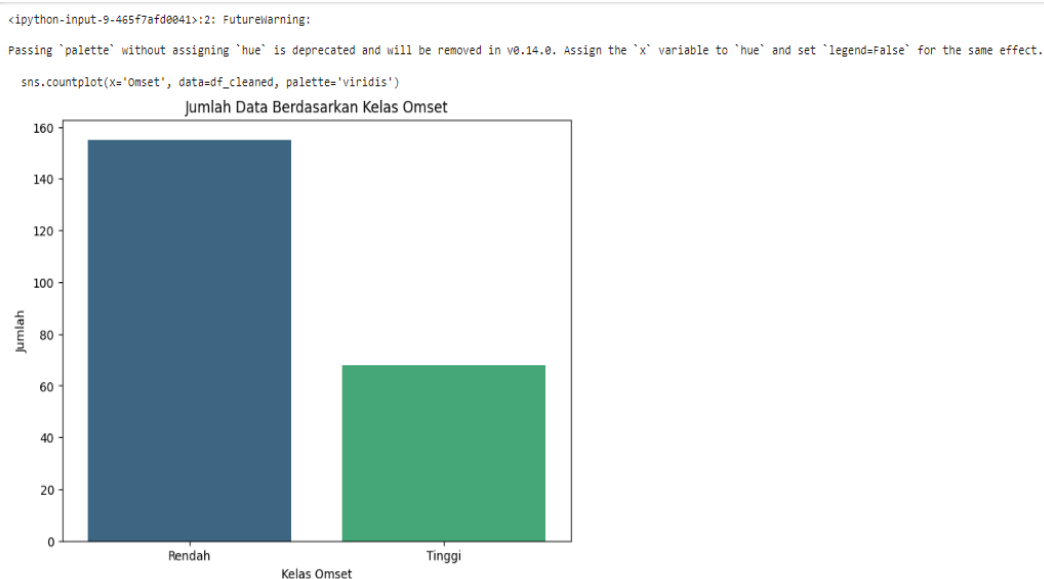
Statistik deskriptif digunakan untuk memberikan gambaran umum tentang karakteristik data dalam setiap kelas. Peneliti akan menghitung statistik deskriptif seperti mean, standar deviasi, minimum, kuartil (25%, 50%, 75%), dan maksimum untuk dua kategori omset: Rendah dan Tinggi. Berikut merupakan *coding* dan hasil yang digunakan untuk mendeskripsikan dataset:



Gambar 3. Statistik Deskriptif

Dari hasil statistik deskriptif menunjukkan bahwa terdapat perbedaan signifikan dalam jumlah pekerja, modal, dan aset antara usaha mikro dengan omset rendah dan tinggi. Usaha mikro dengan omset tinggi cenderung memiliki jumlah pekerja, modal, dan aset yang lebih besar dibandingkan dengan usaha mikro dengan omset rendah. Pada kelas omset ‘Tinggi’ menunjukkan bahwa ada lebih banyak perbedaan di antara usaha mikro yang mungkin mencerminkan strategi bisnis atau model operasional yang berbeda.

3.1.4 Jumlah Data Berdasarkan Kelas Omset



Gambar 4. Jumlah Dataset

Grafik histogram yang menunjukkan jumlah data berdasarkan kelas omset dengan kelas “Rendah” berjumlah 157 data, dan kelas “Tinggi” berjumlah 68 data. Hal ini menunjukkan bahwa

kelas omset ‘Rendah’ memiliki lebih banyak entri daripada ‘Tinggi’, ini menunjukkan bahwa kegagalan dalam meningkatkan omset usaha mikro.

3.1.5 Encoding Target Label Menjadi Nilai Numerik

Mengubah label kategori dalam variabel 'y' menjadi kode numerik. Label 'Rendah' di ubah menjadi kode '0' dan label 'Tinggi' diubah menjadi '1'. Data akan lebih mudah untuk dianalisis dan dibandingkan.

[illegible]

Gambar 5. *Encoding* Target menjadi Nilai Numerik

3.1.6 Membagi Data Menjadi Data *Training* dan Data *Testing*

Data yang telah dilakukan proses *encoding* akan dibagi menjadi dua bagian yaitu data *training* dan data *testing* untuk dilakukan proses *cross validation* menggunakan *holdout*, dimana dataset akan dibagi 80% untuk data *training* dan 20% untuk data *testing*.

```
Data Set Pelatihan (X_train, y_train):
X_train:
  Jumlah Pekerja    Modal    Aset
11              1  10000000  7000000
179             1  10000000  15000000
171             1  50000000  35000000
133             1  50000000  20000000
54              1  35000000  35000000
..             ...      ...      ...
17              1   2000000   500000
98              1  10000000  15000000
66              1   1000000   2500000
126             2  75000000  25000000
109             1   5000000  20000000

[178 rows x 3 columns]

y_train:
[0 1 1 0 1 0 0 0 1 0 1 0 1 0 1 1 0 1 0 1 0 1 0 0 0 0 0 1 1 1 1 1 1 0 0 0 1 1
 0 0 0 0 1 0 0 1 0 1 0 0 0 1 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 1 0 0 1 0 0 1
 0 0 0 1 1 0 1 0 0 1 0 1 0 1 1 0 0 0 1 0 0 0 1 0 0 1 1 0 0 0 0 0 1 0 0 0
 0 0 0 0 1 0 0 1 0 1 0 1 0 0 0 0 0 0 0 1 0 0 0 1 1 1 1 0 0 0 0 1 1 0 1 0 0 0
 1 0 1 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0 0 1 1 1 0 1 1 0 1 0 0 0 1]
```

Gambar 6. *Data Training*

```
Data Set Pengujian (X_test, y_test):
```

X_test:	Jumlah Pekerja	Modal	Aset
177	1	2500000	2500000
214	1	3000000	2000000
71	2	1000000	15000000
20	1	1000000	500000
79	2	2400000	4000000
119	1	1000000	3500000
203	2	20000000	10000000
166	2	35000000	25000000
125	5	50000000	75000000
91	1	1000000	5000000
148	1	10000000	25000000
42	1	4000000	4000000
72	2	500000	15000000
19	1	5000000	2500000
187	1	10000000	10000000
152	1	5000000	15000000
167	1	5000000	3000000
100	1	30000000	20000000
170	1	10000000	30000000
206	4	5000000	5000000
128	2	25000000	15000000
31	1	5000000	6500000
172	1	35000000	35000000
205	1	10000000	18000000
181	1	150000	20000000
143	3	500000000	750000000
193	1	5000000	5500000
144	1	65000000	100000000
201	1	15000000	10000000
130	2	20000000	10000000
4	1	15000000	8000000
208	8	250000000	800000000
194	4	30000000	30000000
21	1	500000	100000
207	7	1500000	10000000
52	1	500000	50000
185	1	15000000	10000000
80	1	5000000	11000000
209	3	5000000	10000000
211	4	5000000	7500000
173	1	50000000	35000000
10	1	8000000	8500000
53	1	350000	500000
212	3	50000000	20000000
26	1	5000000	2000000

```
y_test:  
[0 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0  
 0 0 0 0 0 0 0 1 0]
```

Gambar 7. Data Testing

Jumlah data yang diperuntukan untuk proses *training* sebesar 80% dari dataset yaitu sebanyak 178 data dan untuk proses *testing* sebesar 20% dari dataset yaitu sebanyak 45 data.

3.1.7 Model *K*-Nearest Neighbor

Tahapan yang akan di lakukan pada tahapan *K-Nearest Neighbor* yaitu

- a. Menentukan rentang nilai K yang ingin di uji.

Pada tahap ini melakukan pencarian *grid search* untuk menentukan K-terbaik. Dengan melakukan *grid search* memaksimalkan metrik sendiri sebagai target untuk hasil *cross validation* terbaik.

```
param_grid = {'n_neighbors': range(1, 20)}
grid_search = GridSearchCV(knn, param_grid, cv=5, scoring='accuracy')
grid_search.fit(X_train, y_train)
print("Hasil Grid Search:")
print("Best Parameter:", grid_search.best_params_)
print("Best Score:", grid_search.best_score_)

Hasil Grid Search:
Best Parameter: {'n_neighbors': 10}
Best Score: 0.696984126984127
```

Gambar 8. Jumlah K Terbaik

Terlihat pada gambar di atas parameter terbaik untuk *grid search* adalah 10 tetangga terdekat memberi keseimbangan yang baik antar kompleksitas model untuk generalisasi terhadap data baru.

- b. Menghitung jarak antara data *testing* dengan data *training* menggunakan jarak *euclidian*.

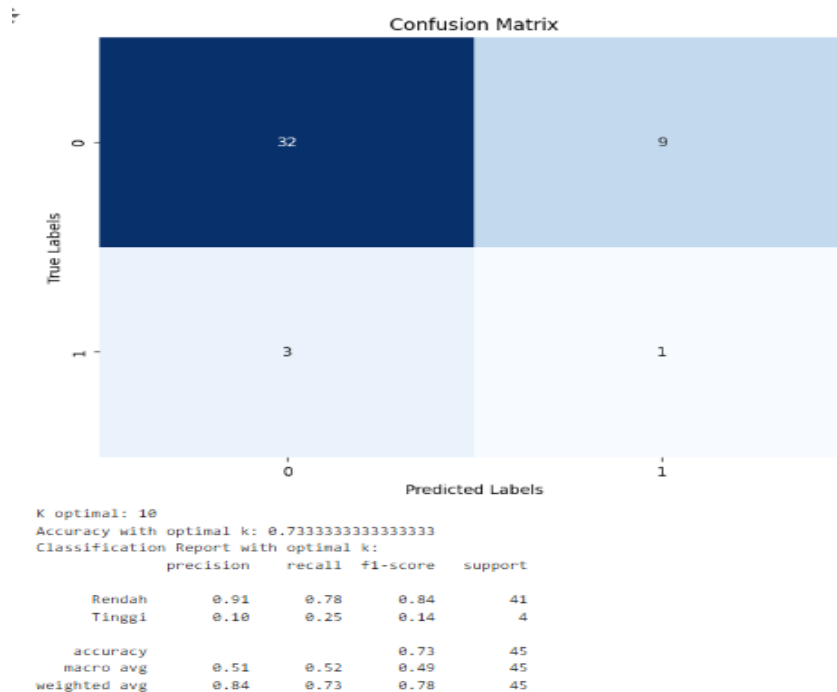
Jarak antara data *testing* dan data *training* yaitu menghitung selisih anantara koordinat masing-masing titik untuk sumbu x(pekerja), y(modal), dan z(aset). Lalu menguadratkan hasil selisih tersebut, setelah itu menjumlahkan hasil kuadrat dari ketiga selisih. Dan mengambil akar kuadrat dari jumlah tersebut untuk meendapatkan jarak. Setelah mengetahui K=10, maka didapatkan hasil sebagai berikut :

No	Jarak ke-1	Jarak ke-2	Jarak ke-3	Jarak ke-4	Jarak ke-5	Jarak ke-6	Jarak ke-7	Jarak ke-8	Jarak ke-9	Jarak ke-10
1	0	707107	1000000	1118034	1414214	1500000	1500000	1581139	1581139	1581139
2	707107	1000000	1118034	1414214	1581139	1581139	1581139	1802776	1802776	1802776
3	0	500000	2000000	2000000	3162278	4000000	5024938	5024938	5024938	5055937
4	1	500000	500000	559017	583095	707107	707107	850000	900000	917878
5	600000	600000	1077033	1077033	1077033	1077033	1077033	1077033	1077033	1166190
6	500000	707107	1000000	1118034	1118034	1118034	1500000	1500000	1500000	1500000
7	2000000	3605551	5000000	5000000	5000000	5385165	6403124	7071068	7071068	7071068
8	0	1	7071068	10000000	10000000	11180340	11180340	14142136	14142136	15811388
9	15000000	25000000	25000000	25000000	26925824	32015621	32015621	36400549	38000000	40000000
10	0	0	0	3	500000	1000000	1000000	1000000	1000000	1000000
11	5000000	5000000	5000000	7071068	7071068	7071068	7071068	8602325	9013878	9250000
12	1000000	1000000	1000000	1000000	1118034	1414214	1414214	1414214	1414214	1414214
13	500000	1000000	2061553	2500000	3354102	4500000	5000000	5006246	5099020	5099020
14	0	500000	500000	500000	1000000	1500000	1500000	1500000	1500000	2500000
15	1414214	1650000	1800000	2000000	2000000	2000000	2828427	3000000	3000000	3000000
16	0	2000000	3500000	4000000	4242641	4472136	5000000	5000000	5000000	5000000
17	0	0	500000	500000	1000000	1000000	1000000	2000000	2000000	2000000
18	5000000	7071068	7071068	7071068	7071068	10000000	10000000	11180340	11180340	13000000
19	0	10000000	10000000	10514870	11180340	11180340	11180340	11180340	12206556	12500000
20	1	3	3	3	1000000	1000000	1000000	1500000	2000000	2000000
21	1	1	5000000	5000000	7071068	7280110	8602325	9000000	11180340	11180340
22	500000	500000	1000000	1118034	1500000	1500000	1500000	1500000	1581139	1802776
23	0	1	7071068	10000000	10000000	11180340	15000000	15000000	15000000	15811388
24	2000000	2000000	3000000	3000000	3000000	5385165	5385165	5385165	5385165	5830952
25	350000	1350000	2350000	2850000	4850000	4850000	5035871	5071735	5179044	5755215
26	390512484	471699057	471699057	540832691	557675757	620987117	650000000	657438210	749166203	763216876
27	500000	500000	500000	500000	500000	1500000	1500000	1500000	1802776	2000000
28	18027756	35000000	40311289	47169906	52201533	52201533	55000000	55901699	58600341	61032778
29	0	0	2000000	3000000	4123106	4242641	5000000	5000000	5265216	5314132
30	2000000	3605551	5000000	5000000	5000000	5385165	6403124	7071068	7071068	7071068
31	2	2000000	2000000	3000000	3000000	3500000	3605551	5000000	5000000	5012235
32	304138127	335410197	335410197	389553912	502493781	503115295	636965462	715891053	728868987	737631344
33	3	5000000	7071068	7071068	7071068	7071068	11180340	14142136	15811388	15811388
34	100000	250000	304138	400000	400000	400000	565685	640312	900000	1077033
35	6	500000	500000	1000000	1250000	1500000	1500000	2061553	2120967	2121320
36	150000	254951	300000	403113	403113	450000	602080	672681	950000	1096586
37	0	0	2000000	3000000	4123106	4242641	5000000	5000000	5265216	5314132
38	1000000	1000000	1000000	1000000	1000000	1414214	1414214	2236068	2236068	2236068
39	1	1	2	2	1000000	1000000	2000000	2000000	2000000	2000000
40	3	500000	1118034	1118034	1500000	1500000	2236068	2500000	2500000	2500000
41	0	0	9	1500000	1500000	1500000	1500000	1500000	18027756	18027756
42	500000	1500000	1500000	1802776	1802776	2005617	2500000	2500000	2500000	2692582
43	150000	250000	335410	412311	471699	471699	474342	522015	650000	1004988
44	2	15000000	15000000	15000000	15811388	15811388	16007811	19209373	21213203	21213203
45	0	500000	1000000	1000000	1000000	1000000	1500000	2000000	2000000	2236068

Gambar 9. Jarak Euclidean 10 Tetangga Terdekat

3.2 Evaluasi Model

Langkah selanjutnya adalah evaluasi model untuk mengukur kinerja dari algoritma K-NN. Untuk menggambarkan kinerja dari klasifikasi adalah *confusion matrix*, matriks ini merupakan tabulasi silang sederhana dari kategori kelas yang diuji dan diprediksi, Dimana sel diagonal mewakili kelas menunjukan kelas yang diprediksi benar. Berikut *confusion matrix* dalam bentuk *heatmap*, menampilkan K terbaik, dan evaluasi model dengan K optimal:



Gambar 10. Confusion Matrix

Dapat dilihat pada gambar didapatkan hasil:

- TP(True Positive) 32 records data yang diprediksi omset rendah dan benar.
- TN(True Negative) 1 records data yang diprediksi omset tinggi dan benar.
- FP(False Positive) 9 records data yang diprediksi omset rendah dan salah.
- FN(False Negative) 3 records data yang diprediksi omset tinggi dan salah.

Pada Gambar 4.16 hasil dari *confusion matrix*, angka dari matriks tersebut dapat digunakan untuk mengukur kinerja model seperti tingkat akurasi, presisi, *recall* dan *f-1 score*. Berikut Rumus untuk mengukur kinerja model:

1. Tingkat Akurasi

Tingkat akurasi adalah rasio jumlah prediksi yang benar terhadap total jumlah sampel.

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN} = \frac{32+1}{32+9+3+1} \times 100 = 0.7333 \times 100 = 73\%$$

2. Presisi(Precision)

Presisi adalah rasio jumlah prediksi yang benar terhadap jumlah prediksi yang dianggap benar.

$$\text{Precision Rendah} = \frac{TP}{TP+FP} = \frac{32}{32+9} = 0.91$$

$$\text{Precision Tinggi} = \frac{TN}{TN+FP} = \frac{1}{1+3} = 0.10$$

$$\text{Precision macro avg} = \frac{P.\text{Rendah}+P.\text{Tinggi}}{2} = \frac{0.91+0.10}{2} = 0.51$$

$$\text{Weighted Rendah} = \frac{\text{Jumlah Contoh Rendah}}{\text{Total Jumlah Contoh}} = \frac{41}{45} = 0.91$$

$$\text{Weighted Tinggi} = \frac{\text{Jumlah Contoh Tinggi}}{\text{Total Jumlah Contoh}} = \frac{4}{45} = 0.09$$

$$\text{Precision weighted avg} = (W.\text{Rendah} \times \text{Presisi Rendah}) + (W.\text{Tinggi} \times \text{Presisi Tinggi}) = (0.91 \times 0.91) + (0.09 \times 0.10) = 0.84$$

3. Recall

Recall adalah rasio jumlah prediksi yang benar terhadap jumlah yang sebenarnya benar.

$$\text{Recall Rendah} = \frac{TP}{TP+FN} = \frac{32}{32+9} = 0.78$$

$$\text{Recall Tinggi} = \frac{TN}{TN+FN} = \frac{1}{1+9} = 0.25$$

$$Recall \text{ macro avg} = \frac{R.Rendah + R.Tinggi}{2} = \frac{0.78 + 0.25}{2} = 0.52$$

$$Weighted \text{ rendah} = \frac{Jumlah \text{ Contoh Rendah}}{Total \text{ Jumlah Contoh}} = \frac{41}{45} = 0.91$$

$$Weighted \text{ tinggi} = \frac{Jumlah \text{ Contoh Tinggi}}{Total \text{ Jumlah Contoh}} = \frac{4}{45} = 0.09$$

$$Recall \text{ Weighted.avg} = (W.Rendah \times Recall \text{ Rendah}) + (W.Tinggi \times Recall \text{ Tinggi}) = (0.91 \times 0.78) + (0.09 \times 0.25) = 0.73$$

4. F1-Score

F1-Score adalah rata-rata dari presisi dan recall, yang digunakan untuk menggabungkan kedua kriteria tersebut.

$$F1\text{-Score Rendah} = 2 \times \frac{Precision \times Recall}{Precision + Recall} = 2 \times \frac{0.91 \times 0.78}{0.91 + 0.78} = 0.84$$

$$F1\text{-Score Tinggi} = 2 \times \frac{P.Tinggi \times R.Tinggi}{P.Tinggi + R.Tinggi} = 2 \times \frac{0.10 \times 0.25}{0.10 + 0.25} = 0.14$$

$$F1\text{-Score Macro avg} = \frac{F1.Rendah + F1.Tinggi}{2} = \frac{0.84 + 0.14}{2} = 0.49$$

$$Weighted \text{ rendah} = \frac{Jumlah \text{ Contoh Rendah}}{Total \text{ Jumlah Contoh}} = \frac{41}{45} = 0.91$$

$$Weighted \text{ tinggi} = \frac{Jumlah \text{ Contoh Tinggi}}{Total \text{ Jumlah Contoh}} = \frac{4}{45} = 0.09$$

$$F1\text{-Score Weighted.avg} = (W.Rendah \times F1\text{-Score Rendah}) + (W.Tinggi \times F1\text{-Score Tinggi}) = (0.91 \times 0.84) + (0.09 \times 0.14) = 0.78$$

Untuk membuktikan hasil dari klasifikasi di atas, maka peneliti menguji model di atas dengan sebuah data baru untuk diprediksi. Berikut merupakan data hasil prediksi.

Tabel 4. Hasil Prediksi

No	Jumlah Pekerja	MODAL	ASET	OMSET
1	1	2,500,000	2,500,000	Rendah
2	1	3,000,000	2,000,000	Rendah
3	2	1,000,000	15,000,000	Tinggi
4	1	1,000,000	500,000	Rendah
5	2	2,400,000	4,000,000	Rendah
6	1	1,000,000	3,500,000	Rendah
7	2	20,000,000	10,000,000	Rendah
8	2	35,000,000	25,000,000	Rendah
9	5	50,000,000	75,000,000	Rendah
10	1	1,000,000	5,000,000	Tinggi
11	1	10,000,000	25,000,000	Tinggi
12	1	4,000,000	4,000,000	Rendah
13	2	500,000	15,000,000	Tinggi
14	1	5,000,000	2,500,000	Rendah
15	1	10,000,000	10,000,000	Rendah
16	1	5,000,000	15,000,000	Tinggi
17	1	5,000,000	3,000,000	Rendah
18	1	30,000,000	20,000,000	Rendah
19	1	10,000,000	30,000,000	Tinggi
20	4	5,000,000	5,000,000	Rendah
21	2	25,000,000	15,000,000	Rendah
22	1	5,000,000	6,500,000	Rendah
23	1	35,000,000	35,000,000	Rendah
24	1	10,000,000	18,000,000	Tinggi
25	1	150,000	20,000,000	Tinggi
26	3	500,000,000	750,000,000	Rendah
27	1	5,000,000	5,500,000	Rendah
28	1	65,000,000	100,000,000	Rendah
29	1	15,000,000	10,000,000	Rendah
30	2	20,000,000	10,000,000	Rendah

31	1	15,000,000	8,000,000	Rendah
32	8	250,000,000	800,000,000	Tinggi
33	4	30,000,000	30,000,000	Rendah
34	1	500,000	100,000	Rendah
35	7	1,500,000	10,000,000	Tinggi
36	1	500,000	50,000	Rendah
37	1	15,000,000	10,000,000	Rendah
38	1	5,000,000	11,000,000	Rendah
39	3	5,000,000	10,000,000	Rendah
40	4	5,000,000	7,500,000	Rendah
41	1	50,000,000	35,000,000	Rendah
42	1	8,000,000	8,500,000	Rendah
43	1	350,000	500,000	Rendah
44	3	50,000,000	20,000,000	Rendah
45	1	5,000,000	2,000,000	Rendah

Berdasarkan hasil prediksi omset usaha mikro yang sebelum diprediksi Ada 41 usaha mikro dikategorikan sebagai omset "Rendah." dan 4 usaha mikro dikategorikan sebagai omset "Tinggi." setelah diprediksi ada 35 usaha mikro dikategorikan sebagai omset "Rendah." dan 10 usaha mikro dikategorikan sebagai omset "Tinggi.". Artinya jumlah usaha mikro yang dikategorikan sebagai omset rendah berkurang dari 41 menjadi 35. Ini menunjukkan bahwa ada beberapa usaha yang sebelumnya dianggap beromset rendah, tetapi setelah prediksi, teridentifikasi memiliki potensi omset yang lebih tinggi. Dan jumlah usaha mikro yang dikategorikan sebagai omset tinggi meningkat signifikan dari 4 menjadi 10. Ini adalah indikasi positif bahwa model prediksi mampu mengidentifikasi usaha-usaha yang memiliki potensi omset tinggi yang sebelumnya tidak terdeteksi. Sebanyak 6 usaha mikro yang awalnya dikategorikan sebagai omset rendah kini tetap berada di kategori rendah. Hal ini mungkin menunjukkan bahwa meskipun ada upaya untuk meningkatkan kinerja, beberapa usaha tersebut masih belum mampu mencapai kriteria omset tinggi, karena kurangnya modal mengakibatkan jumlah pekerja yang tidak optimal dan aset yang tidak diperbarui.

Hasil prediksi menunjukkan adanya pergeseran positif dalam kategori omset usaha mikro, dengan peningkatan jumlah usaha dalam kategori tinggi. Ini dapat dipengaruhi oleh berbagai atribut seperti modal, jumlah pekerja, dan aset. Untuk mempertahankan dan meningkatkan kinerja ini, penting bagi pemilik usaha untuk terus mengelola modal dengan baik, meningkatkan kualitas tenaga kerja, serta memanfaatkan aset secara efisien. Selain itu, perhatian terhadap faktor eksternal seperti kondisi pasar dan strategi pemasaran juga sangat penting untuk mencapai pertumbuhan berkelanjutan. Dengan demikian, hasil prediksi omset usaha mikro ini bukan hanya memberikan informasi tentang klasifikasi omset, tetapi juga membuka peluang untuk tindakan yang lebih strategis dan terarah dalam mendukung pertumbuhan dan pengembangan sektor usaha mikro.

4. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan dapat disimpulkan sebagai berikut :

1. Klasifikasi Omset Usaha Mikro dengan Algoritma *K-Nearest Neighbor* dilakukan dengan pengumpulan data yang relevan, seperti jumlah pekerja, modal, aset, dan omset. Normalisasi untuk memastikan semua fitur berada dalam skala yang sama, membagi dataset menjadi data pelatihan dan data pengujian, menggunakan teknik GridSearchCV untuk menentukan K yang optimal, melatih model KNN dengan data pelatihan, menggunakan model untuk memprediksi kelas omset pada data pengujian.
2. Hasil yang diperoleh dalam menentukan klasifikasi omset usaha mikro dengan menggunakan *K-Nearest Neighbor* menunjukkan bahwa algoritma ini dapat mencapai akurasi yang cukup baik yaitu 73,33%. Selain itu, evaluasi metrik lain seperti presisi, recall, dan F1-Score juga menunjukkan performa yang baik. Maka *K-Nearest Neighbor* dapat digunakan sebagai alat bantu dalam pengambilan keputusan terkait klasifikasi usaha mikro.

Daftar Pustaka

- [1] Aliyah, Atsna Himmatul. "Peran Usaha Mikro Kecil Dan Menengah (UMKM) Untuk Meningkatkan Kesejahteraan Masyarakat.". *Jurnal Ilmu Ekonomi*. 2022; 3(1): 64–72. doi:10.37058/wlfrv3i1.4719.
- [2] Arisandi, Riza Rizqi Robbi, Budi Warsito, and Arief Rachman Hakim. "Aplikasi Naïve Bayes Classifier (Nbc) Pada Klasifikasi Status Gizi Balita Stunting Dengan Pengujian K-Fold Cross Validation.". *Jurnal Gaussian* 2022; 11(1): 130–39. doi:10.14710/jgaussv11i1.33991.
- [3] Azizah, Hira Wahyuni, Odi Nurdiawan, Gifthera Dwilestari, Kaslani Kaslani, and Edi Tohidi. "Klasifikasi Pemberian Bantuan UMKM Cirebon Dengan Menggunakan Algoritma K-Nearest Neighbor." *Journal of Computer System and Informatics (JoSYC)*. 2022: 3(3): 110–15. doi:10.47065/josyc.v3i3.1392.
- [4] Hidayat, Farhan. "Implementasi Algoritma Knn Dalam Mengukur Ketepatan Kelulusan Mahasiswa Uin Syarif Hidayatullah Jakarta." 2022.
- [5] Loka, Septi Kenia Pita, and Arif Marsal. "Perbandingan Algoritma K-Nearest Neighbor Dan Naïve Bayes Classifier Untuk Klasifikasi Status Gizi Pada Balita." *MALCOM: Indonesian Journal of Machine Learning and Computer Science*. 2023: 3(1): 8–14. doi:10.57152/malcomv3i1.474.
- [6] Oktafiani, Rian, Arief Hermawan, and Donny Avianto. "Pengaruh Komposisi Split Data Terhadap Performa Klasifikasi Penyakit Kanker Payudara Menggunakan Algoritma Machine Learning."; *Jurnal Sains dan Informatika* .2023: 9(April): 19–28. doi:10.34128/jsiv9i1.622.
- [7] Prasetya, Tio, Irfan Ali, Cep Lukman Rohmat, and Odi Nurdiawan. "Klasifikasi Status Stunting Balita Di Desa Slangit Menggunakan Metode K-Nearest Neighbor." "Klasifikasi Status Stunting Balita Di Desa Slangit Menggunakan Metode K-Nearest Neighbor." *INFORMATICS FOR EDUCATORS AND PROFESSIONAL: Journal of Informatics*. 2020. 5(1): 93. doi:10.51211/itbiv5i1.1431
- [8] Susanti, Shantika Martha, and Evy Sulistianingsih. "K Nearest Neighbor Dalam Imputasi Missing Data.". *Buletin Ilmiah Math. Stat. dan Terapannya*. 2018. (Bimaster) 07(1): 9–14.
- [9] Wahyudin, Edi, Kaslani Kaslani, and Rizka Widiyanti. "Implementasi Algoritma K-Nearest Neighbor Untuk Klasifikasi Data Pasien Rehabilitasi Narkoba Berdasarkan Rentang Usia." *Kopertip: Jurnal Ilmiah Manajemen Informatika dan Komputer*. 2023. 7(3): 76–79. doi:10.32485/kopertipv7i3.333.
- [10] Widaningsih, Sri. "Penerapan Data Mining Untuk Memprediksi Siswa Berprestasi Dengan Menggunakan Algoritma K Nearest Neighbor." *JATISI (Jurnal Teknik Informatika dan Sistem Informasi)*. 2022: 9(3): 2598–2611. doi:10.35957/jatisi.v9i3.859.
- [11] Y. P. K. Kelen, " Sistem Pendukung Keputusan : Konsep, Metode dan Implementasi" .
- [12] Wahyuni, Ni Made Putri, Luh Arida Ayu Rahning Putri, I Gusti Ngurah Anom Cahyadi Putra, Dewa Made Bayu Atmaja Darmawan, Made Agung Raharja, and Agus Muliantara. "Implementasi Metode K-Nearest Neighbor Dalam Mengklasifikasikan Jenis Suara Berdasarkan Jangkauan Vokal." *JELIKU (Jurnal Elektronik Ilmu Komputer Udayana)*. 2022. 11(1): 187. doi:10.24843/jlk2022.v11i01p20.
- [13] Angreany, D.M.S. "Confusion Matix". 2020
- [14] Brilliant., E.H. "Klasifikasi K-Nearest Neighbor Menggunakan R. Medium". 2019
- [15] Katarina, K., and Sriwinarti, N.K. "Analisis Metode K-Nearest Neighbors (KNN) Dan Naïve Bayes Dalam Memprediksi Kelulusan Mahasiswa". *JTIM: Jurnal Teknologi Informasi Dan Multimedia*". 2021: 3(2), 106-112.



ZONAsi: Jurnal Sistem Informasi

is licensed under a [Creative Commons Attribution International \(CC BY-SA 4.0\)](https://creativecommons.org/licenses/by-sa/4.0/)