KLASIFIKASI PENYAKIT GINJAL KRONIS MENGGUNAKAN INFORMATION GAIN DAN LVQ

Widya Maulida Putri[1)] , Elvia Budianita[2)] , Fadhilah Syafria[3)] , Iis Afrianty[4)]

[1,2,3,4)] Teknik Informatika Universitas Islam Negeri Sultan Syarif Kasim Riau
email : 12150124092@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)]

Abstraksi

Penyakit Ginjal Kronis (PGK) terjadi ketika fungsi ginjal menurun secara bertahap selama lebih dari tiga bulan tanpa penyebab yang jelas. Penelitian ini bertujuan mengklasifikasikan PGK dengan menggunakan seleksi fitur Information Gain dan Learning Vector Quantization (LVQ). Dataset yang digunakan terdiri dari 1659 data dengan 53 atribut. Proses penelitian meliputi preprocessing data, penerapan SMOTE Oversampling, seleksi fitur Information Gain, dan penerapan model LVQ. Pengujian menghasilkan akurasi tertinggi sebesar 93,37% tanpa seleksi fitur, serta 36 fitur terpilih dengan threshold 0,3 setelah seleksi fitur. Learning rate digunakan antara 0,1 hingga 0,9, min learning rate 0,001, dan pengurangan alpha 0,1. Penggunaan SMOTE dan LVQ meningkatkan nilai presisi, recall, dan f1 score, tetapi akurasi menurun menjadi 84,59%. Hasil ini menunjukkan bahwa metode LVQ efektif dalam klasifikasi penyakit ginjal kronis, membantu ahli identifikasi penyakit ginjal kronis menggunakan data mining dan Jaringan Syaraf Tiruan.

***Kata Kunci* :**
Ginjal Kronis, *Information Gain*, LVQ, Klasifikasi

Abstract

Chronic Kidney Disease (CKD) occurs when kidney function gradually declines over three months without a clear cause. This study aims to classify CKD using Information Gain feature selection and Learning Vector Quantization (LVQ). The dataset used consists of 1659 data with 53 attributes. The research process includes data preprocessing, SMOTE Oversampling application, Information Gain feature selection, and LVQ model application. The test produces the highest accuracy of 93.37% without feature selection, and 36 selected features with a threshold of 0.3 after feature selection. The learning rate used is between 0.1 and 0.9, min learning rate 0.001, and alpha reduction 0.1. The use of SMOTE and LVQ increases the precision, recall, and f1 score values, but the accuracy decreases to 84.59%. These results indicate that the LVQ method is effective in classifying chronic kidney disease, helping experts identify chronic kidney disease using data mining and Artificial Neural Networks.

***Keywords* :**
Chronic Kidney, Information Gain, LVQ, Classification

Pendahuluan

Salah satu organ tubuh yang paling penting bertanggung jawab untuk menjaga komposisi darah dengan mencegah kotoran atau limbah menumpuk di dalam tubuh dan menjaga keseimbangan cairan dalam tubuh adalah ginjal [1]. Penyakit Ginjal Kronis (PGK) didefinisikan sebagai penurunan atau kerusakan fungsi ginjal yang berlangsung selama minimal tiga bulan tanpa identifikasi penyebabnya [2]. Penyakit ginjal kronis lebih sering ditemukan pada lansia, wanita, kelompok ras minoritas, serta individu yang disertai mengalami diabetes melitus dan hipertensi terutama di negara-negara yang tidak siap menghadapi dampaknya [3].

Dalam mengevaluasi pasien PGK Dokter menggunakan dua tes yang sama yaitu tes darah yang memeriksa seberapa baik ginjal menyaring darah, disebut GFR dan tes *urine* untuk memeriksa protein yang dapat masuk ke dalam *urine* saat ginjal rusak [4]. Pencegahan terhadap resiko terkena penyakit ginjal kronis dan perlunya diagnosis tepat yang bertujuan untuk membantu ahli dalam mengidentifikasi faktor yang dikaitkan dengan penyakit ginjal kronis [1]. Salah satu cara lain dalam membantu ahli untuk mengidentifikasi penyakit ginjal kronis dengan melakukan penelitian menggunakan pendekatan data mining dalam Jaringan Syaraf Tiruan (JST) yang menyerupai jaringan biologis karena memiliki kemampuan berbagai perhitungan selama proses pembelajaran [5].

Salah satu penerapan data mining dalam pengklasifikasian ginjal kronis diantaranya penelitian yang dilakukan oleh [2] menggunakan K – *Nearest*

Neighbor memperoleh akurasi yang cukup tinggi 85,83%. Penelitian selanjutnya terhadap algoritma Naïve Bayes mampu mencapai akurasi sebesar 96,43%, dengan sejumlah atribut yang memengaruhi klasifikasi penyakit ginjal kronis, antara lain: usia, tekanan darah, gravitasi spesifik, kadar albumin, kadar gula, sel darah merah, kadar urea, serum kreatinin, natrium, magnesium, hipertensi, diabetes mellitus, gejala penyakit jantung koroner, nafsu makan, pembengkakan pada betis atau kaki, keberadaan sel nanah, serta kondisi anemia [6].

Penelitian mengenai pengklasifikasian penyakit ginjal kronis telah berkembang dari waktu ke waktu dan sudah dianggap penting untuk menemukan model terbaik dalam mengidentifikasi penyakit ginjal kronis dengan indikasi – indikasi yang menjadi penunjangnya, salah satu nya pada penelitian ini akan dilakukan pengklasifikasian menggunakan *Learning Vector Quantization* (LVQ) [7]. *Learning Vector Quantization* (LVQ) adalah metode pelatihan *supervised learning* dan termasuk dalam kategori *competitive layer* [8]. Outputnya akan mempresentasikan kategori atau kelas yang sudah ditentukan sebelumnya, dan jarak antar vector input menentukan kelas yang diperoleh [9].

Menurut penelitian yang dilakukan [10] metode LVQ, akurasi optimal yang dicapai dalam penelitian ini adalah sebesar 97,14%, dengan nilai Mean Square Error (MSE) sebesar 0,028571. Tujuan penelitian berikutnya adalah untuk mengkategorikan persalinan ke dalam dua kategori: normal atau beresiko. Nilai akurasi (α) = 0,1, rasio pembelajaran (c) = 0,1, konstanta pengurangan LR (c) = 0,1, minimum LR = 10-7, dan maxEpoch/iterasi maksimum sebanyak 24 kali menunjukkan nilai akurasi 93,78 % [11].

Dalam penelitian penyakit ginjal kronis menggunakan *dataset Kaggle* yang berjumlah 1659 *dataset* dengan 53 atribut dan 1 atribut label kelas. Sehingga akan digunakan seleksi fitur untuk mengoptimalkan kinerja metode *Learning Vector Quantization* (LVQ) [12]. Seleksi fitur yang akan digunakan adalah *Information Gain*. Adapun beberapa penelitian terkait seleksi fitur *Information Gain* yaitu menghasilkan pengujian menggunakan 6 fitur yang diseleksi 166 data menghasilkan nilai akurasi 96,8%, sedangkan yang yang tidak menggunakan Information Gain atau jumlah fitur 24 menghasilkan nilai akurasi 79,9% [13].

Berdasarkan uraian di atas, maka penelitian penyakit ginjal kronis ini akan menggunakan metode *feature selection Information Gain* dan metode *Learning Vector Quantization* (LVQ). Dengan tujuan menghasilkan klasifikasi yang akurat dalam memprediksi penyakit ginjal kronis, dan penelitian ini diharapan dapat membantu ahli dalam bidang teknologi kesehatan mendiagnosis dengan lebih efisien.

Tinjauan Pustaka

Gangguan pada ginjal dapat memengaruhi sistem dan fungsi organ ginjal, ketika ginjal tidak bekerja dengan optimal maka tubuh biasanya menunjukkan gejala tertentu meskipun gejala tersebut terkadang kurang disadari oleh penderita [2]. Penelitian klasifikasi penyakit ginjal kronis oleh [2] menggunakan metode K-Nearest Neighbor (KNN) memberikan hasil yang cukup tinggi dengan akurasi sebesar 85,83%, sehingga dapat dikatakan metode K-Nearest Neighbor (KNN) cukup baik untuk digunakan dalam pengklasifikasian dataset penyakit ginjal kronis. Namun tingkat keakuratan tergolong masih rendah, penelitian selanjutnya sebaiknya mengkombinasikan metode lain untuk optimasi KNN dalam menemukan nilai k optimal dan untuk meningkatkan akurasi pengklasifikasian penyakit ginjal kronis.
Penelitian selanjutnya yang menggunakan metode klasifikasi Naive Bayes pada dataset penyakit ginjal kronis memperoleh akurasi sebesar 97,00% dan nilai AUC sebesar 99,8%. Setelah optimasi atribut dilakukan menggunakan Particle Swarm Optimization, akurasi klasifikasi Naive Bayes meningkat menjadi 98,75% dengan AUC mencapai 99% [1].

Penelitian yang menggunakan seleksi fitur *Information Gain* oleh [13] menghasilkan nilai akurasi 96,8%, sementara yang tidak menggunakan *Information Gain* menghasilkan akurasi sebesar 79,9%. Algoritma Learning Vector Quantization (LVQ) telah diterapkan oleh [9] pada klasifikasi penyakit ISPA, yang mampu mengenali pola dengan sangat baik, dengan persentase rata-rata akurasi mencapai 96,5% dan akurasi tertinggi sebesar 100%. Parameter yang digunakan adalah learning rate (α) = 0,02, error goal = 0,01, iterasi maksimum 20, dengan perbandingan data latih dan data uji 80:20. Penelitian lain oleh [14] dalam mengidentifikasi penyakit mata merah visus normal memperoleh akurasi sebesar 95%, dengan nilai precision, recall, dan f1-score berturut-turut sebesar 96,3%, 95%, dan 95,2%, yang menunjukkan bahwa sistem dapat berfungsi dengan baik

Information Gain

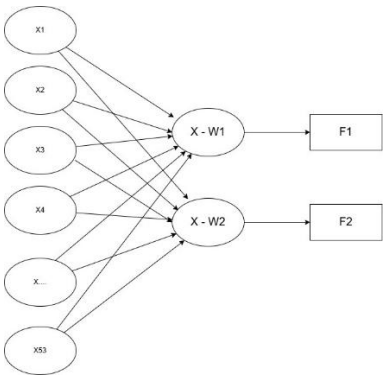
Information Gain ialah sebuah teknik yang digunakan untuk menyaring fitur yang relevan dan mengurangi dimensi fitur dalam data yang akan dianalisis [15]. Adapun tahapan yang dilakukan sebagai berikut:

1) Identifikasi Fitur
2) Pemilihan fitur
3) Perhitungan *Information Gain*
4) Pemilihan fitur terbaik

Learning Vector Quantization (LVQ)

Learning Vector Quantization (LVQ) merupakan suatu metode pembelajaran terarah dan terawasi [8]. Kelas yang dihasilkannya hanya bergantung pada jarak antar vektor input, jika vektor input berdekatan

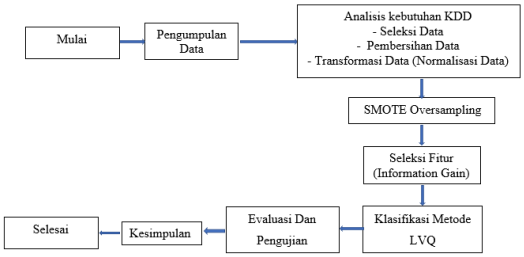
satu sama lain maka lapisan kompetitif akan mengkategorikan kedua vektor input kedalam kelas yang sama [16] Arsitektur Jaringan LVQ pada Gambar 1.



Gambar 1 Arsitektur Jaringan LVQ

Metode Penelitian

Proses ilmiah yang teliti dan cermat untuk mengumpulkan, mengolah, menganalisis, dan menarik kesimpulan secara objektif dikenal sebagai metode penelitian. Tahapan-tahapan dalam penelitian tersebut dapat dilihat pada Gambar 2.



Gambar 2 Metodologi Penelitian

A. Pengumpulan Data

Dataset yang digunakan dalam penelitian ini bersifat sekunder dan diperoleh dari situs Kaggle di https://www.kaggle.com/datasets/rabieelkharoua/chronic-kidney-disease-dataset-analysis/data, yang terdiri dari 1659 data dengan 53 atribut. *Dataset* awal terlihat pada Tabel 1.

Tabel 1 Dataset Awal

Patient ID	Age	Gender	…	…	Diagnosis
1	71	0	…	…	1
2	34	0	…	…	1
3	80	1	…	…	1
4	40	0	…	…	1
5	43	0	…	…	1
…	…	…	…	…	…
…	…	…	…	…	…
1659	34	1	…	…	1

B. Analisis Kebutuhan KDD:

Pada tahapan analisis kebutuhan KDD akan menyeleksi tabel yang saling berkaitan atau berelasi. Hasil dari proses KDD ini akan digunakan sebagai data dasar untuk pemrosesan lebih lanjut. Adapun tahapan dalam analisis kebutuhan KDD.

1) Seleksi Data

Tahapan ini merupakan proses pemilihan atribut agar mengurangi noise dan berkonsentrasi pada karakteristik yang paling penting dalam pemilihan atribut yang akan digunakan pada dataset penyakit ginjal kronis.

2) Pembersihan Data

Tahapan ini dilakukan proses pembersihan data (data *cleaning*). Setelah atribut yang relavan dipilih, data yang bernilai *missing value* akan ditangani untuk memastikan fitur siap digunakan dalam pelatihan model.

3) Transformasi Data

Dataset yang akan digunakan sudah dalam bentuk numerik, tetapi akan dilakukan penskalaan karena terdapat beberapa data yang tidak berada dalam rentang 0 hingga 1. Salah satu jenis transformasi data adalah normalisasi data. Metode *Min-Max Scaler* digunakan untuk menentukan hal ini dari 53 atribut terdapat 36 atribut yang akan dirubah menjadi rentang 0 – 1. Adapun rumus yang digunakan dalam persamaan (1).

$$x_n \frac{x_0 - x_{min}}{x_{max} - x_{min}} \qquad \text{……(1)}$$

Keterangan:
Xn = adalah hasil normalisasi data
X0 = adalah nilai data asli
Xmin = nilai terkecil dari fitur
Xmax = nilai terbesar dari fitur

C. SMOTE Oversampling

Metode SMOTE digunakan untuk menghasilkan data sintetis pada dataset kelas minoritas (dalam hal ini kelas 0), dengan tujuan meningkatkan jumlah instance pada kelas tersebut sehingga setara dengan jumlah instance pada kelas mayoritas (kelas 1) [17]. Langkah ini diambil untuk mencapai keseimbangan antara data kedua kelas.

D. Seleksi Fitur *Information Gain*

Dalam upaya mengidentifikasi dan memilih fitur yang paling informatif untuk meningkatkan akurasi dan efisiensi model, serta mengurangi dimensi data dengan fokus pada fitur yang paling berpengaruh terhadap hasil prediksi, penelitian ini menerapkan metode *Information Gain* untuk mendukung proses pemilihan fitur.

Perhitungan *information gain* mencari selisih antara *entropy dataset* sebelum dan sesudah pembagian [18]. Dalam proses seleksi fitur, langkah awal yang

dilakukan adalah menghitung nilai entropy. Entropy digunakan sebagai ukuran tingkat ketidakpastian suatu kelas berdasarkan probabilitas kemunculan atribut tertentu [19]. Langkah – Langkah perhitungan *Information Gain* adalah sebagai berikut [20].

1. Mengitung nilai *entropy. Entropy* adalah ukuran ketidakpastian kelas yang memanfaatkan kemungkinan peristiwa atau atribut tertentu.

$$Entropy(S) = \sum_{i}^{n} = 1 \; - pi \; log2 \; pi \qquad \ldots\ldots(2)$$

2. Melakukan perhitungan *information gain* rumus:

$$\boldsymbol{Gain\ (S, A) = Entropy\ (S) - \sum_{values\ (A)} \frac{|Sv|}{|S|}\ Entropy\ (Sv)} \qquad \ldots(3)$$

E. *Learning Vector Quantization* (LVQ)

Salah satu tujuan utama dalam algoritma *Learning Vector Quantization* (LVQ) adalah untuk menemukan nilai bobot yang optimal dalam mengelompokkan vektor input ke dalam kelas tujuan yang telah ditentukan selama pembentukan jaringan LVQ. Sementara itu, algoritma pengujiannya bertujuan untuk menghitung nilai output atau kelas vektor yang paling dekat dengan vektor input. Proses ini dapat disamakan dengan proses pengelompokan [21].

Adapun tahapan – taha pan pada metode LVQ:

1. Tetapkan bobot awal dari unit ke-j terhadap kelas ke-i yang mempresentasikan unit input ke-j (Wij), maksimum epoh (MaxEpoch) yang digunakan, parameter learning rate (α), pengurangan learning rate (Dec α), dan menggunakan minimal learning rate (Min α).
2. Masukkan: data input Xii dan kelas target Tk.
3. Tetapkan kondisi awal untuk $Epoch = 0$.
4. Kerjakan $Epoch$ sampai $MaxEpoch$ jika ($Epoch < MaxEpoch$) atau ($\alpha > Eps$).
5. Tentukan nilai jarak minimum dengan Euclidean Distance:

$$d_j = \sqrt{\sum_{j=1}^{j} (\boldsymbol{x_i} - \boldsymbol{c_{i,j}})^2} \qquad \ldots\ldots\ldots\ldots(4)$$

Keterangan:
di,j : jarak euclidean distance vector i dengan banyak data j.
$\sum$: Sigma atau jumlah dari perhitungan input vector dengan codebook vector.
xi,j : Input Vector ke-i,j
ci,j : Codebook Vector ke- i,j

Nilai di,j akan direpresentasikan sebagai kelas jarak minimum yaitu cw dari array C dengan baris dan kolom $i\ x\ j$ dimana i adalah total codebook vectors ke-i dan j adalah total neuron ke-j. Sedangkan untuk w, merupakan indeks pemenang dari setiap iterasi yang dilakukan

6. Perbaiki/ Perbaharui/Update W_j dengan ketentuan:
   - Jika T = Cj maka: Wj (baru) = Wj (lama) + α (Xi - Wj (lama)) ……(5)
   - Jika T ≠ Cj maka: Wj (baru) = Wj (lama) – α (Xi - Wj (lama)) ……(6)

   Keterangan :

   $baru$): Bobot baru terhadap pembaharuan yang dilakukan.

   ($lama$): Bobot lama hasil pembaharuan sebelumnya.

7. Pengurangan Learning Rate (α), dengan rumus:

$$(\alpha) = \alpha * 0{,}1 \qquad \ldots\ldots\ldots\ldots..(7)$$

8. Penambahan epoch, dengan rumus:

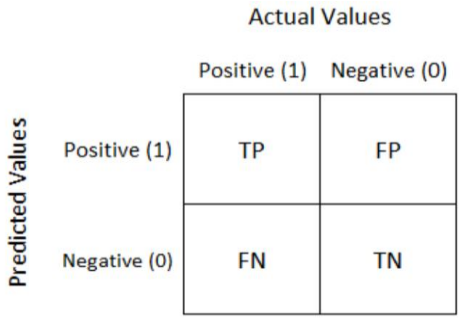
$$Epoch = Epoch + 1 \qquad \ldots..(8)$$

9. Setelah pembaharuan bobot dilakukan, maka diperoleh bobot akhir jika sudah mencapai *max epoch* selanjutnya dapat digunakan untuk proses testing.

F. Evaluasi dan Pengujian

Pengujian dalam penelitian ini dilakukan terhadap beberapa model, yaitu model LVQ, seleksi fitur *Information Gain* dan LVQ, SMOTE dan LVQ, seleksi fitur *Information Gain*; SMOTE dengan threshold 0.3 dan LVQ, serta seleksi fitur *Information Gain*; SMOTE dengan threshold 0.3 dan LVQ dengan berbagai kombinasi learning rate. Evaluasi hasil pengujian menggunakan *confusion matrix* untuk menilai kinerja model klasifikasi.

Dalam confusion matrix, terdapat empat istilah utama yang digunakan untuk merepresentasikan hasil dari proses klasifikasi yang dilakukan yaitu *True Positif* (TP) dimana jumlah data positif yang yang terklasifikasi dengan benar oleh sistem, *True Negatif* (TP) jumlah data negatif yang terklasifikasi dengan benar oleh sistem, *False Positif* (FP) jumlah data positif namun terklasifikasi salah oleh sistem, dan *False Negatif* (FN) jumlah data negatif namun terklasifikasi salah oleh sistem [22]. *Confusion matrix* dapat dilihat pada Gambar 3.



Gambar 3 *confussion matrik*

Maka hasil dari confussion matrix dapat dianalisis dengan perhitungan sebagai berikut;

1. Akurasi adalah ukuran yang menunjukkan seberapa baik model dalam mengklasifikasikan data secara keseluruhan.

$$\boldsymbol{Akurasi} = \frac{\boldsymbol{TN+TP}}{\boldsymbol{TP+TN+FP+FN}} \quad \text{.....(9)}$$

2. Presisi menunjukkan proporsi data yang diprediksi sebagai kelas positif dan memang benar-benar termasuk dalam kelas positif

$$\boldsymbol{Presisi} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FP}} \quad \text{.....(10)}$$

3. Recall menilai seberapa banyak data yang memang positif berhasil dideteksi secara akurat oleh model.

$$\boldsymbol{Recall} = \frac{\boldsymbol{TP}}{\boldsymbol{TP+FN}} \quad \text{.....(11)}$$

4. F1 score digunakan ketika membutuhkan keseimbangan antara presisi dan recall.

$$\boldsymbol{F1\ Score} = \frac{\boldsymbol{2\times(presisi\times recall)}}{\boldsymbol{presisi\times recall}} \quad \text{.....(12)}$$

Hasil dan Pembahasan

A. Pengumpulan Data

Pada penelitian ini, digunakan *dataset* yang terdiri dari 1659 data dengan 53 fitur dan 2 kelas, yaitu ginjal kronis dan tidak ginjal kronis. Seluruh tahapan penelitian dilakukan di *Google Colab* dengan menggunakan bahasa pemrograman Python.

B. Preprocessing Data

1. Seleksi Data: dilakukan penghapusan fitur Patient ID dan DoctorInCharge, maka fitur yang tersisa berjumlah 51 fitur. Dapat dilihat pada Tabel 2.

Tabel 2 Hasil Seleksi Data

Age	Gender	Ethnicity	…	Diagnosis
71	0	0	…	1
34	0	0	…	1
80	1	1	…	1
40	0	2	…	1
…	…	…	…	…
34	1	1	…	1

2. Pembersihan Data: dilakukan pengecekan terhadap *missing value* dan duplikat data. Hasil yang didapat bahwa pada data tidak terdapat *missing value* maupun duplikat data. Dilihat pada Gambar 4.

```
[6] # Duplicate data information
    data.duplicated().sum()

    np.int64(0)

    # Mengecek Missing Values
    print("\nMissing Values:")
    print(data.isnull().sum())

    # Menampilkan total missing values
    total_missing = data.isnull().sum().sum()
    print("\nTotal Missing Values:", total_missing)

    Missing Values:
    Age                        0
    Gender                     0
    Ethnicity                  0
    SocioeconomicStatus        0
    EducationLevel             0
    BMI                        0
    WaterQuality               0
    MedicalCheckupsFrequency   0
    MedicationAdherence        0
    HealthLiteracy             0
    Diagnosis                  0
    dtype: int64

    Total Missing Values: 0
```

Gambar 4 Hasil pengecekan duplikat data dan *missing value*

3. *Transformasi* Data: dilakukan penskalaan pada 36 fitur dengan menggunakan normalisasi *Min-Max Scaler* agar nilainya berada dalam rentang 0 hingga 1. Hasil Normalisasi menggunakan persamaan 1. Hasil dari Normalisasi dapat dilihat pada Tabel 3.

Tabel 3 Hasil Normalisasi Data

Age	Ethnicity	Socioeconomic Status	…	Health Literacy
0.728571	0.000000	0.0	…	0.987756
0.200000	0.000000	0.5	…	0.716498
0.857143	0.333333	0.0	…	0.735806
0.285714	0.666667	0.0	…	0.663224
…	…	…	…	…
0.200000	0.333333	0.0	…	0.455404

C. *SMOTE Oversampling*

Pada data penyakit ginjal kronis dilakukan teknik *smote oversampling* untuk kelas *minoritas* (kelas 0), sehingga jumlah data pada kelas tersebut menjadi seimbang dengan jumlah data pada kelas *mayoritas* (kelas 1). Dapat dilihat pada Gambar 5.

```
Jumlah data sebelum SMOTE:
Diagnosis
1    1524
0     135
Name: count, dtype: int64

Jumlah data setelah SMOTE:
Diagnosis
1    1524
0    1524
```

Gambar 5 Hasil *Smote Oversampling*

D. *Information Gain Feature Selection*

Seleksi fitur dilakukan dengan menghitung nilai *entropy* pada persamaan 2 dan mencari nilai *information gain* pada persamaan 3. Maka hasil perhitungan dapat dilihat pada Tabel 4.

Tabel 4 Hasil *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
16	SerumElectrolytesPhosphorus	0.4070
17	HemoglobinLevels	0.4070
18	CholesterolTotal	0.4070
19	CholesterolLDL	0.4070
20	CholesterolHDL	0.4070
21	CholesterolTriglycerides	0.4070
22	NSAIDsUse	0.4070
23	FatigueLevels	0.4070
24	NauseaVomiting	0.4070
25	MuscleCramps	0.4070
26	Itching	0.4070
27	QualityOfLifeScore	0.4070
28	MedicalCheckupsFrequency	0.4070
29	MedicationAdherence	0.4070
30	HealthLiteracy	0.4070
31	SystolicBP	0.0537
32	Age	0.0304
33	DiastolicBP	0.0281
34	EducationLevel	0.0025
35	FamilyHistoryKidneyDisease	0.0021
36	Edema	0.0020
37	Gender	0.0013
38	SocioeconomicStatus	0.0013
39	UrinaryTractInfections	0.0009
40	Diuretics	0.0006
41	Smoking	0.0005
42	FamilyHistoryHypertension	0.0004
43	WaterQuality	0.0002
44	Ethnicity	0.0002
45	OccupationalExposureChemicals	0.0002
46	AntidiabeticMedications	0.0002
47	FamilyHistoryDiabetes	0.0001
48	HeavyMetalsExposure	0.0001
49	Statins	0.0000
50	PreviousAcuteKidneyInjury	0.0000
51	ACEInhibitors	0.0000

E. Klasifikasi Metode LVQ

Proses pelatihan dan pengujian Learning Vector Quantization (LVQ) dilakukan dengan pembagian data latih dan uji menggunakan rasio 90:10, 80:20, serta 70:30. Parameter learning rate yang digunakan antara lain 0,1, 0,3, 0,6, dan 0,9, dengan nilai minimum learning rate 0,001 dan pengurangan alpha sebesar 0,1.

F. Evaluasi dan Pengujian

Pengujian dilakukan dengan 5 tahapan skenario yaitu:

1. Pengujian menggunakan model LVQ
2. Pengujian menggunakan seleksi fitur *information gain* dan LVQ
3. Pengujian menggunakan *SMOTE* dan LVQ
4. Pengujian menggunakan seleksi fitur *information gain*, *SMOTE* dengan *threshold* 0.3 dan LVQ
5. Pengujian menggunakan seleksi fitur *information gain*, *SMOTE* dengan *threshold* 0.7 dan LVQ

Evaluasi dilakukan setelah tahapan pengujian menggunakan confusion matrix, yang menghasilkan nilai akurasi, presisi, recall, dan F1 Score dengan persamaan (9) hingga persamaan (12). Berikut adalah tabel – tabel hasil pengujian dari setiap skenario.

Tabel 5 Hasil Pengujian model LVQ

Rasio	(α)	Akurasi	Presisi	Recall	F1-Score
90:10	0,1 – 0,9	93.37%	46.69%	50.00%	48.29%
80:20	0,1 – 0,9	92.77%	46.39%	50.00%	48.12%
70:30	0,1 – 0,9	91.37%	45.68%	50.00%	47.74%

Tabel 6 Hasil Pengujian *Information Gain*

Rasio	(α)	Akurasi	Presisi	Recall	F1-Score
90:10	0,1 – 0,9	93.37%	46.69%	50.00%	48.29%
80:20	0,1 – 0,9	92.77%	46.39%	50.00%	48.12%
70:30	0,1 – 0,9	91.16%	45.67%	49.89%	47.69%

Tabel 7 Hasil Pengujian menggunakan *SMOTE* dan LVQ

Rasio	(α)	Akurasi	Presisi	Recall	F1 Score
90:10	0,1	84.59%	84.72%	84.56%	84.57%
	0,3	49.51%	24.75%	50.00%	33.11%
	0,6	49.51%	24.75%	50.00%	33.11%
	0,9	49.51%	24.75%	50.00%	33.11%
80:20	0,1	82.79%	82.85%	82.85%	82.79%
	0,3	51.31%	25.66%	50.00%	33.91%
	0,6	51.31%	25.66%	50.00%	33.91%
	0,9	51.31%	25.66%	50.00%	33.91%
70:30	0,1	81.86%	81.92%	81.88%	81.86%
	0,3	50.60%	25.30%	50.00%	33.60%
	0,6	50.60%	25.30%	50.00%	33.60%
	0,9	50.60%	25.30%	50.00%	33.60%

Tabel 8 Hasil pengujian menggunakan *IG,* SMOTE threshold 0,3 dan LVQ

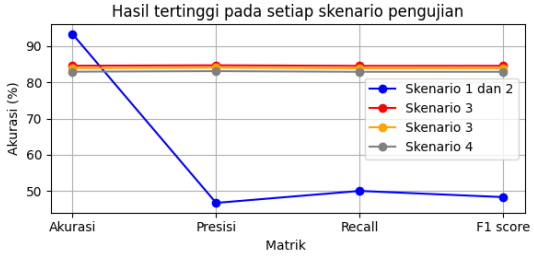
Rasio	(α)	Akurasi	Presisi	Recall	F1 Score
90:10	0,1	83.93%	84.06%	83.90%	83.91%
	0,3	49.51%	24.75%	50.00%	33.11%
	0,6	49.51%	24.75%	50.00%	33.11%
	0,9	49.51%	24.75%	50.00%	33.11%
80:20	0,1	81.15%	81.26%	81.22%	81.15%
	0,3	51.31%	25.66%	50.00%	33.91%
	0,6	48.69%	24.34%	50.00%	32.75%
	0,9	51.31%	25.66%	50.00%	33.91%
70:30	0,1	81.42%	81.53%	81.45%	81.41%
	0,3	50.60%	25.30%	50.00%	33.60%
	0,6	50.60%	25.30%	50.00%	33.60%
	0,9	50.60%	25.30%	50.00%	33.60%

Tabel 9 Hasil pengujian menggunakan IG, SMOTE threshold 0,7 dan LVQ

Rasio	(α)	Akurasi	Presisi	Recall	F1 Score
90:10	0,1	82.95%	83.10%	82.92%	82.92%
	0,3	49.51%	24.75%	50.00%	33.11%
	0,6	50.49%	25.25%	50.00%	33.55%
	0,9	49.51%	24.75%	50.00%	33.11%
80:20	0,1	79.18%	79.45%	79.30%	79.17%
	0,3	51.31%	25.66%	50.00%	33.91%
	0,6	51.31%	25.66%	50.00%	33.91%
	0,9	51.31%	25.66%	50.00%	33.91%
70:30	0,1	79.45%	79.67%	79.50%	79.43%
	0,3	49.40%	24.70%	50.00%	33.07%
	0,6	50.60%	25.30%	50.00%	33.60%
	0,9	50.60%	25.30%	50.00%	33.60%

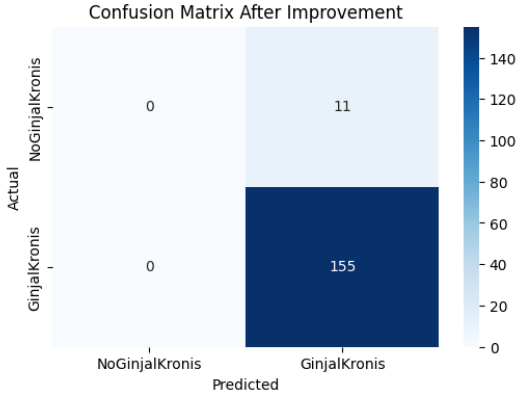
Hasil kesimpulan dari setiap pengujian menunjukkan bahwa akurasi tertinggi diperoleh pada pengujian skenario 1 dan 2, dengan parameter learning rate antara 0.1 hingga 0.9 dan rasio 90:10, menghasilkan akurasi yang sama, yaitu 93,37%. Namun, untuk nilai presisi, hanya didapatkan 46,69%, recall 50,00%, dan F1 score 48,29%.

Pengujian skenario 3 mendapatkan akurasi tertinggi pada ratio 90:10 dengan learning rate 0.9 didapat akurasi tertinggi 84.59% dengan nilai presisi 84.72% recall 84.56% dan F1 score 84.57%.
Pengujian scenario 4 menghasilkan akurasi tertinggi dengan *learning rate* 0.1 ratio 90:10 dengan jumlah fitur yang digunakan sebanyak 36 fitur menghasilkan akurasi 83.93% dengan presisi 84.06% recall 83.90% dan F1 score 83.91%. Sedangkan untuk pengujian 5 menghasilkan akurasi tertinggi dengan learning rate 0.1 ratio 90:10 dengan jumlah fitur yang digunakan sebanyak 33 fitur mendapatkan akurasi 82.95% dengan presisi 83.10% recall 82.92% dan F1 score 82.92%. Hasil akurasi tertinggi dari setiap skenario pada Gambar 6.



Gambar 6 Hasil tertinggi setiap skenario pengujian

Evaluasi dilakukan setelah tahapan pengujian menggunakan *confussion matrik* dengan menganalisis jumlah prediksi yang benar dan salah. Maka didapat hasil pengujian model LVQ dengan akurasi tertinggi model mengklasifikasikan 0 data kelas yang tidak terkena PGK dengan benar (*true positif*), namun 11 data kelas yang tidak terkena PGK yang salah klasifikasi sebagai yang terkena PGK (*false posotif*) dan 155 data kelas yang terkena PGK berhasil diklasifikasikan (*true negative*) dan tidak ada data yang terkena PGK salah klasifikasi (*false negative*). Hasil c*confussion matrik* dilihat pada Gambar 7.



Gambar 7 confussion matrik dengan akurasi tertinggi

Kesimpulan dan Saran

Berdasarkan hasil dari seluruh pengujian skenario dan evaluasi terhadap klasifikasi penyakit ginjal kronis, penelitian ini menyimpulkan bahwa metode *Learning Vector Quantization* (LVQ) mampu menghasilkan akurasi klasifikasi yang tinggi. Dengan parameter *learning rate* antara 0,1 hingga 0,9, minimal *learning rate* 0,001, dan pembagian data

90:10, akurasi mencapai 93,37% pada skenario 1 dan 2. Namun, meskipun akurasi tinggi, model tidak dapat mengenali kelas dengan baik pada *confusion matrix*. Tetapi pada pengujian skenario 3, 4 dan 5 memperoleh hasil akurasi yang lebih rendah dibandingkan dengan skenario 1 dan 2 dengan hasil *confussion matrik* yang lebih baik. Pengujian dengan teknik SMOTE untuk menyeimbangkan data pada *dataset* memberikan dampak signifikan terhadap nilai model *confussion matrik*, khususnya pada seleksi fitur *information gain* dengan *threshold* 0.3, *learning rate* 0.1, pembagian data 90:10 dengan jumlah fitur yang digunakan sebanyak 36 fitur menghasilkan akurasi tertinggi sebesar 83,93% dengan hasil confussion matrik yang baik. Oleh karena itu, hubungan antara seleksi fitur dan *balancing* data terhadap model LVQ dapat menjadi pendekatan efektif dalam pengolahan data medis guna mendukung diagnosis penyakit secara akurat dan efisien. Untuk penelitian selanjutnya, disarankan agar dilakukan pengujian dengan menggunakan teknik validasi yang lain seperti *K-fold Cross-Validation* untuk menghasilkan model pengujian agar meningkatkan kestabilan kinerja model.

Daftar Pustaka

[1] T. Arifin and D. Ariesta, "Prediksi Penyakit Ginjal Kronis Menggunakan Algoritma Naive Bayes Classifier Berbasis Particle Swarm Optimization," *J. Tekno Insentif*, vol. 13, no. 1, pp. 26–30, 2019, doi: 10.36787/jti.v13i1.97.

[2] A. Ariani, K. Kunci-Penyakit, and G. Kronis, "Klasifikasi Penyakit Ginjal Kronis menggunakan K-Nearest Neighbor," *Pros. Annu. Res. Semin.*, vol. 5, no. 1, pp. 148–151, 2019, [Online]. Available: http://archive.ics.uci.edu/ml/datasets/Chronic_Kidney_Dise

[3] C. P. Kovesdy, "Epidemiology of chronic kidney disease: an update 2022," *Kidney Int. Suppl.*, vol. 12, no. 1, pp. 7–11, 2022, doi: 10.1016/j.kisu.2021.11.003.

[4] V. K. Gliselda, "Diagnosis dan Manajemen Penyakit Ginjal Kronis (PGK)," *J. Med. Hutama*, vol. 2, no. 04 Juli, pp. 1135–1141, 2021.

[5] R. Novita Sari, W. Saptha Negoro, R. Perangkat Lunak, and U. Potensi Utama, "Jaringan Syaraf Tiruan Dengan Learning Vector Quantization (Lvq) Untuk Klasifikasi Daun Artificial Neural Network Using Learning Vector Quantization (Lvq) for Leaf Classification," vol. 16, no. 1, pp. 25–34, 2024.

[6] Q. A'yuniyah *et al.*, "Implementasi Algoritma Naïve Bayes Classifier (NBC) untuk Klasifikasi Penyakit Ginjal Kronik," *J. Sist. Komput. dan Inform.*, vol. 4, no. 1, p. 72, 2022, doi: 10.30865/json.v4i1.4781.

[7] S. N. Chotimah and A. R. Rozzaqi, "Klasifikasi Diagnosis Penyakit Ginjal Kronis Dengan Menerapkan Konsep Algoritma Naïve Bayes," *JIPETIKJurnal Ilm. Penelit. Teknol. Inf. Komput.*, vol. 4, no. 1, pp. 8–15, 2023, doi: 10.26877/jipetik.v4i1.16174.

[8] A. R. Aziz, B. Warsito, and A. Prahutama, "Pengaruh Transformasi Data Pada Metode Learning Vector Quantization Terhadap Akurasi Klasifikasi Diagnosis Penyakit Jantung," *J. Gaussian*, vol. 10, no. 1, pp. 21–30, 2021, doi: 10.14710/j.gauss.v10i1.30933.

[9] E. Setyowati and S. Mariani, "Penerapan Jaringan Syaraf Tiruan dengan Metode Learning Vector Quantization (LVQ) untuk Klasifikasi Penyakit Infeksi Saluran Pernapasan Akut (ISPA)," *Prism. Pros. Semin. Nas. Mat.*, vol. 4, pp. 514–523, 2021, [Online]. Available: https://journal.unnes.ac.id/sju/index.php/prisma/article/view/44356

[10] F. Tawakal and A. Azkiya, "Diagnosa Penyakit Demam Berdarah Dengue (DBD) menggunakan Metode Learning Vector Quantization (LVQ)," *JISKA (Jurnal Inform. Sunan Kalijaga)*, vol. 4, no. 3, p. 56, 2020, doi: 10.14421/jiska.2020.43-07.

[11] R. Tantiati, M. T. Furqon, and C. Dewi, "Implementasi Metode Learning Vector Quantization (LVQ) untuk Klasifikasi Persalinan," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 10, pp. 9701–9707, 2019.

[12] N. T. Romadloni and Hilman F Pardede, "Seleksi Fitur Berbasis Pearson Correlation Untuk Optimasi Opinion Mining Review Pelanggan," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 3, pp. 505–510, 2019, doi: 10.29207/resti.v3i3.1189.

[13] M. Ramanda Hasibuan and Marjin, "Pemilihan Fitur dengan Information Gain untuk Klasifikasi Penyakit Gagal Ginjal menggunakan Metode Modified K-Nearest Neighbor (MKNN)," *J. Pengemb. Teknol. Inf. dam Ilmu Komput.*, vol. 3, no. 11, pp. 3659–875, 2019, [Online]. Available: http://j-ptiik.ub.ac.id

[14] G. DAMAYANTI, "Penerapan Metode Learning Vector Quantization (Lvq) Untuk Mengidentifikasi Penyakit Mata Merah Visus Normal," *Repository.Unsri.Ac.Id*, 2022, [Online]. Available: https://repository.unsri.ac.id/75458/66/RAMA_55201_09021281722039_0001108401_0203128701_01_front_ref.pdf

[15] B. S. Prakoso, D. Rosiyadi, H. S. Utama, and D. Aridarma, "Klasifikasi Berita Menggunakan Algoritma Naive Bayes Classifer Dengan Seleksi Fitur Dan Boosting," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 227–232, 2019, doi: 10.29207/resti.v3i2.1042.

[16] M. Melisa, "Implementasi Learning Vector Quantization (LVQ) Dalam Mengidentifikasi Gula Aren Asli dengan Gula Aren Campuran," *Sci-Tech J.*, vol. 1, no. 1, pp. 39–51, 2022, doi: 10.56709/stj.v1i1.18.

[17] A. A. Arifiyanti and E. D. Wahyuni, "Smote: Metode Penyeimbang Kelas Pada Klasifikasi Data Mining," *SCAN - J. Teknol. Inf. dan Komun.*, vol. 15, no. 1, pp. 34–39, 2020, doi: 10.33005/scan.v15i1.1850.

[18] I. Setiawati, A. P. Wibowo, and A. Hermawan, "Pendahuluan Tinjauan Pustaka Penelitian Sebelumnya Klasifikasi," *J. Inf. Syst. Manag.*, vol. 1, no. 1, pp. 13–17, 2019.

[19] D. S. Atmaja, Y. A. Sari, and R. C. Wihandika, "Seleksi Fitur Information Gain pada Klasifikasi Citra Makanan Menggunakan Ekstraksi Fitur Haralick dan YUV Color Moment," *Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 3, no. 2, pp. 1917–1924, 2019.

[20] S. Murni, D. Widiyanto, and C. N. P. Dewi,

"Klasifikasi Citra Penyakit Daun Kopi Arabika Menggunakan Support Vector Machine (SVM) Dengan Seleksi Fitur Information Gain," *Semin. Nas. Mhs. Ilmu Komput. dan Apl.*, pp. 700–709, 2022.

[21] M. Sari, A. C. Nurcahyo, C. Cahyaningtyas, and E. M. Salfarini, "Pengenalan Pola Aksara Dunging Kalbar menggunakan Metode Learning Vector Quantization (Lvq)," *J. Inf. Technol.*, vol. 4, no. 1, pp. 143–149, 2024, doi: 10.46229/jifotech.v4i1.874.

[22] B. P. Pratiwi, A. S. Handayani, and S. Sarjana, "Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix," *J. Inform. Upgris*, vol. 6, no. 2, pp. 66–75, 2021, doi: 10.26877/jiu.v6i2.6552.