

Deteksi Penyakit Gagal Ginjal Kronis Menggunakan Algoritma ID3

Safuan¹

Fakultas Teknik, Universitas Muhammadiyah Semarang

Jl. Kedungmundu Np 18 Semarang, telp/fax 02476740296, Email: safuan@unimus.ac.id

ARTICLE INFO

ABSTRACT

Article history:

Received 30 Maret 2020

Received in revised form 2 April 2020

Accepted 10 April 2020

Available online 25 April 2020

Chronic kidney failure is the failure of kidney function in maintaining metabolism and fluid and electrolyte balance in the body. Chronic kidney disease initially does not show significant symptoms and signs but can develop rapidly into kidney failure. Kidney disease can be prevented and treated if known earlier. One way to find out chronic kidney failure is to detect using data mining.

Iterative Dichotomiser 3 (ID3) algorithm is one of the classification methods and is a type of method that can map or separate two or more different classes. Based on the measurement of performance classification of 80% of training data from 400 data used, it shows that the accuracy value reached 96.25%. It can be concluded that the ID3 Algorithm method is feasible to be used in research predictions for chronic kidney failure.

Keywords : chronic kidney disease, classification , ID3

1. Pendahuluan

Penyakit gagal ginjal kronis diakibatkan karena terjadinya gangguan fungsi ginjal sehingga proses sekresi tubuh tidak bisa dilakukan dengan baik [1]. Gangguan ini dapat berupa menurunnya kemampuan ginjal dalam menyaring elektrolit, menstabilkan cairan dan zat kimia yang ada di dalam tubuh.

Penyakit gagal ginjal kronis (PGGK) menjadi masalah kesehatan masyarakat dunia dengan tingkat prevalens dan insidens gagal ginjal yang tinggi, prognosis yang jelek dan biaya yang tidak murah. Prevalensi PGK bertambah sejalan dengan meningkatnya jumlah penduduk lanjut usia dan penyakit hipertensi serta diabetes melitus. Hasil penelitian yang dilakukan oleh Hill et al [2], mendapatkan prevalensi global PGGK sebesar 13,4%. Menurut hasil survei dari *Global Burden of Disease* tahun 2010, PGGK menjadi penyebab kematian urutan ke-27 di dunia tahun 1990 dan meningkat menjadi peringkat ke-18 pada tahun 2010. Perawatan penyakit ginjal menempati posisi kedua pembiayaan terbesar dari BPJS kesehatan di Indonesia, setelah penyakit jantung [3].

Penyakit ginjal kronis semula tidak menampakkan gejala dan tanda namun dapat berkembang dengan cepat menjadi gagal ginjal. Penyakit ginjal bisa dicegah dan diatasi dan kemungkinan mendapatkan pengobatan yang efektif jika diketahui lebih dini.

Klasifikasi adalah proses penemuan model yang memberikan gambaran dan perbedaan kelas data atau konsep, dengan tujuan memakai model tersebut agar dapat memprediksi kelas objek yang labelnya tidak diketahui [4]. Di bidang klasifikasi, ada banyak cabang yang berkembang yaitu klasifikasi bayesian, jaringan saraf tiruan, pohon keputusan dan algoritma genetika [5]. Di antara cabang-cabang tersebut, pohon keputusan menjadi alat yang sering digunakan dengan beberapa alasan: (a) lebih mudah diinterpretasikan oleh manusia dibandingkan dengan jaringan saraf tiruan atau klasifikasi berbasis bayesian; (b) lebih efisien pada saat menggunakan data pelatihan yang besar dibanding dari jaringan saraf tiruan; (c) tidak membutuhkan pengetahuan sebelumnya; dan, (d) menampilkan akurasi yang lebih baik dibandingkan dengan teknik lain.

Beberapa penelitian yang sudah dilakukan untuk prediksi PGGK diantaranya yaitu Robi Rianto *et al* [6] yang membuat rancang bangun aplikasi pendeteksi PGGK dengan menggunakan algoritma C4.5 yang menghasilkan tingkat akurasi sebesar 91,50%.

Penerapan metode algoritma ID3 untuk prediksi GKG dilakukan oleh Rusda Wajhillah *et al* [7] dengan 10 atribut data yang diambil dari RSUD Sekarwangi di Kabupaten Sukabumi dengan akurasi mencapai 96,08%.

El-Houssainy *et al* [8] membandingkan akurasi empat algoritma yaitu *Probabilistic Neural Networks* (PNN), *Multilayer Perceptron* (MLP), *Support vector machine* (SVM) dan *Radial Basis Function* (RBF) untuk mendeteksi GKG. Hasilnya akurasi tertinggi dicapai oleh MLP sebesar 96,7%.

Pada penelitian ini akan digunakan algoritma *Iterative Dichotomiser 3* (ID3) untuk klasifikasi, karena algoritma ini lebih baik dibanding algoritma pohon keputusan yang lain seperti C4.5, CHAID dan CART [9]. Algoritma ID3 paling banyak digunakan di pohon keputusan [10], terkenal dengan kecepatan tinggi, konstruksi mudah dan kemampuan belajar yang tinggi [11]. Tetapi ID3 mempunyai kelemahan yaitu tidak dapat mengklasifikasikan fitur kontinyu dan rentan terhadap *missing value* [12] sehingga proses klasifikasi tidak dapat dilakukan.

Pada penelitian ini juga akan dilakukan proses penyiapan data (*data preprocessing*) untuk penanganan *missing value* dan konversi numerik ke nominal. Tahapan ini dilakukan untuk mengatasi kelemahan ID3 dan dilanjutkan dengan pembuatan aplikasi untuk menerapkan aturan-aturan yang dihasilkan dalam proses klasifikasi.

2. Metode Penelitian

2.1 Dataset

Data yang di pergunakan dalam penelitian ini adalah data sekunder yang diperoleh dari UCI. Data ini disusun oleh L.Jerlin Rubini (Research Scholar) Alagappa University dan Dr. P.Soundarapandian. M.D.,D.M (Senior Consultant Nephrologist), Apollo Hospitals, Managiri, Madurai Main Road, Karaikudi, Tamilnadu, India.

Tabel 1 Atribut data

No	Atribut	Dekskripsi	Kelas	Tipe	Data Kosong	Nilai
1	age	<i>Age</i>	Prediktor	Numerik	0	<i>age in years</i>
2	bp	<i>Blood Pressure</i>	Prediktor	Numerik	5	<i>in mm/Hg</i>
3	sg	<i>Specific Gravity</i>	Prediktor	Nominal	7	<i>(1.005,1.010,1.015,1.020,1.025)</i>
4	al	<i>Albumin</i>	Prediktor	Nominal	3	<i>(0,1,2,3,4,5)</i>

No	Atribut	Dekskripsi	Kelas	Tipe	Data Kosong	Nilai
5	su	<i>Sugar</i>	Prediktor	Nominal	4	<i>(0,1,2,3,4,5)</i>
6	rbc	<i>Red Blood Cells</i>	Prediktor	Nominal	0	<i>normal, abnormal</i>
7	pc	<i>Pus Cell</i>	Prediktor	Nominal	2	<i>normal, abnormal</i>
8	pcc	<i>Pus Cell Clumps</i>	Prediktor	Nominal	4	<i>present, notpresent</i>
9	ba	<i>Bacteria</i>	Prediktor	Nominal	4	<i>present, notpresent</i>
10	bgr	<i>Blood Glucose Random</i>	Prediktor	Numerik	5	<i>mgs/dl</i>
11	bu	<i>Blood Urea</i>	Prediktor	Numerik	3	<i>mgs/dl</i>
12	sc	<i>Serum Creatinine</i>	Prediktor	Numerik	0	<i>mgs/dl</i>
13	sod	<i>Sodium</i>	Prediktor	Numerik	0	<i>mEq/L</i>
14	pot	<i>Potassium</i>	Prediktor	Numerik	7	<i>mEq/L</i>
15	hemo	<i>Hemoglobin</i>	Prediktor	Numerik	8	<i>gms</i>
16	pcv	<i>Packed Cell Volume</i>	Prediktor	Numerik	10	<i>cells / cumm</i>
17	wc	<i>White Blood Cell Count</i>	Prediktor	Numerik	7	<i>cells / cumm</i>
18	rc	<i>Red Blood Cell Count</i>	Prediktor	Numerik	7	<i>Millions / cmm</i>
19	htn	<i>Hypertension</i>	Prediktor	Nominal	2	<i>yes, no</i>
20	dm	<i>Diabetes Mellitus</i>	Prediktor	Nominal	2	<i>yes, no</i>
21	cad	<i>Coronary Artery Disease</i>	Prediktor	Nominal	2	<i>yes, no</i>
22	appet	<i>Appetite</i>	Prediktor	Nominal	1	<i>good, poor</i>
23	pe	<i>Pedal Edema</i>	Prediktor	Nominal	1	<i>yes, no</i>
24	ane	<i>Anemia</i>	Prediktor	Nominal	1	<i>yes, no</i>
25	class	<i>CKD STAGE</i>	Target	Nominal	0	<i>ckd, notckd</i>

2.2 Tahapan Penelitian

2.2.1 Pengisian *Missing Value* dengan Proses *Replace All Missings* (RAM)

Secara umum penanganan *missing value* dilakukan dengan mengisi data yang kosong tersebut dengan "MISSING" untuk tipe data nominal. Untuk tipe data numerik digantikan dengan nilai rata-rata dari keseluruhan data yang terisi. Jika semua data numerik tersebut kosong semua maka terisi kosong, sedangkan untuk atribut dengan tipe tanggal yang kosong diisi dengan *zero date* (1 Jan 1970), baik hanya sebagian atau semua data.

2.2.2 Merubah Atribut *Numerik* menjadi *Binominal*

Proses Numerik ke Binominal mengubah tipe atribut numerik menjadi tipe binominal (juga disebut biner). Operator ini tidak hanya mengubah tipe atribut yang dipilih tetapi juga memetakan semua nilai atribut ini ke nilai binominal yang sesuai. Atribut binominal hanya dapat memiliki dua nilai yaitu '*true*' atau '*false*'. Jika nilai atribut antara nilai minimal dan maksimal yang ditentukan, maka menjadi '*false*', jika sebaliknya akan menjadi '*true*'. Nilai minimal dan maksimal dapat ditentukan oleh parameter minimal dan maksimal pada masing-masing atribut. Jika nilai dalam satu atribut kosong, maka nilainya akan kosong semua. Batas-batas default keduanya diatur ke 0,0, sehingga hanya 0,0 yang dipetakan ke '*false*' dan semua nilai lain dipetakan ke '*true*' secara default.

2.2.3 Klasifikasi dengan Algoritma ID3

Setelah 2 tahapan pemrosesan data di atas selesai, tahap selanjutnya adalah melakukan klasifikasi dengan menggunakan algoritma ID3. Algoritma ID3 membuat pohon keputusan secara *top-down* (dari atas ke bawah) dengan mengevaluasi semua atribut yang ada menggunakan

information gain (IG), untuk mengukur efektifitas suatu atribut dalam mengklasifikasikan kumpulan sampel data.

Sebelum menentukan IG, ada aturan lain yang harus dipahami yaitu entropi. Dalam ilmu *Information Theory*, entropi sering digunakan sebagai salah satu parameter untuk mengukur keberagaman dari kumpulan data sampel. Jika kumpulan data sampel semakin beragam, maka nilai entropinya semakin besar. Secara matematis, entropi dirumuskan sebagai berikut:

$$Entropy = \sum_{i=1}^c p_i \cdot \log_2 p_i \quad (1)$$

dimana c adalah jumlah nilai yang ada pada atribut target (jumlah kelas klasifikasi). Sedangkan p_i menyatakan jumlah sampel untuk kelas i .

Setelah mendapatkan nilai entropi untuk suatu kumpulan data sampel, maka kita dapat mengukur efektifitas suatu atribut dalam mengklasifikasikan data. Ukuran efektifitas ini disebut sebagai IG. Secara matematis, IG dari suatu atribut A , dituliskan sebagai berikut:

$$Gain(S,A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v) \quad (2)$$

(3)

dimana:

- A : atribut
- V : nilai yang diisikan untuk atribut A
- $Values(A)$: himpunan nilai pada atribut A
- $|S_v|$: jumlah sampel untuk nilai v
- $|S|$: jumlah seluruh data
- $Entropy(S_v)$: entropy untuk sampel-sampel yang memiliki nilai v .

Secara ringkas, tahapan algoritma ID3 dapat digambarkan sebagai berikut:

1. Menghitung nilai IG dari setiap atribut
2. Memilih atribut yang memiliki nilai IG paling besar
3. Membentuk simpul yang berisi atribut tersebut
4. Proses perhitungan IG akan terus dilaksanakan sampai semua data telah termasuk dalam kelas yang sama. Atribut yang telah dipilih tidak diikuti lagi dalam perhitungan nilai IG selanjutnya.

Akurasi yang dihasilkan dihitung berdasarkan *confusion matrix*. Perhitungan pada *confusion matrix* dihitung sesuai dengan prediksi *positif* yang benar (*True Positif*), prediksi *positif* yang salah (*False Positif*), prediksi negatif yang benar (*True Negatif*) dan prediksi *negatif* yang salah (*False Negatif*).

$$Akurasi = \frac{tp + tn}{tp + fp + tn + fn} \quad (3)$$

Semakin tinggi nilai akurasi yang didapat maka semakin baik pula metode yang dihasilkan.

Pembuatan Aplikasi

Pembuatan aplikasi dimaksudkan untuk menerapkan aturan yang diperoleh dari proses pada Rapid Miner ke dalam program Visual Basic, sehingga bisa digunakan oleh pihak yang berkepentingan dengan penentuan penyakit gagal ginjal kronis (misal: tenaga kesehatan).

3. Hasil dan Pembahasan

Deteksi Penyakit Gagal Ginjal Kronis Menggunakan Algoritma ID3 (Safuan)

Eksperimen dilaksanakan dengan menggunakan laptop dengan spesifikasi processor Intel Core i5 @ 2.60 GHz (4 CPUs), 4 GB RAM, dan sistem operasi Microsoft Windows 10 *Professional* 64-bit. *Software* yang digunakan adalah bahasa pemrograman Visual Basic 6.0 dan RapidMiner 9.5.

Data set yang digunakan meliputi atribut numerik berjumlah 11 dan nominal berjumlah 14. Sampel data disediakan sebanyak 400 baris dengan 250 sampel adalah kelompok ckd (positif gagal ginjal kronis) dan 150 sampel kelompok notckd (negatif gagal ginjal kronis).

Tabel 1 Data UCI *Chronic Kidney Disease*

id	age	bp	sg	al	su	Rbc	pc	pcc
0	48.0	80.0	1.02	1.0	0.0		normal	notpresent
1	7.0	50.0	1.02	4.0	0.0		normal	notpresent
2	62.0	80.0	1.01	2.0	3.0	Normal	normal	notpresent
3	48.0	70.0	1.005	4.0	0.0	Normal	abnormal	present
4	51.0	80.0	1.01	2.0	0.0	Normal	normal	notpresent
5	60.0	90.0	1.015	3.0	0.0			notpresent
6	68.0	70.0	1.01	0.0	0.0		normal	notpresent
7	24.0		1.015	2.0	4.0	Normal	abnormal	notpresent
8	52.0	100.0	1.015	3.0	0.0	Normal	abnormal	present
9	53.0	90.0	1.02	2.0	0.0	abnormal	abnormal	present
10	50.0	60.0	1.01	2.0	4.0		abnormal	present
11	63.0	70.0	1.01	3.0	0.0	abnormal	abnormal	present
12	68.0	70.0	1.015	3.0	1.0		normal	present
13	68.0	70.0						notpresent
14	68.0	80.0	1.01	3.0	2.0	normal	abnormal	present
15	40.0	80.0	1.015	3.0	0.0		normal	notpresent
16	47.0	70.0	1.015	2.0	0.0		normal	notpresent
17	47.0	80.0						notpresent

Dilihat dari data pada Tabel 2 bisa diketahui banyak terdapat *missing value* (data kosong) sehingga perlu dilakukan pengisian data melalui proses RAM. Hasil dari proses ini yaitu pengisian data yang kosong sesuai dengan aturan RAM, seperti terlihat pada Tabel 3.

Tabel 2 Data Hasil Proses *Replace All Missings*

id	age	bp	sg	al	su	rbc	pc	pcc
0	48,0	80,0	1,0	1,0	0,0	Missing	Normal	Notpresent
1	7,0	50,0	1,0	4,0	0,0	Missing	Normal	Notpresent
2	62,0	80,0	1,0	2,0	3,0	Normal	Normal	Notpresent
3	48,0	70,0	1005,0	4,0	0,0	Normal	Abnormal	Present
4	51,0	80,0	1,0	2,0	0,0	Normal	Normal	Notpresent
5	60,0	90,0	1015,0	3,0	0,0	Missing	Missing	Notpresent
6	68,0	70,0	1,0	0,0	0,0	Missing	Normal	Notpresent
7	24,0	76,5	1015,0	2,0	4,0	Normal	Abnormal	Notpresent

id	age	bp	sg	al	su	rbc	pc	pcc
8	52,0	100,0	1015,0	3,0	0,0	Normal	Abnormal	Present
9	53,0	90,0	1,0	2,0	0,0	Abnormal	Abnormal	Present
10	50,0	60,0	1,0	2,0	4,0	Missing	Abnormal	Present
11	63,0	70,0	1,0	3,0	0,0	Abnormal	Abnormal	Present
12	68,0	70,0	1015,0	3,0	1,0	Missing	Normal	Present
13	68,0	70,0	471,3	1,0	0,5	Missing	Missing	Notpresent
14	68,0	80,0	1,0	3,0	2,0	Normal	Abnormal	Present
15	40,0	80,0	1015,0	3,0	0,0	Missing	Normal	Notpresent
16	47,0	70,0	1015,0	2,0	0,0	Missing	Normal	Notpresent
17	47,0	80,0	471,3	1,0	0,5	Missing	Missing	Notpresent

Hasil dari proses RAM dilanjutkan dengan merubah atribut bertipe numerik menjadi binominal. Proses ini diperlukan karena algoritma ID3 tidak bisa memroses data dengan tipe numerik. Hasil proses ini terlihat pada Tabel 4.

Tabel 3 Data Hasil Konversi Numerik ke Binominal

id	age	bp	sg	al	su	rbc	pc	pcc
0	True	True	True	True	False	Missing	Normal	Notpresent
1	True	True	True	True	False	Missing	Normal	Notpresent
2	True	True	True	True	True	Normal	Normal	Notpresent
3	True	True	True	True	False	Normal	Abnormal	Present
4	True	True	True	True	False	Normal	Normal	Notpresent
5	True	True	True	True	False	Missing	Missing	Notpresent
6	True	True	True	False	False	Missing	Normal	Notpresent
7	True	True	True	True	True	Normal	Abnormal	Notpresent
8	True	True	True	True	False	Normal	Abnormal	Present
9	True	True	True	True	False	Abnormal	Abnormal	Present
10	True	True	True	True	True	Missing	Abnormal	Present
11	True	True	True	True	False	Abnormal	Abnormal	Present
12	True	True	True	True	True	Missing	Normal	Present
13	True	True	True	True	True	Missing	Missing	Notpresent
14	True	True	True	True	True	Normal	Abnormal	Present
15	True	True	True	True	False	Missing	Normal	Notpresent
16	True	True	True	True	False	Missing	Normal	Notpresent
17	True	True	True	True	True	Missing	Missing	Notpresent

Eksperimen dilanjutkan dengan proses klasifikasi menggunakan algoritma ID3 dengan merubah prosentasi data training. Hasil eksperimen seperti terlihat pada Tabel 5.

Tabel 4 Hasil Ekperimen ID3 dengan Merubah Rasio Data Training

Data training(%)	50	60	70	80	90
Akurasi (%)	95,00	96,25	95,00	96,25	95,00

Hasil eksperimen menunjukkan akurasi tertinggi dicapai pada penggunaan data training sebesar 60% dan 80% yaitu 96,25%. Proses selanjutnya yaitu pembuatan aplikasi dengan menggunakan aturan yang sudah didapatkan dari proses pada Rapid Miner. Aturan-aturan dihasilkan adalah sebagai berikut:

```

al = false
| rbc = MISSING
| | pcc = MISSING: notckd
| | | pcc = notpresent
| | | | pc = MISSING
| | | | | dm = no
| | | | | | age = true
| | | | | | | bp = true
| | | | | | | | sg = true
| | | | | | | | | su = false
| | | | | | | | | | bgr = true
| | | | | | | | | | | bu = true
| | | | | | | | | | | | sc = true
| | | | | | | | | | | | | sod = true
| | | | | | | | | | | | | | pot = true
| | | | | | | | | | | | | | | hemo = true
| | | | | | | | | | | | | | | | pcv = true
| | | | | | | | | | | | | | | | | wc = true
| | | | | | | | | | | | | | | | | | rc = true
| | | | | | | | | | | | | | | | | | | ba = notpresent
| | | | | | | | | | | | | | | | | | | | htn = no
| | | | | | | | | | | | | | | | | | | | | cad = no
| | | | | | | | | | | | | | | | | | | | | | appet = good
| | | | | | | | | | | | | | | | | | | | | | | pe = no
| | | | | | | | | | | | | | | | | | | | | | | | ane = no: notckd
| | | | | | | | | | | | | | | | | | | | | | | | dm = yes: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | pc = abnormal: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | | pc = normal: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | | rbc = abnormal: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | | rbc = normal
| | | | | | | | | | | | | | | | | | | | | | | | | | | pe = MISSING: notckd
| | | | | | | | | | | | | | | | | | | | | | | | | | | pe = no: notckd
| | | | | | | | | | | | | | | | | | | | | | | | | | | pe = yes: ckd
al = true
| htn = no
| | su = false: ckd
| | | su = true
| | | | dm = no
| | | | | pe = no
| | | | | | pcc = notpresent
| | | | | | | rbc = MISSING
| | | | | | | | age = true
| | | | | | | | | bp = true
| | | | | | | | | | sg = true
| | | | | | | | | | | bgr = true
| | | | | | | | | | | | bu = true
| | | | | | | | | | | | | sc = true
| | | | | | | | | | | | | | sod = true
| | | | | | | | | | | | | | | pot = true
| | | | | | | | | | | | | | | | hemo = true
| | | | | | | | | | | | | | | | | pcv = true
| | | | | | | | | | | | | | | | | | wc = true
| | | | | | | | | | | | | | | | | | | rc = true
| | | | | | | | | | | | | | | | | | | | pc = MISSING
| | | | | | | | | | | | | | | | | | | | | ba = notpresent
| | | | | | | | | | | | | | | | | | | | | | cad = no
| | | | | | | | | | | | | | | | | | | | | | | appet = good
| | | | | | | | | | | | | | | | | | | | | | | | ane = no: notckd
{ckd=3, notckd=4}
| | | | | | | | | | | | | | | | | | | | | | | | rbc = abnormal: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | rbc = normal: notckd
| | | | | | | | | | | | | | | | | | | | | | | | | | pcc = present: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | | | pe = yes: ckd
| | | | | | | | | | | | | | | | | | | | | | | | | | | dm = yes: ckd
| htn = yes: ckd

```

Dari aturan-aturan di atas dibuat aplikasi seperti pada Gambar 1

Gambar 1 Screenshot Halaman *Input*

Pada Gambar 1 menunjukkan tampilan aplikasi yang digunakan untuk melakukan percobaan deteksi PGGK dengan menggunakan ID3. Beberapa dari input berbentuk combo box untuk mengisi atribut yang bertipe nominal, hal ini diperlukan untuk menghindari agar pengguna tidak memasukkan nilai yang tidak seharusnya dimasukkan dalam kotak input tersebut.

Pada Gambar 2 menunjukkan tampilan input yang sudah menampilkan hasil prediksi. Pada gambar tersebut menunjukkan bahwa hasil prediksi adalah Ginjal Kronis, yang berarti data yang dimasukan diprediksikan PGGK.

Dalam penerapan pada aplikasi yang dibuat, data yang digunakan sebanyak 40 *record* yang diambil pada 80 data uji secara acak. Hasil yang diperoleh yaitu 20 data positif GGK, 12 data negatif GGK dan 8 *false* positif.

Pemeriksaan Pasien

::: Data Pemeriksaan Pasien :::

Input Data | Pencarian data dan browse data

No Periksa: 0005 Kode Pasien: PG001
 Tanggal: 19/04/2020 Nama Pasien: Berry Sulistyo

Rincian Pemeriksaan

Umur: 11	Bacteria: notpresent	WBC Count: 8600
Blood Pressure: 80	Blood Glucose Random: 0	RBC Count: 0
Specific Gravity: 1,01	Blood Urea: 17	Hypertension: no
Albumin: 3	Serum Creatin: 0,8	Diabetes Militus: no
Sugar: 0	Sodium: 0	Coronary Artery Disease: no
Red Blood Cell: [dropdown]	Potassium: 0	Appetite: good
Pus Cell: normal	Hemoglobin: 15	Padal edema: no
Pus Cells Clumps: notpresent	Packaged Cell Volume: 45	Anemia: no

Baru Simpan Batal Diagnosa Keluar Hasil Diagnosa **Ginjal Kronis**

4. Kesimpulan

Dari hasil penelitian dapat diambil kesimpulan bahwa model prediksi dengan algoritma ID3 ke prediksi data PGGK berhasil, dengan 24 atribut yang ada dan tidak ada atribut yang tidak terpakai.

Berdasarkan pengukuran kinerja klasifikasi dari data *training* menunjukkan bahwa nilai akurasi klasifikasi mencapai 96,25%, sehingga metode algoritma ID3 layak untuk digunakan pada penelitian prediksi data PGGK.

Namun ada beberapa faktor yang perlu dicoba untuk penelitian selanjutnya, agar bisa menghasilkan akurasi yang lebih baik lagi, yaitu:

- Penerapan *pruning* pada ID3 untuk meningkatkan akurasi. *Pruning* adalah proses yang dilakukan untuk menghilangkan beberapa cabang (*branches*) yang tidak dibutuhkan. Cabang atau simpul yang tidak diperlukan dapat menyebabkan ukuran pohon menjadi sangat besar yang disebut *over-fitting*. *Over-fitting* akan menyebabkan terjadinya kesalahan klasifikasi, sehingga tingkat akurasi menjadi rendah
- Penelitian ini dapat dikembangkan lebih lanjut dengan menggunakan algoritma *data mining* lainnya seperti *K-Nearest Neighbor* (KNN), *Support Vector Machine* (SVM), *CART* dan lainnya atau dapat mengoptimalkan parameter dengan menggunakan *Particle Swarm Optimization* (PSO), *Genetic Algorithm* (GA) dan lainnya

Referensi

- [1] K. B. Muntiasuti, L. E. Nugroho, and T. B. Adji, "Model Ontologi Diet Nutrisi Klinis Gagal Ginjal Kronis," *J. ISSN 2085-6350*, pp. 400–404, 2017.
- [2] T. Anothaisintawee, S. Rattanasiri, A. Ingsathit, J. Attia, and A. Thakkinstian, "Prevalence of chronic kidney disease: A systematic review and meta-analysis," *Clin. Nephrol.*, vol. 71, no. 3, pp. 244–254, 2009, doi: 10.5414/CNP71244.

- [3] M. S. Processing, "Forecasting Exchange Rates by Genetic Algorithms Based Back Propagation Network Model," 2009, doi: 10.1109/IIH-MSP.2009.245.
- [4] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, vol. 54, no. Second Edition. 2006.
- [5] C.-J. Tsai, C.-I. Lee, and W.-P. Yang, "A discretization algorithm based on Class-Attribute Contingency Coefficient," *Inf. Sci. (Ny)*, vol. 178, no. 3, pp. 714–731, 2008, doi: 10.1016/j.ins.2007.09.004.
- [6] R. Rianto and N. M. S. Iswari, "Rancang Bangun Aplikasi Pendeteksi Penyakit Ginjal Kronis dengan Menggunakan Algoritma C4.5," *J. Ultim.*, vol. 9, no. 1, pp. 10–18, 2017, doi: 10.31937/ti.v9i1.558.
- [7] R. Wajhillah, "Penerapan Metode Algoritma Id3 Untuk Prediksi Diagnosa Gagal Ginjal Kronis (Studi Kasus: Rsud Sekarwangi Sukabumi)," *Klik - Kumpul. J. Ilmu Komput.*, vol. 6, no. 1, p. 97, 2019, doi: 10.20527/klik.v6i1.211.
- [8] E. H. A. Rady and A. S. Anwar, "Prediction of kidney disease stages using data mining algorithms," *Informatics Med. Unlocked*, vol. 15, no. December 2018, p. 100178, 2019, doi: 10.1016/j.imu.2019.100178.
- [9] J. J. Sheu, "An efficient two-phase spam filtering method based on e-mails categorization," *Int. J. Netw. Secur.*, vol. 9, no. 1, pp. 34–43, 2009.
- [10] C. Jin and L. De-lin, "An Improved ID3 Decision Tree Algorithm," *Proc. 2009 4th Int. Conf. Comput. Sci. Educ.*, pp. 127–130, 2009.
- [11] Liu Yuxun and Xie Niuniu, "Improved ID3 algorithm," in *2010 3rd International Conference on Computer Science and Information Technology*, 2010, doi: 10.1109/ICCSIT.2010.5564765.
- [12] A. Al-ibrahim, "Discretization of Continuous Attributes in Supervised Learning algorithms," *Res. Bull. Jordan ACM - ISWSA*, vol. 7952, no. Iv, 2011.