Information System Journal (INFOS)
Vol. 8, No. 1, Mei 2025, pp. 29-39
E-ISSN : 2655-142X , P-ISSN : 2655-190X, DOI:https://doi.org/10.24076/infosjournal.2025v8i01.2117

PENERAPAN METODE *INFORMATION GAIN* DAN *LEARNING VECTOR QUANTIZATION* 3 PADA KLASIFIKASI PENYAKIT GINJAL

Muhamad Dzaky Aprima[1)], Elvia Budianita[2)], Fadhilah Syafria[3)], Iis Afrianty[4)]
[1) 2) 3) 4)] Teknik Informatika, Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim Riau

email: 12150111896@students.uin-suska.ac.id[1)], elvia.budianita@uin-suska.ac.id[2)], fadhilah.syafria@uin-suska.ac.id[3)], iis.afrianty@uin-suska.ac.id[4)]

INFO ARTIKEL

Riwayat Artikel:

Diterima Mei, 2025
Revisi Mei, 2025
Terbit Mei, 2025

Penulis Korespondensi:

Elvia Budianita
Teknik Informatika, Sains dan Teknologi, Universitas Islam Negeri Sultan Syarif Kasim Riau

Email:
elvia.budianita@uin-suska.ac.id

ABSTRAK

Penyakit Ginjal Kronis (PGK) adalah kondisi progresif yang ditandai penurunan fungsi ginjal akibat akumulasi sisa metabolik, yang dapat menyebabkan kegagalan fungsi ginjal. Prediksi dengan *data mining* berperan penting dalam upaya pencegahan penyakit ini. Penelitian ini menerapkan seleksi fitur *information gain* pada metode *Learning Vector Quantization* 3 (*LVQ* 3) dalam mengklasifikasikan penyakit ginjal kronis. Pengujian dilakukan berdasarkan 5 skenario pengujian dengan jumlah data sebanyak 1659 data dan 53 atribut. Seleksi fitur menerapkan *information gain* dengan *threshold* 0.3 dengan 36 fitur terpilih dan 0.7 dengan 33 fitur terpilih. Model diuji dengan kombinasi parameter *learning rate* dan *window* serta dievaluasi menggunakan akurasi, presisi, *recall*, dan *F1-Score*. Hasil akurasi tertinggi diperoleh tanpa menerapkan seleksi fitur sebesar 92.77%, setelah seleksi fitur, akurasi menurun 6.32% menjadi 86.45%. Kombinasi *SMOTE* dan seleksi fitur pada *threshold* 0.3 menurunkan akurasi hingga 81.64%. Hasil penelitian berhasil menerapkan *LVQ* 3 dalam klasifikasi penyakit ginjal kronis.

Kata Kunci :
Ginjal Kronis; *Information Gain*; *Learning Vector Quantization* 3; *Confusion Matrix*

ABSTRACT

Chronic Kidney Disease (CKD) is a progressive condition marked by declining renal function due to the accumulation of metabolic waste. Early prediction through data mining plays a critical role in prevention. This study investigates the application of feature selection using Information Gain combined with the Learning Vector Quantization 3 (LVQ 3) algorithm for CKD classification. The dataset comprises 1,659 records with 53 attributes. Feature selection thresholds of 0.3 and 0.7 yielded 36 and 33 features, respectively. Model performance was evaluated across various learning rate and window size configurations using accuracy, precision, recall, and F1-score. The highest accuracy (92.77%) was obtained without feature selection. Accuracy declined to 86.45% with feature selection and further decreased to 81.64% after applying SMOTE alongside feature selection (threshold 0.3). These results highlight the effective implementation of LVQ 3 in CKD classification, while also underscoring the impact of feature selection and data balancing on model performance.

***Keywords*:**
Chronic Kidney Disease; *Information Gain*; *Learning Vector Quantization* 3; *Confusion Matrix*

1. PENDAHULUAN

Penyakit Ginjal Kronis (PGK) adalah kondisi penurunan fungsi ginjal yang berlangsung lebih dari 3 bulan. Kondisi ini disebabkan oleh kerusakan ginjal yang mengakibatkan penumpukan sisa metabolik dalam tubuh [1]. Pada tahun 2019, *World Health Organization* (*WHO*) menyatakan penyakit ginjal kronis mempengaruhi 15% populasi global dengan jumlah kematian sebanyak 1,2 juta kematian. Angka kematian meningkat pada tahun 2020 menjadi 254.028 juta jiwa, pada tahun 2021 angka kematian melebihi 843,6 juta jiwa dan diperoyeksikan akan meningkat menjadi 41,5% pada tahun 2040 dan menandakan penyakit ginjal kronis merupakan penyebab kematian urutan ke-12 di antara semua penyebab kematian [2].

PGK dapat dipengaruhi beberapa faktor seperti diabetes, hipertensi, konsumsi makanan asin, lingkar pinggang, dan *Body Mass Index* (*BMI*) [3]. Penyakit ginjal kronis biasanya berbahaya, dan sebagian besar orang yang terkena tidak menunjukkan gejala sampai penyakitnya menjadi lanjut (misalnya, *eGFR* kurang dari 30 *mL*/menit per 1-73 m^2). Fungsi ginjal yang hilang kecepatannya bervariasi berdasarkan paparan, intervensi, dan etiologi , namun, kebanyakan kasus berkembangnya menjadi gagal ginjal biasanya memerlukan waktu beberapa bulan hingga beberapa dekade [4]. Prediksi dini penyakit ginjal kronis berperan penting dalam pencegahannya. Melalui prediksi yang tepat, penanganan medis dapat dilakukan lebih awal. Hal ini dapat memperlambat progresi penyakit dan meningkatkan kualitas hidup pasien [5]. Metode manual bergantung pada penilaian klinis dan hasil pengujian laboratorium, tetapi berkat perkembangan teknologi metode seperti *data mining* semakin digunakan untuk memprediksi pola dan faktor risiko yang dapat membantu dalam diagnosis dini dan perawatan yang lebih efektif [6]. *Data mining* merupakan upaya dalam menggali informasi dan pola berharga dari suatu data yang berjumlah sangat besar [7].

Penelitian ini bertujuan mengimplementasikan teknik *data mining* dalam klasifikasi tingkat akurasi penyakit ginjal kronis. Banyak penelitian mengenai klasifikasi penyakit ginjal kronis telah banyak dilakukan. Salah satunya klasifikasi data penyakit ginjal kronis di rumah sakit kabupaten gresik menggunakan metode KNN, mendapatkan hasil akurasi sebesar 90% dan tingkat kesahalan sebesar 10% [8]. Penelitian sebelumnya dalam klasifikasi penyakit ginjal kronis menggunakan algoritma *Support Vector Machine* dengan lima atribut yang digunakan seperti tekanan darah, *kreatinin serum*, volume sel padat, faktor hipertensi dan faktor anemia, penelitian ini mendapatkan hasil akurasi sebesar 96.34% [9]. Kemudian, impelementasi lain dalam klasifikasi penyakit ginjal kronis menggunakan 4 algoritma berbeda, penelitian ini melakukan perbandingan antara algoritma *Naive Bayes*, C4.5, *Logistic Regression*, dan KNN dalam analisa hubungan keempat metode, data penelitian diperoleh dari *UCI Repository* dan diolah menggunakan keempat metode tersebut. Dari hasil penelitian ini, didapatkan hasil akurasi tertinggi menggunakan algoritma *Naïve Bayes* sebesar 92,92 % [10].

Dari berbagai penelitian tersebut, penelitian ini menawarkan untuk membangun model klasifikasi penyakit ginjal kronis. Namun, pemodelan yang digunakan dalam penelitian ini menggunakan jaringan syaraf tiruan yaitu algoritma *learning vector quantization* 3 (*LVQ* 3). *learning vector quantization* (*LVQ*) mampu mengklasifikasikan pola dengan setiap *output unit* merepresentasikan kategori atau kelas tertentu, algoritma ini dikembangkan oleh *Teuvo Kohonen* pada tahun 1989 [11]. Algoritma *LVQ* 3 merupakan pengembangan dari algoritma *LVQ* 1, *LVQ* 2, dan *LVQ* 2.1, pengembangan ini merupakan penyempurnaan proses *output* yang terdapat pada proses pembobotan dan penentuan kondisi *window* [12]. Adapun penelitian yang menggunakan *LVQ* 3, dalam klasifikasi penyakit paru – paru menggunakan data rekam medis sebanyak 113 data didapatkan hasil akurasi terbaik menggunakan *epsilon* = 0.15 dan *learning rate* = 0.15 sebesar 87.5% [13]. Penelitian lain dalam klasifikasi stunting pada balita berdasarkan *anthropometric data*, penelitian ini menunjukkan bahwa meskipun dipengaruhi penggunaan variabel yang umum dan kurangnya varian data, *LVQ* 3 memiliki kinerja cukup baik dan memiliki akurasi yang signifikan yaitu sebesar 74.2% pada *window* diantara 0.3 dan 1 dengan pembagian data 70:30. Untuk mengidentifikasi nilai yang paling cocok dengan kondisi kumpulan data, diperlukan lebih banyak eksplorasi parameter [14].

Beberapa penelitian juga melakukan kombinasi penggunaan seleksi *fitur information* dengan tujuan untuk meningkatkan kinerja dan akurasi algoritma. Data yang digunakan pada penelitian ini memilki 52 atribut dan 1 kelas, karena banyaknya jumlah atribut dilakukan seleksi untuk mengidentifikasi atribut-atribut yang paling berpengaruh pada risiko penyakit ginjal kronis. Oleh karena itu, seleksi fitur *information gain* diaplikasikan dalam penelian ini. Penelitian dalam prediksi diabetes menggunakan algoritma KNN dan seleksi fitur *information gain*, mendapatkan hasil akurasi sebesar 69.11% tanpa menggunakan seleksi fitur *information gain* dengan 17 atribut, sedangkan menggunakan seleksi fitur dengan 5 atribut yang terpilih tingkat akurasi

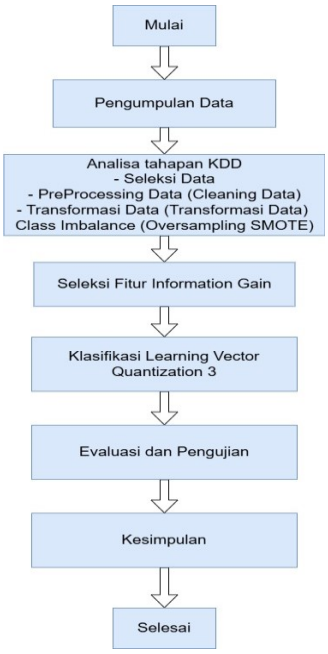
Information System Journal (INFOS)
Vol. 8, No. 1, Mei 2025, pp. 29-39
E-ISSN : 2655-142X , P-ISSN : 2655-190X, DOI:https://doi.org/10.24076/infosjournal.2025v8i01.2117

yang dihasilkan mencapai 72.93%. ini membuktikan bahwa dengan menggunakan seleksi fitur *information gain* dapat meningkatkan kinerja dan akurasi algoritma [15].

Berdasarkan pendahuluan yang telah diuraikan, maka penelitian ini bertujuan melakukan analisis akurasi menggunakan *learning vector quantization* 3 (*LVQ* 3) dan seleksi fitur *information gain* dalam klasifikasi penyakit ginjal kronis. Diharapkan implementasi ini mampu menghasilkan model klasifikasi dengan akurasi yang lebih unggul.

2. METODOLOGI PENELITIAN

Metode penelitian merupakan pendekatan sistematis untuk menyelidiki masalah secara ilmiah, yang melibatkan serangkaian tahapan terstruktur untuk mengumpulkan, menganalisis, dan menginterpretasikan data secara objektif. Tujuannya adalah memperoleh pengetahuan baru, memecahkan permasalahan, dan menguji hipotesis melalui proses penelitian yang cermat dan metodis. Adapun tahapan penelitian ditunjukkan sebagaimana Gambar 1.



Gambar 1. Tahapan Penelitian.

2.1 Pengumpulan Data

Proses pengumpulan data dilakukan dengan mengambil data sekunder sebagai bahan penelitian dari situs *Kaggle* https://www.kaggle.com/datasets/rabieelkharoua/chronic-kidney-disease-datasetanalysis/data.

2.2 Seleksi Data

Proses seleksi data adalah langkah dalam memilih atribut yang akan digunakan dalam dataset penyakit ginjal. Penelitian ini menggunakan 1659 data dengan 53 atribut dan 1 label kelas.

2.3 Preprocessing Data

Tahapan *prepocessing* data dilakukan untuk membersihkan data agar sesuai dengan kebutuhan analisis. Tahapan ini dilakukan proses pembersihan data (*data cleaning*). Selain memastikan bahwa semua fitur siap untuk digunakan dalam pelatihan model, langkah ini bertujuan untuk menghindari bias yang disebabkan oleh data yang hilang.

2.4 Transformasi Data

Transformasi data merupakan proses perubahan data yang dipilih agar sesuai model informasi atau tujuan yang ingin dicapai [16]. Normalisasi data termasuk dalam tranformasi data yang mengubah nilai menjadi 0 dan 1. Dalam penelitian ini metode normalisasi data yang digunakan yaitu *min – max* dengan rumus (1).

$$x_n = \frac{x_0 - x_{min}}{x_{max} - x_{min}} \quad (1)$$

Keterangan:
x_n = nilai yang dinormalisasi
x_0 = nilai awal atribut
x_{max} = nilai maksimal atribut
x_{min} = nilai minimal atribut

2.5 Seleksi Fitur *Information Gain*

Information gain adalah metode seleksi fitur sederhana yang berfungsi untuk merangking fitur. Metode ini sering digunakan dalam pengolahan data komputasi, klasifikasi, dan analisis citra. Keunggulan utamanya adalah kemampuannya dalam menentukan peringkat setiap atribut, semakin besar nilai suatu atribut, semakin relevan atribut tersebut untuk digunakan. Hal ini penting dilakukan karena beberapa fitur tidak diperlukan dan membuat kinerja algoritma menjadi tidak efisien. Langkah-langkah perhitungan *information gain* adalah sebagai berikut [17].

2.5.1 Menghitung *Entropy*

Menghitung *entropy* dari masing-masing fitur. Teknik ini digunakan untuk mengurangi ketidakpastian atribut tertentu, perhitungan *entropy* menggunakan persamaan (2).

$$E(S) = \sum_i^n = 1 - pi\ log2\ pi \quad (2)$$

Keterangan:
n = jumlah kelas
pi = rasio sampel pada setiap kelas
$E(S)$ = nilai *entropy* atribut

2.5.2 Menghitung Nilai *Information Gain*

Nilai *information gain* dihitung menggunakan persamaan (3).

$$Gain\ (S, A) = E(S) - \sum_{v(A)} \frac{|Sv|}{|S|}\ E(Sv) \quad (3)$$

Keterangan:
S = jumlah seluruh sampel
Sv = jumlah sampel nilai v
A = atribut
$E(Sv)$ = nilai *entropy* untuk sampel nilai v

2.6 *Learning Vector Quantization* 3

Learning Vector Quantization 3 (*LVQ* 3) merupakan salah satu metode yang digunakan untuk klasifikasi data. *LVQ* 3 merupakan pengembangan dari metode sebelumnya, *LVQ* 1 dan *LVQ* 2. *LVQ* adalah arsitektur jaringan syaraf tiruan satu lapis yang terdiri dari unit *input* dan *output*. Metode *LVQ* 3 memiliki arsitektur jaringan yang memiliki 2 lapisan utama: lapisan *input* dan lapisan kompetitif. Lapisan kompetitif bertanggung jawab untuk secara otomatis melakukan pembelajaran klasifikasi vektor input berdasarkan jaraknya. Jika dua vektor input memiliki jarak yang dekat, maka lapisan ini menempatkan vektor di kelas yang sama. Tahapan – tahapan pada metode *LVQ* 3 adalah sebagai berikut:

1. Tentukan data *input* (*Xi*), Target Kelas (T), bobot awal (W), *learning rate* (a), penurunan (dec_a), *minimum learning rate* (min_a), nilai *window* (ε) dan *epoch* (*max_epoch*).

Information System Journal (INFOS)
Vol. 8, No. 1, Mei 2025, pp. 29-39
E-ISSN : 2655-142X , P-ISSN : 2655-190X, DOI:https://doi.org/10.24076/infosjournal.2025v8i01.2117

2. Tetapkan nilai iterasi awal dengan *epoch* = 0.
3. Cek kondisi jika bernilai benar, yaitu (a > *min_a*) dan (ep < *max_epoch*) maka lanjutkan ke langkah selanjutnya. Jika salah maka langsung mendapatkan nilai bobot akhir dari pengujiannya.
4. Membaca inputan data.
5. Setelah *inputan* data dibaca, lakukan perhitungan dengan mencari jarak antara vektor *input* (*Xi*) dan vektor bobot (Wj) menggunakan persamaan (4).

$$d = \sqrt{\sum(W_j - X_t)^2} \quad (4)$$

Keterangan:
d = jarak *euclidean distance*
W_j = vektor bobot
X_t = vektor *input*

6. Selanjutnya dari hasil tahap ke - 5 tentukan jarak terdekat pertama (dc) dan jarak rterdekat kedua (dr).
7. Nilai c ditentukan sebagai kelas dari dc (jarak terdekat pertama) dan tentukan r sebagai kelas dari dr (jarak terdekat kedua).
8. nilai bobot (W) diubah jika $c = T$ dan $T \neq r$, hanya bobot dari jarak terdekat pertama yang diubah dengan persamaan (5).

$$W_{c\,(baru)} = W_{c\,(lama)} - a(x_i - W_{c\,(lama)} \quad (5)$$

9. Ubah bobot (W) jika:
   a. $c \neq r$ dan $r = T$ maka tentukan kondisi window dengan menggunakan rumus (6).

$$min\left[\frac{dc1}{dc2}, \frac{dc2}{dc1}\right] > (1 - \varepsilon)(1 + \varepsilon) \quad (6)$$

   b. Jika kondisi window terpenuhi maka bobot akan diperbaharui dengan persamaan (7) dan (8).

$$W_{c\,(baru)} = W_{c\,(lama)} - a\left(x_i - W_{c\,(lama)}\right) \quad (7)$$

$$W_{r\,(baru)} = W_{r\,(lama)} + a\left(x_i - W_{r\,(lama)}\right) \quad (8)$$

   c. Jika kondisi window tidak terpenuhi maka bobot akan diperbaharui dengan persamaan (9) dan (10).

$$W_{c\,(baru)} = W_{c\,(lama)} + \beta\left(x_i - W_{c\,(lama)}\right) \quad (9)$$

$$W_{c\,(baru)} = W_{c\,(lama)} + \beta\left(x_i - W_{c\,(lama)}\right) \quad (10)$$

Keterangan:
β = $m * a$, Dimana $0.1 < m < 0.5$

10. Turunkan *learning rate* setiap iterasi (*epoch*) dengan rumus (11).

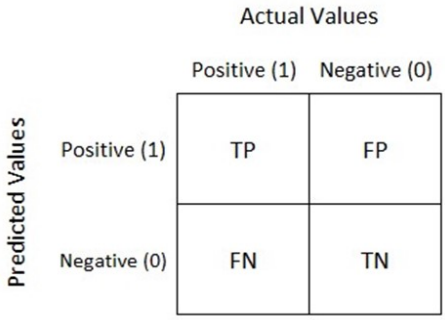
$$a = a * dec\_a \quad (11)$$

11. Proses berhenti jika jumlah iterasi maksimum tercapai atau *learning rate* turun dibawah nilai minimum yang di tetapkan.

2.7 Evaluasi dan Pengujian

Pengujian pada penelitian ini dilakukan seleksi fitur *information gain* dengan nilai *threshold* 0.3 dan 0.7. Pada model pengujian *LVQ* 3 digunakan berbagai variasi parameter seperti *learning rate* 0.1, 0.3, 0.6, dan 0.9, *window* 0.2, 0.3, 0.4, *dec_a* = 0.1, *min_a* = 0.001 dan *epsilon* = 0.3 < *m* > 0.5. Selanjutnya tahap evaluasi

menggunakan *confusion matrix*, yaitu matriks persegi dengan ukuran $N \times N$, di mana N menunjukkan jumlah kelas keluaran. Setiap baris dari matriks mewakili jumlah contoh dari kelas yang diprediksi dan setiap kolom mewakili jumlah contoh dari kelas yang sebenarnya, struktur dari *confusion matrix* dapat dilihat pada gambar 2 [18].



Gambar 2. Struktur *Confusion Matrix*.

Ada dua tipe prediksi: *correct* dan *incorrect* (*error*), *True Positive* (TP): nilai aktual dan prediksinya positif, *False Positive* (FP): Meskipun prediksinya positif, nilai aktualnya negatif, *True Negative* (TN): Nilai aktual negatif, dan prediksi negatif, *False Negative* (FN): Meskipun diprediksi negatif, namun sampelnya positif. Bagian pertama dari TP, TN, FP, dan FN, yaitu istilah benar atau salah berkaitan dengan apakah prediksi benar atau tidak. Prediksi model tentang apakah sampel atau titik data positif atau negatif adalah bagian kedua dari TP, TN, FP, dan FN.

Dengan *confusion matrix*, matrik evaluasi dapat dihitung sebagai berikut:

a. Akurasi: fraksi prediksi yang benar di antara semua prediksi atau seberapa sering sebuah prediksi benar.

$$\frac{(TP+TN)}{TP+TN+FP+FN} * 100\% \quad (12)$$

b. Presisi: adalah hasil positif yang diprediksi dengan benar.

$$\frac{(TP+TN)}{TP+FP} * 100\% \quad (13)$$

c. Recall: mengukur proporsi positif aktual yang diprediksi dengan benar, atau seberapa akurat model memprediksi kasus positif.

$$\frac{TP}{TP+FN} * 100\% \quad (14)$$

d. F1 Score: nilai gabungan antara presisi dan *recall*, terutama digunakan saat ingin menyeimbangkan antara keduanya

$$\frac{2(Precision*Recall)}{(Precision+Recall)} \quad (15)$$

3. HASIL DAN PEMBAHASAN

3.1 Pengumpulan Data

Dataset yang digunakan dalam penelitan ini diperoleh dari situs *kaggle dataset* https://www.kaggle.com/datasets/rabieelkharoua/chronic-kidney-disease-datasetanalysis/data, dataset ini berjumlah 1659 data dengan mempunyai 53 atribut dan 1 label kelas. Data dapat dilihat pada Tabel 1.

Tabel 1. Dataset Penyakit Ginjal Kronis.

Patient ID	Age	Gender	…	Diagnosis	Doctor InCharge

Information System Journal (INFOS)
Vol. 8, No. 1, Mei 2025, pp. 29-39
E-ISSN : 2655-142X , P-ISSN : 2655-190X, DOI:https://doi.org/10.24076/infosjournal.2025v8i01.2117

1	71	0	…	1	Confidential
2	34	0	…	1	Confidential
3	80	1	…	1	Confidential
4	40	0	…	1	Confidential
…	…	…	…	…	…
1659	34	1	…	1	Confidential

3.2 Seleksi Data

Pada tahap seleksi data, atribut *PatientID* dan *DoctorInCharge,* dihapus karena merupakan *identifier* yang tidak relevan untuk tujuan prediksi dalam pengujian. Proses ini mengurangi jumlah atribut dari 54 menjadi 51, dengan *Diagnosis* sebagai target kelas.

3.3 *Preprocessing* Data

Pada tahapan *pre-processing* data akan dilakukan pengecekan *missing values* dan data duplikat. Melalui Gambar 3 dapat dilihat bahwa setelah pengecekan *missing values*, diketahui bahwa dataset tidak memiliki *missing values* dan tidak ada data yang duplikat atau data yang ganda.

```
Itching                          0
QualityOfLifeScore               0
HeavyMetalsExposure              0
OccupationalExposureChemicals    0
WaterQuality                     0
MedicalCheckupsFrequency         0
MedicationAdherence              0
HealthLiteracy                   0
Diagnosis                        0
dtype: int64

Total missing values: 0
```

```
duplicated_rows = ckd.duplicated().sum()
duplicated_rows

np.int64(0)
```

Gambar 3. Cek *Missing Values* dan *Data Duplicate*.

3.4 Transformasi Data

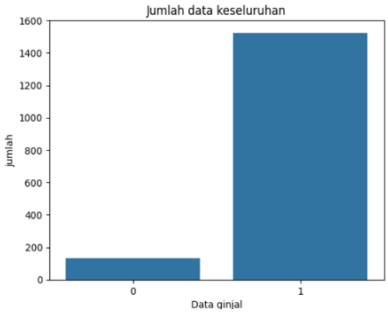
Transformasi data dilakukan dengan normalisasi menggunakan *Min-Max Normalization* karena data yang telah diseleksi memiliki nilai numerik. Hasil normalisasi direpresntasikan pada Tabel 2.

Tabel 2. Hasil Transformasi Data.

BMI	Alcohol Consumption	GFR	….	HealthLiteracy
0.642458	0.255690	0.291906	….	0.987756
0.587277	0.930741	0.388090	….	0.716498
0.895884	0.593896	0.500429	….	0.735806
0.652886	0.801084	0.172639	….	0.663224
…….	……..	……..	….	……
0.987321	0.067519	0.625275	….	0.204396
0.556423	0.167760	0.981935	….	0.032032
0.277141	0.791359	0.039837	….	0.493594
0.397853	0.648227	0.511985	….	0.031850

3.5 *Class Imbalance*

Pada tahap ini dilakukan penyeimbangan data karena data 0 dan 1 jumlah nya tidak seimbang.




```
smote = SMOTE(random_state=42)
X_resampled, y_resampled = smote.fit_resample(X_scaled, y)

print("Jumlah data asli:", len(y))
print("Jumlah data setelah SMOTE:", len(y_resampled))
print("Class distribution after oversampling:")
print(pd.Series(y_resampled).value_counts())
```

```
Jumlah data asli: 1659
Jumlah data setelah SMOTE: 3048
Class distribution after oversampling:
Diagnosis
1    1524
0    1524
Name: count, dtype: int64
```

Gambar 4. *Class Imbalance.*

Gambar 4., menunjukkan bahwa kelas data 1 memiliki jumlah lebih besar dibandingkan dengan data pada kelas 0, yakni kelas data 1 berjumlah 1524 data, dan kelas data 0 berjumlah 135 data. Penanganan ketidakseimbangan data kemudian dilakukan menggunakan metode SMOTE yang akan membuat kelas data 0 memiliki nilai yang seimbang dengan kelas data 1.

3.6 Seleksi Fitur *Information Gain*

Tahap ini melakukan seleksi fitur dengan menghitung nilai *information* setiap fitur. Hasil perhitungan dengan *threshold* 0.3 terpilih 30 fitur, fitur yang terseleksi dapat dilihat pada Tabel 3.

Tabel 3. *Selection Feature Information Gain*.

No	Fitur	Information Gain
1	BMI	0.4070
2	AlcoholConsumption	0.4070
3	PhysicalActivity	0.4070
4	DietQuality	0.4070
5	SleepQuality	0.4070
6	FastingBloodSugar	0.4070
7	HbA1c	0.4070
8	SerumCreatinine	0.4070
9	BUNLevels	0.4070
10	GFR	0.4070
11	ProteinInUrine	0.4070
12	ACR	0.4070
13	SerumElectrolytesSodium	0.4070
14	SerumElectrolytesPotassium	0.4070
15	SerumElectrolytesCalcium	0.4070
…	……	……...
30	HealthLiteracy	0.4070

3.7 Klasifikasi Learning Vector Quantization 3

Pada tahap klasifikasi dilakukan pelatihan dan pengujian, dengan menggunakan perbandingan data rasio 90:10, 80:20, dan 70:30. Pengujian pada data ini dilakukan dengan kriteria nilai *learning rate* yaitu 0.1, 0.3, 0.6 dan 0.9, dengan *minimum learning rate* 0.001, pengurangan α 0.01, nilai *window* 0.2, 0.3, dan 0.4, dan maksimal *epoch* sebesar 100 dengan *LVQ* 3 menggunakan *confussion matrix*.

3.8 Evaluasi dan Pengujian

Pada tahap ini akan dilakukan dengan 5 skenario yaitu pengujian menggunakan model *LVQ* 3, Pengujian menggunakan seleksi fitur *information gain* dan *LVQ* 3, dan pengujian menggunakan teknik *SMOTE*, *information gain* dan *LVQ* 3. Pada Tabel 4 dapat dilihat jumlah skenario pengujian yang akan dilakukan.

Tabel 4. Skenario Pengujian.

No	Skenario			
		α	ε	m
I	*LVQ* 3 tanpa menerapkan seleksi fitur *information gain*	0.1, 0.3, 0.6, 0.9	0.2, 0.3, 0.4	0.2, 0.4
II	*LVQ* 3 dengan *SMOTE* tanpa menerapkan seleksi fitur *information gain*	0.1, 0.3, 0.6, 0.9	0.2, 0.3, 0.4	0.2, 0.4
III	*LVQ* 3 dengan menerapkan seleksi fitur *information gain*	0.1, 0.3, 0.6, 0.9	0.2, 0.3, 0.4	0.2, 0.4

IV	*LVQ* 3 dengan menerapkan seleksi fitur *information gain* dan *SMOTE threshold* 0,3	0.1, 0.3, 0.6, 0.9	0.2, 0.3, 0.4	0.2, 0.4
V	*LVQ* 3 dengan menerapkan seleksi fitur *information gain* dan *SMOTE threshold* 0,7	0.1, 0.3, 0.6, 0.9	0.2, 0.3, 0.4	0.2, 0.4

Hasil akurasi dari 5 skenario pengujian pada pembagian data dengan rasio 90:10, diperoleh akurasi tertinggi pada skenario I sebesar 89.16%. hasil tersebut diperoleh pada pengujian ke-8 berdasarkan penggunaan parameter *learning rate* (α) 0.3 dan *window* (ε) 0.3. Skenario II mendapatkan hasil akurasi tertinggi pada penggunaan *learning rate* (α) 0.1 dan *window* (ε) 0.2, yakni sebesar 82. 95%. Pada skenario III, pengujian dengan menerapkan seleksi fitur *information gain*, menunjukkan hasil akurasi yang di dapat seimbang pada semua percobaan yang dilakukan. Skenario IV dan V, pengujian dengan menerapkan *SMOTE* dan *Information gain*, menunjukkan adanya *overfitting* pada data. Kembali ke pengujian skenario III hasil akurasi tertinggi diperoleh pada pengujian ke-1 dengan parameter *learning rate* sebesar 0.1 dan *window* 0.2 dengan *threshold* 0.3 yakni sebesar 84.57%. Hasil akurasi tertinggi pada skenario IV juga diperoleh pada pengujian ke-1 dengan menerapkan parameter *learning rate* 0.1 dan *window* 0.2 dengan *threshold* sebesar 0.3 yakni 84.59%. Hasil akurasi seluruh pengujian dapat dilihat pada Tabel 5.

Tabel 5. Perbandingan Skenario Pengujian dengan Ratio 90:10.

Uji	I	II	III	IV	V
1	87.35%	82.95%	84.34%	84.57%	84.59%
2	87.35%	82.95%	84.34%	84.57%	84.59%
3	87.35%	82.95%	84.34%	84.57%	84.59%
4	87.35%	82.95%	84.34%	84.57%	84.59%
5	87.35%	82.95%	84.34%	84.57%	84.59%
6	87.35%	82.95%	84.34%	84.57%	84.59%
7	88.85%	82.62%	84.34%	84.57%	84.59%
8	89.16%	82.62%	84.34%	84.57%	84.59%
9	89.16%	82.62%	84.34%	84.57%	84.59%
10	88.85%	82.62%	84.34%	84.57%	84.59%
11	89.16%	82.62%	84.34%	84.57%	84.59%
12	89.16%	82.62%	84.34%	84.57%	84.59%
13	87.35%	81.97%	84.34%	84.57%	84.59%
14	87.95%	81.97%	84.34%	84.57%	84.59%
15	87.95%	81.97%	84.34%	84.57%	84.59%
16	87.35%	81.97%	84.34%	84.57%	84.59%
17	87.95%	81.97%	84.34%	84.57%	84.59%
18	87.95%	81.97%	84.34%	84.57%	84.59%
19	6.63%	50.49%	84.34%	84.57%	84.59%
20	86.75%	81.64%	84.34%	84.57%	84.59%
21	87.95%	81.64%	84.34%	84.57%	84.59%
22	6.63%	50.49%	84.34%	84.57%	84.59%
23	86.75%	81.64%	84.34%	84.57%	84.59%
24	87.95%	81.64%	84.34%	84.57%	84.59%

Dari 5 skenario pengujian dengan pembagian data rasio 80:20. diperoleh akurasi tertinggi pada skenario I. didapatkan hasil akurasi tertinggi sebesar 92.77% pada pengujian ke-13 dengan penggunaan parameter *learning rate* sebesar 0.6. dengan *window* 0.2. Hasil akurasi tertinggi pada skenario II diperoleh dengan menerapkan nilai parameter *learning rate* sebesar 0.3 dan *window* 0.4 yakni menghasilkan akurasi 80.74%. Pada skenario III. hasil akurasi tertinggi diperoleh pada pengujian ke-2 yang mengimplementasikan parameter *learning rate* 0.1. *window* 0.3. dan *threshold* sebesar 0.3 yang menghasilkan nilai akurasi 86.45%. Pada skenario IV hasil akurasi tertinggi sebesar 81.80% diperoleh pada pengujian ke-14 dengan parameter *learning rate* 0.6. *window* 0.3. dan *threshold* 0.3. Hasil akurasi tertinggi pada skenario V diperoleh pada pengujian ke-19 dengan parameter *learning rate* 0.9. *window* 0.2 dan *threshold* 0.7 yakni sebesar 81.97%. Perbandingan akurasi selengkapnya untuk pengujian dengan perbandingan rasio 80:20 dapat dilihat pada Tabel 6.

Tabel 6. Perbandingan Skenario Pengujian dengan Ratio 80:20.

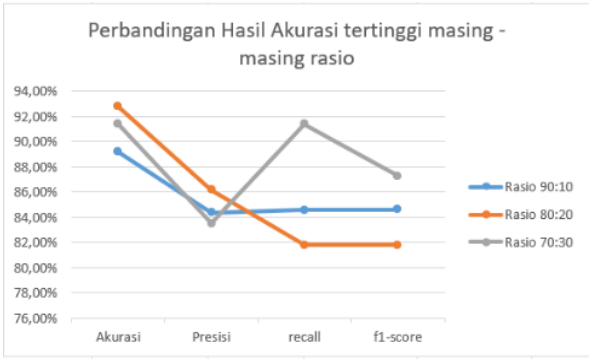
Uji	I	II	III	IV	V
1	90.06%	80.74%	86.45%	81.80%	81.80%
2	90.06%	80.74%	86.45%	81.80%	81.80%
3	90.06%	80.74%	86.45%	81.80%	81.80%
4	90.06%	80.74%	86.45%	81.80%	81.80%
5	90.06%	80.74%	86.45%	81.80%	81.80%
6	90.06%	80.74%	86.45%	81.80%	81.80%
7	90.36%	80.05%	86.14%	81.64%	81.80%
8	90.36%	80.05%	86.14%	81.64%	81.80%

Uji	I	II	III	IV	V
9	90.36%	80.05%	86.14%	81.64%	81.80%
10	90.36%	80.05%	86.14%	81.64%	81.80%
11	90.36%	80.05%	86.14%	81.64%	81.80%
12	90.36%	80.05%	86.14%	81.64%	81.80%
13	92.77%	79.31%	86.14%	81.80%	81.80%
14	89.76%	79.31%	86.14%	81.80%	81.80%
15	89.76%	79.31%	86.14%	81.80%	81.80%
16	92.77%	79.31%	86.14%	81.80%	81.80%
17	89.76%	79.31%	86.14%	81.80%	81.80%
18	89.76%	79.31%	86.14%	81.80%	81.80%
19	92.77%	23.71%	85.84%	81.48%	81.97%
20	88.25%	77.96%	86.45%	81.48%	81.97%
21	87.95%	77.96%	86.14%	81.48%	81.97%
22	92.77%	23.71%	85.84%	81.48%	81.97%
23	88.25%	77.96%	86.45%	81.48%	81.97%
24	87.95%	77.96%	86.14%	81.48%	81.97%

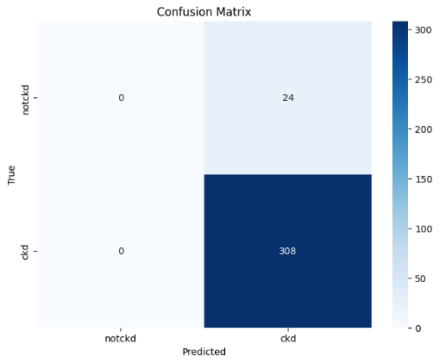
Tabel 7. Hasil Akurasi Tertinggi.

Rasio	Akurasi	Presisi	Recall	F1-Score
Rasio 90:10	89,16%	92,29%	89,16%	90,46%
Rasio 80:20	92,77%	86,14%	92,77%	89,29%
Rasio 70:30	91,37%	83,48%	91,37%	87,24%

Pada Tabel 7., dapat dilihat hasil akurasi tertinggi diperoleh pada skenario I yaitu pengujian *LVQ* 3 tanpa *information gain* pada rasio 80:20 dengan parameter *learning rate* (α) sebesar 0.6, *window* (ε) 0.2 dan *epsilon* 0.2 sebesar 92.77%. Dengan menggunakan data uji sebanyak 332 data, proses klasifikasi berhasil memprediksi data yang sebenarnya ginjal kronis dengan benar sebagai ginjal kronis (*True Positive*) sebanyak 308 data, 0 data ginjal kronis yang salah diprediksi sebagai tidak ginjal kronis (*False Negative*). Hasil prediksi juga menunjukkan adanya data yang sebenarnya tidak ginjal kronis teridentifikasi dengan benar sebagai tidak ginjal kronis (*True Negative*) sebanyak 0 data, serta setidaknya tercatat ada 24 data tidak ginjal kronis yang salah diprediksi sebagai ginjal kronis (*False Positive*). Perbandingan dan *confusion matrix* dari pengujian dapat dilihat pada Gambar 5 dan 6.



Gambar 5. Perbandingan Hasil Akurasi Tertinggi.



Gambar 6. *Confusion Matrix* Akurasi Tertinggi Rasio 80:20

4. KESIMPULAN

Berdasarkan hasil penelitian yang dilakukan dan melalui 5 skenario pengujian, disimpulkan bahwa akurasi tertinggi dihasilkan pada skenario I pengujian *LVQ* 3 tanpa *information gain* dan *SMOTE* sebesar 92.77% pada rasio pembagian data 80:20 dengan parameter *learning rate* (α) 0.6, *window* (ε) 0.2 dan *epsilon* 0.2. Namun, pengujian pada skenario II, III, IV, dan V hasil akurasi yang diperoleh lebih rendah dibandingkan pada skenario I dengan hasil *confussion matrix* nilai *True Positive* dan *True Negative* yang tinggi. Pada skenario II akurasi yang dihasilkan menurun hingga 80.74%, skenario III menghasilkan akurasi sebesar 86.45% akurasi meningkat dikarenakan menggunakan seleksi fitur dengan *threshold* 0.3. Pada skenario IV dan V, kombinasi *SMOTE* dan seleksi fitur digunakan. Dengan nilai *threshold* 0.3, akurasi tertinggi yang dicapai adalah 81,80%. Sementara itu, pada *threshold* 0.7 akurasi tertinggi mencapai 81.97%. Penggunaan Teknik SMOTE berpengaruh dalam menyeimbangkan data dan hasil prediksi model pada *confussion matrix*, teknik ini mampu memprediksi data *True Positive* dan data *True Negative*. Namun, performa model menurun dibandingkan

dengan pengujian tanpa penerapan *SMOTE* dan seleksi fitur. Penelitian selanjutnya disarankan menggunakan variasi parameter dan model evaluasi lain seperti *cross validation*. Dikarenakan adanya *overfitting* data antara 0 dan 1, penggunaan penyeimbangan data disarankan untuk diterapkan.

DAFTAR PUSTAKA

[1] M. J. Baroleh, B. T. Ratag, and F. L. F. G. Langi, "Faktor-Faktor Yang Berhubungan Dengan Penyakit Ginjal Kronis Pada Pasien Di Instalasi Rawat Jalan RSU Pancaran Kasih Manado," Kesmas, vol. 8, no. 7, p. 8, 2019, [Online]. Available: https://ejournal.unsrat.ac.id/index.php/kesmas/article/view/27233

[2] N. Z. Aditama, H. Kusumajaya, and N. Fitri, "Faktor-faktor yang Berhubungan dengan Kualitas Hidup Pasien Gagal Ginjal Kronis," https://www.jurnal.globalhealthsciencegroup.com/index.php/JPPP/article/view/1919/1579. Accessed: Dec. 29, 2024. [Online]. Available: https://www.jurnal.globalhealthsciencegroup.com/index.php/JPPP/article/view/1919/1579

[3] M. Hidayat, F. Motulo, S. Chandra, S. Andamari, J. Sulungbudi, and R. Lesmana, "Analysis of Influencing Factors in Chronic Kidney Disease Incidence in Indonesia," J. Kedokt. dan Kesehat. Indones., vol. 14, no. 3, pp. 296–305, Dec. 2023, doi: 10.20885/jkki.vol14.iss3.art10.

[4] K. Kalantar-Zadeh, T. H. Jafar, D. Nitsch, B. L. Neuen, and V. Perkovic, "chronic kidney disease," Aug. 28, 2021, Elsevier B.V. doi: 10.1016/S0140-6736(21)00519-5.

[5] R. G. Wardhana, G. Wang, and F. Sibuea, "Penarapan Machine Learning Dalam Prediksi Tingkat Kasus Penyakit Di Indonesia," J. Inf. Syst. Manag. e-ISSN, vol. 5, no. 1, pp. 2715–3088, 2023, doi: https://doi.org/10.24076/joism.2023v5i1.1136.

[6] S. Rukiastiandari, L. Rohimah, A. Aprillia, C. Chodidjah, and F. Mutia, "Model Hibrida K-Nearest Neighbors Berbasis Genethic Algorithm untuk Prediksi Penyakit Ginjal Kronis," Infotek J. Inform. dan Teknol., vol. 8, no. 1, pp. 44–55, Jan. 2025, doi: 10.29408/jit.v8i1.27918.

[7] T. S. Saptadi et al., Data Mining: Konsep Data Mining. Yayasan Cendikia Mulia Mandiri, 2024. Accessed: Jan. 05, 2025. [Online]. Available: https://www.academia.edu/124238987/Data_Mining_Konsep_Data_Mining_Oktober_2024_

[8] P. A. Rakhma Devi, "Klasifikasi Penyakit Gagal Ginjal Kronis Dengan Metode KNN (Studi Kasus RS Di Kab Gresik)," JIPI (Jurnal Ilm. Penelit. dan Pembelajaran Inform., vol. 9, no. 3, pp. 1739–1748, Sep. 2024, doi: 10.29100/jipi.v9i3.6226.

[9] Taryadi, E. Yunianto, and Kasmari, "Diagnostik Penyakit Ginjal Kronis Menggunakan Model Klasifikasi Support Vector Machine," IC-Tech Maj. Ilm. Pus. Penelit. dan Pengabdi. Masy., vol. 19, no. 1, pp. 39–44, Apr. 2024, doi: 10.47775/ictech.v19i1.291.

[10] M. Rizal, M. Zakhy Syahaf, S. Rully Priyambodo, Y. Ramdhani, and U. Adhirajasa Reswara Sanjaya, "Optimasi Algoritma Naive Bayes Menggunakan Forward Selection Untuk Klasifikasi Penyakit Ginjal Kronis," NARATIF J. Ilm. Nas. Ris. Apl. dan Tek. Inform., vol. 05, no. 1, pp. 71–80, 2023, doi: 10.53580/naratif.v5i1.200.

[11] Y. Zahara and L. Qadriah, "Implementasi Metode Learning Vector Quantization Untuk Klasifikasi Kesehatan Bayi Dan Ibu Hamil Berbasis Web," JPTIIK J. Pengemb. Tek. Inf. dan Ilmu Komput., vol. 3, no. 2, pp. 1834–1841, 2024.

[12] A. Muzaqi, A. Junaidi, and W. Andi Saputra, "Klasifikasi Status Gizi Pada Lansia Menggunakan Learning Vector Quantization 3 (LVQ 3)," Data Inst. Teknol. Telkom Purwokerto, vol. 2, no. 1, pp. 28–36, 2022, [Online]. Available: http://journal.ittelkom-pwt.ac.id/index.php/dinda

[13] D. marisa Midyanti, R. Hidayati, D. Marisa Midyanti, and S. Bahri, "Diagnosis of Lung Disease Using Learning Vector Quantization 3 (LVQ3)," Sci. J. Informatics, vol. 7, no. 2, pp. 2407–7658, 2020, doi: 10.15294/sji.v7i2.25368.

[14] R. Fatrika, A. Nazir, S. Sanjaya, E. Haerani, S. Kurnia Gusti, and C. Author, "Classifying toddler stunting based on anthropometric data using the learning vector quantization 3 (LVQ 3) method," 2024, doi: 10.54660/. IJMRGE.2024.5.3.921-930.

[15] N. Devian et al., "Prediksi Penyakit Diabetes Dengan Metode K-Nearest Neighbor (KNN) Dan Seleksi Fitur Information Gain," 2024.

[16] M. Atalya, A. Leza, W. Utami, P. Anugrah, and C. Dewi, "Prediksi Prestasi Siswa Smas Katolik Santo Yoseph Denpasar Berdasarkan Kedisiplinan Dan Tingkat Ekonomi Orang Tua Menggunakan Metode Knowledge Discovery In Database Dan Algoritma Regresi Linier Berganda," 2024.

[17] E. Ergi Prayogo and C. Dewi, "Klasifikasi Bidang Keunggulan Mahasiswa menggunakan Metode Backpropagation dan Seleksi Fitur Information Gain (Studi Kasus: Departemen Teknik Informatika Universitas Brawijaya)," 2023. [Online]. Available: http://j-ptiik.ub.ac.id

[18] S. Sathyanarayanan and B. R. Tantri, "Confusion Matrix-Based Performance Evaluation Metrics," no. December, 2024, doi: 10.53555/AJBR.v27i4S.4345