UNP JOURNAL OF STATISTICS AND DATA SCIENCE
Volume 2 Nomor 1 Tahun 2024 Halaman 90-98.
ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

Comparison of the C5.0 Algorithm and the CART Algorithm in Stroke Classification

Indah Lestari, Dina Fitria*, Syafriandi, dan Admi Salma
Departemen Statistika, Universitas Negeri Padang, Padang, Indonesia
*Corresponding author: dinafitria@fmipa.unp.ac.id

Submitted : 22 Desember 2023
Revised : 02 Februari 2024
Accepted : 16 Februari 2024

ABSTRACT

The C5.0 and CART algorithms are similar in terms of velocity and handling of categorical and numeric type data. However, these two algorithms are differences in terms the CART algorithm is binary and classifies categorical, numerical and continuous response variables resulting in classification and regression decision trees. Meanwhile, the C5.0 algorithm is non-binary and classifies categorical response variables resulting in a classification tree. This research aims to classify the Kaggle's Stroke Prediction Dataset to compare the results of the classification of the both algorithms. The results of the study showed that CART algorithm has a higher value of accuracy and precision, but its recall value is lower than C5.0. The accuracy value of each algorithm is 77.9% and 77.5%, presision is 89.5% and 83.2%, recall is 67% and 71.4%. Overrall,the average classificatian result of the two algorithm show almost the same value. it can be concluded that there is no difference in classification between the C5.0 and CART algorithm.

Keywords: *C5.0 algorithm, CART algorithm, classification, stroke*

I. PENDAHULUAN

Data mining merupakan suatu proses untuk mendapatkan informasi penting dan bermanfaat dari suatu data, salah satu metodenya adalah klasifikasi. Pada metode klasifikasi, dilakukan pengelompokkan objek secara sistematis ke dalam kelas-kelas tertentu berdasarkan karakteristik yang sama. Salah satu algoritma yang populer dan sering digunakan dalam klasifikasi adalah *decision tree*. *Decision tree* dimodelkan dengan satu set keputusan yang disusun dalam bentuk seperti struktur pohon. Proses pada *decision tree* yakni mengubah bentuk data tabel menjadi sebuah model pohon (Bahri & Lubis, 2020). Beberapa algoritma yang dapat dilakukan diantaranya ID3, C4.5, C5.0, CART, dan CHAID.

Algoritma C5.0 adalah pengembangan dari algoritma C4.5 yang sebelumnya dikembangkan dari algoritma ID3, dimana proses penghitungan yang dilakukan hampir sama meliputi nilai *entropy* dan *gain* dan menghasilkan model berupa pohon keputusan klasifikasi. Algoritma CART (*Classfication and Regression Tree*) merupakan suatu algoritma pohon keputusan yang diperkenalkan oleh empat ilmuan pada tahun 1984, yaitu Leo Breiman, Jerome H. Friedman, Richard A. Olshen, dan Charles J. Stone. CART dikembangkan untuk memperoleh pohon keputusan baik regresi ataupun klasifikasi. Algoritma C5.0 dan CART memiliki kesamaan dalam hal kecepatan dan menangani data bertipe numerik maupun kategorik. Amalda dkk (2022) menyatakan algoritma C5.0 memiliki kecepatan dalam membuat model dan mampu mengatasi data dalam jumlah besar. Algoritma C5.0 merupakan salah satu algoritma klasifikasi yang efektif dalam mengolah data dengan variabel numerik maupun kategorik (Sofyan dkk, 2023). Pada algoritma CART, interpretasinya lebih mudah, lebih akurat dan lebih cepat dalam perhitungannya (Pratiwi & Zain, 2014) dan secara efisien menangani *dataset* yang besar, baik data tersebut berupa variabel numerik maupun kategorik (Yomeldi dkk, 2019).

Dibalik kesamaan tersebut, terdapat beberapa hal yang membedakan algoritma C5.0 dan CART. Algoritma C5.0 merupakan algoritma pohon keputusan yang bersifat *non biner*. Berdasarkan Pratiwi & Zain (2014), algoritma C5.0 memperlakukan variabel kontinu sama dengan yang dilakukan oleh algoritma CART, tetapi untuk variabel bertipe kategorik algoritma C5.0 memperlakukan nilai variabel kategorik sebagai pemisah. Sementara itu, pada algoritma CART, Prabawati dkk (2019) menyatakan bahwa CART merupakan suatu metode pohon keputusan biner. Perbedaan lainnya yakni variabel pada algoritma C5.0 bertipe kategorik dan menghasilkan pohon klasifikasi, sedangkan pada algoritma CART variabel respon bertipe kategorik dan numerik atau kontinu. Algoritma CART menurut (Mardika, 2016) memiliki sifat jika variabel respon pada data bertipe kategorik akan menghasilkan pohon klasifikasi

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

(*classification tree*), namun jika variabel respon pada data bertipe numerik atau kontinu akan menghasilkan pohon regresi (*regresision tree*).

Metode klasifikasi banyak digunakan di berbagai bidang seperti kesehatan (Subarkah, 2020), kependudukan (Pratiwi dkk, 2020), dan perekonomian (Yusuf, 2007). Pada bidang kesehatan dapat digunakan untuk mengklasifikasikan penyakit. *Stroke* termasuk salah satu penyakit utama yang dihadapi masyarakat dunia. Suwaryo dkk (2019) menyatakan bahwa setiap tahunnya terdapat sekitar 15 juta orang di dunia mengalami *stroke* dan satu dari enam orang di dunia berisiko mengalami *stroke* dalam hidup mereka. Menurut WHO, pada tahun 2019 tercatat bahwa *stroke* menjadi penyakit yang menyebabkan kematian tertinggi kedua di dunia. Untuk itu, perlu dilakukan analisis klasifikasi agar mengurangi risiko terjadinya *stroke*.

Penelitian yang telah dilakukan oleh Pratiwi dkk (2020) mengenai perbandingan algoritma C5.0 dan CART memperlihatkan bahwa CART lebih baik dalam melakukan klasifikasi. Namun, pada penelitian lain yang dilakukan oleh Patil dkk (2012) dengan membandingkan algoritma C5.0 dan CART menggunakan teknik pemangkasan menunjukkan bahwa C5.0 lebih baik dalam melakukan klasifikasi. Berdasarkan kedua penelitian tersebut, terlihat bahwa hasil perbandingan C5.0 dan CART tidak konsisten. Oleh karena itu, pada artikel ini dilakukan perbandingan algoritma C5.0 dan algoritma CART untuk mengetahui hasil ketepatan klasifikasi kedua algoritma pada data *stroke*. Ketepatan klasifikasi yang baik adalah yang memiliki persentase yang lebih tinggi.

II. METODE PENELITIAN

A. Jenis Penelitian dan Sumber Data

Penelitian ini merupakan penelitian terapan. Data yang digunakan adalah data "*Stroke Prediction Dataset*" yang diperoleh dari *website kaggle* (https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset/download?datasetVersionNumber=1) yang terdiri dari 11 variabel dengan 5110 amatan. Variabel respon yang digunakan adalah *stroke* (Y) dan variabel prediktor yang digunakan adalah jenis kelamin (X_1), usia (X_2), riwayat hipertensi (X_3), riwayat penyakit jantung (X_4), status pernikahan (X_5), jenis pekerjaan (X_6), jenis tempat tinggal (X_7), kadar glukosa darah (X_8), BMI (X_9), dan status merokok (X_{10}).

B. Tahapan Analisis Data

Tahapan analisis data yang dilakukan sebagai berikut.

1. Melakukan *data preprocessing*

   Data preprocessing digunakan agar data sesuai dengan analisis yang dilakukan sehingga mengurangi masalah dalam pemprosesan. Dilakukan pengubahan data mentah menjadi data yang berkualitas dan layak untuk diolah pada tahap analisis selanjutnya. Pada *data preprocessing*, terdapat 5 proses yang dilakukan yakni pembersihan data, optimasi data, transformasi data, integrasi data, dan konversi data (Joshi dan Patel, 2020). Pada penelitian, terdapat 2 proses yang dilakukan yaitu pembersihan data dan transformasi data. Pembersihan data dilakukan dengan penanganan data hilang dan melakukan *undersampling* pada data tidak seimbang.

2. Membagi *data training* dan *data testing*

   Data training berguna untuk melatih algoritma dalam memperoleh model, dan *data testing* berguna dalam melakukan pengujian model yang dihasilkan. Pembagian *data training* dan *data testing* dapat menggunakan perbandingan 70%:30%, 80%:20%, dan 90%:10%. Menurut Joseph (2022), tidak terdapat aturan yang jelas mengenai pembagian *data training* dan *data testing.* Untuk memperoleh rasio optimal, ketiga pembagian rasio tersebut digunakan dalam analisis.

3. Melakukan klasifikasi menggunakan algoritma C5.0

   Klasifikasi pada C5.0 meliputi penghitungan nilai *entropy*, *gain*, dan *gain ratio*. Variabel yang memiliki *gain ratio* tertinggi akan dipilih sebagai *parent* bagi simpul selanjutnya (Kusrini dan Luthfi, 2009). Hal ini menunjukkan bahwa nilai *gain ratio* berguna untuk menentukan simpul yang menjadi pemilah.

   a. Menghitung nilai *entropy*

      Entropy digunakan untuk mengukur keberagaman dari *dataset*. Semakin kecil nilai *entropy* maka semakin baik digunakan dalam mengekstrasi suatu kelas (Setio dkk, 2020). *Entropy* memiliki rentang nilai 0 sampai 1. *Entropy* bernilai 0 artinya data pada simpul tersebut berada di kelas yang sama; *entropy* bernilai 1 artinya data pada simpul berada pada kelas berbeda, dengan proporsi tiap kelasnya sama; dan *entropy* bernilai besar dari 0 dan kecil dari 1 artinya data pada simpul berada pada kelas yang berbeda, dengan proporsi tiap kelasnya tidak sama. Berikut persamaan nilai *entropy*.

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

$$Entropy\ (S) = \sum_{i=1}^{n} -p_i log_2 p_i \quad (1)$$

dimana:
S : himpunan kasus
n : jumlah himpunan kasus
p_i : proporsi Si terhadap S

b. Menghitung nilai *gain*

Gain adalah selisih dari nilai *entropy* total terhadap nilai masing-masing *entropy* dari setiap variabel kriteria dikalikan dengan proporsi nilai variabel kriteria terhadap jumlah data sampel. Puspita dkk (2021) menyatakan bahwa nilai *gain* berfungsi untuk melakukan pengukuran keefektifan tiap atribut/variabel kriteria dalam melakukan klasifikasi data. Persamaan *nilai gain* adalah sebagai berikut.

$$Gain\ (S.A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} Entropy(S_i) \quad (2)$$

dimana :

S : himpunan kasus
A : variabel
n : jumlah partisi atribut A
$|S_i|$: jumlah kasus pada partisi ke-i
$|S|$: jumlah kasus dalam S

c. Menghitung nilai *gain ratio*

Nilai *gain ratio* dihitung untuk setiap variabel. Variabel yang mempunyai nilai *gain ratio* tertinggi akan menjadi simpul. Persamaan *nilai gain ratio* adalah sebagai berikut.

$$Gain\ Ratio = \frac{Gain\ (S,A)}{\sum_{i=1}^{n} Entropy(S_i)} \quad (3)$$

dimana:
$Gain\ (S,A)$: nilai *gain* dari suatu variabel
$\sum_{i=1}^{n} Entropy(S_i)$: jumlah *entropy* dari suatu variabel.

4. Melakukan klasifikasi dengan algoritma CART

CART merupakan metode klasifikasi yang menghasilkan model berupa pohon keputusan. Metode klasifikasi pada CART terdiri dari dua metode yaitu metode pohon klasifikasi dan pohon regresi (Prabawati, 2019). Pohon klasifikasi dapat dibentuk apabila variabel respon bersifat kategorik dan pohon regresi dapat dibentuk apabila variabel respon numerik atau kontinu Adapun langkah analisis algoritma CART adalah sebagai berikut.

a. Pemilihan pemilah

Pemilahan data dilakukan sesuai aturan pemilah dan kriteria *goodness of split*. Setelah data dipilah, himpunan bagian yang diperoleh harus lebih homogen dibandingkan pemilahan sebelumnya. Indeks *gini* (Breiman dkk, 1984:104), merupakan fungsi keheterogenan yang sangat mudah dan cocok diterapkan pada berbagai kasus. Pemilahan terbaik yakni pemilahan yang memiliki nilai penurunan heterogenan yang tertinggi. Fungsi indeks *gini* didefinisikan sebagai berikut.

$$i(t) = \sum_{i,j=1} p(j|t)p(i|t)\,, i \neq j \quad (4)$$

dimana:
p(j|t) : proporsi kelas j pada simpul t
p(i|t) : proporsi kelas i pada simpul t

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

Setelah melakukan perhitungan indeks *gini*, dilakukan penghitungan *goodness of split* ($\phi(s,t)$) untuk mengevaluasi pemilah. *Goodness of split* merupakan penurunan keheterogenan dengan persamaan sebagai berikut.

$$(\phi(s,t)) = \Delta i(s,t) = i(t) - P_L i(t_L) - P_R i(t_R) \quad (5)$$

dimana:

$i(t)$	:	Nilai indeks *gini* pada simpul t
$i(t_L)$	:	Nilai indeks *gini* pada simpul anak kiri
$i(t_R)$	:	Nilai indeks *gini* pada simpul anak kanan
P_L	:	Probabilitas amatan pada simpul kiri
P_R	:	Probabilitas amatan pada simpul kanan

Pemilah terbaik merupakan pemilah yang mempunyai nilai $\Delta i(s,t)$ yang lebih tinggi karena dapat menurunkan heterogenitas lebih tinggi. Nilai $\Delta i(s,t)$ menyatakan perubahan dari keheterogenan pada simpul t karena pemilah s. Apabila kelas pada simpul tidak homogen, tahapan terus diulangi sehingga pohon klasifikasi menjadi suatu konfigurasi dan memenuhi:

$$\Delta i(s^*, t_1) = max_{s \in S} \Delta i(s, t_1) \quad (6)$$

b. Penentuan Simpul Terminal
Simpul terminal terbentuk apabila data pada simpul berada pada kelas yang sama. Menurut Breiman dkk (1984: 178), pengembangan pohon keputusan dihentikan jika pada simpul terdapat pengamatan berjumlah minimum 5.

c. Penandaan Label Kelas
Label kelas diberikan berdasarkan kelas yang memiliki amatan lebih banyak pada simpul terminal, yaitu apabila:

$$p(j_0|t) = max_j p(j|t) = max_j \frac{N_j(t)}{N(t)} \quad (7)$$

dimana:
$p(j|t)$: proporsi kelas *j* pada simpul *t*
$N_j(t)$: jumlah amatan kelas *j* pada *terminal node t*
$N(t)$: jumlah total amatan pada *terminal node t*

5. Pemangkasan Pohon Keputusan
Ukuran pohon yang terlalu besar dapat menimbulkan terjadinya *overfitting* pada pohon. Namun, apabila pohon dibatasi dengan ukuran tertentu akan menyebabkan terjadinya *underfitting*. Oleh karena itu, untuk memperoleh pohon keputusan yang layak dapat dilakukan dengan melakukan pemangkasan pohon keputusan. Pemangkasan pohon dilakukan untuk memperoleh pohon yang lebih sederhana (Lewis, 2000: 6). Pemangkasan dapat dilakukan menggunakan *Cost Complexity Pruning* (CPP).
Berikut persamaan *cost complexity pruning* (Breiman, 1984: 66).

$$R_\alpha(T) = R(T) + \alpha|\bar{T}| \quad (8)$$

dimana:
R(T) : *resubstitution estimate* (proporsi kesalahan pada sub pohon)
α : *complexity parameter* (kompleksitas parameter)
$|\bar{T}|$: ukuran banyaknya simpul terminal pohon T

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

6. Menghitung Ketepatan Klasifikasi
Metode *confusion matrix* akan mengukur kinerja atau tingkat kebenaran dari proses klasifikasi yang meliputi nilai akurasi, presisi, dan sensitivitas/*recall* dengan persentase nilai berkisar dari 0%-100%. Semakin tinggi nilai persentase, maka model dianggap semakin baik.

Tabel 1. *Confusion matrix*

	Prediksi Negatif	**Prediksi Positif**
Aktual Negatif	*True Negatif* (TN)	*False Positif* (FP)
Aktual Positif	*False Negatif* (FN)	*True Positif* (TP)

Berikut penghitungan nilai akurasi, presisi, dan sensitivitas/*recall* berdasarkan tabel *confusion matrix*.

a. Akurasi
Akurasi mengukur keakuratan model dalam melakukan klasifikasi data dengan benar. Akurasi diperoleh dengan membandingkan jumlah amatan yang diklasifikasi secara benar dengan jumlah keseluruhan amatan. Semakin tinggi persentase nilai akurasi maka semakin besar tingkat kedekatan nilai prediksi dan nilai aktual.

$$\text{Akurasi} = \frac{TP+TN}{TP+FN+FP+TN}$$

b. Presisi
Presisi mengukur keakuratan model ketika memprediksi kelas yang positif. Nilai presisi diperoleh dari perbandingan jumlah prediksi benar positif terhadap jumlah total prediksi positif.

$$\text{Presisi} = \frac{TP}{TP+FP}$$

c. Recall
Recall mengukur keakuratan model menemukan kembali sebuah informasi. Nilai *recall* diperoleh dari perbandingan jumlah prediksi benar positif dengan jumlah total amatan yang sebenarnya positif.

$$Recall = \frac{TP}{TP+FN}$$

dimana:
TP : jumlah amatan positif yang benar diprediksi sebagai positif
TN : jumlah amatan negatif yang benar diprediksi sebagai negatif
FP : jumlah amatan negatif yang diprediksi sebagai positif
FN : jumlah amatan positif yang diprediksi sebagai negatif

III. HASIL DAN PEMBAHASAN

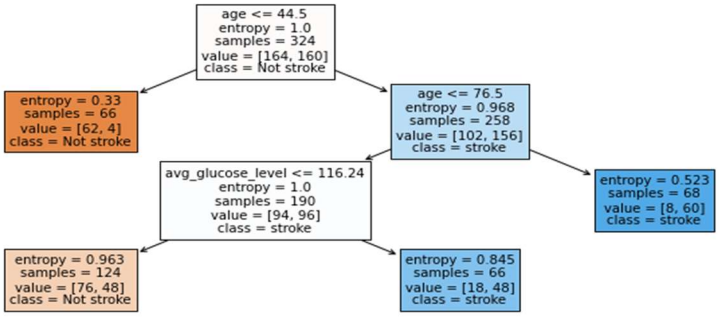
Analisis data dilakukan menggunakan *software python*. Sebelum melakukan analisis data, dilakukan proses *data preprocessing* yang meliputi penanganan data hilang, transformasi data, dan penanganan data tidak seimbang. Penanganan data hilang pada variabel BMI dan status merokok dilakukan dengan melakukan penghapusan amatan yang memiliki data hilang. Analisis data menggunakan CART dan C5.0 dapat menangani data yang bertipe numerik dan kategorik, namun pada analisis menggunakan *python* data kategorik tidak data dapat diolah sehingga harus dilakukan transformasi data. Oleh karena itu, dilakukan tranformasi data kategorik menjadi numerik pada variabel jenis kelamin, status pernikahan, jenis pekerjaan, jenis tempat tinggal, dan status merokok. Selanjutnya, dilakukan *undersampling* untuk mengatasi data tak seimbang pada variabel *stroke* sehingga jumlah data sampel mayoritas diturunkan sampai sama dengan jumlah kelas minoritas dan diperoleh jumlah amatan kategori mengalami *stroke* dan tidak mengalami *stroke* masing-masing sebesar 180 amatan. Pada analisis CART dan C5.0, pembagian *data training* dan *data testing* dilakukan dengan perbandingan 70%:30%, 80%:20%, 90%:10%.

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

Pembentukan pohon keputusan pada C5.0 dilakukan dengan menghitung nilai *entropy* untuk masing-masing kategori pada variabel dan nilai *gain* masing-masing variabel. Dan diperoleh nilai *gain ratio* dari perbandingan nilai *gain* dan *entropy*. Variabel dengan *gain ratio* tertinggi akan dijadikan sebagai simpul. Proses penghitungan *entropy*, *gain*, dan *gain ratio* terus berlanjut sampai amatan pada simpul homogen dan terbentuklah sebuah pohon keputusan. Pada *python*, proses C5.0 dilakukan dengan mengimpor *library DecisionTreeClasiifier* dari *sklearn.tree.* dengan kriteria *entropy*.

Pada pembentukan pohon keputusan, dilakukan *pruning* (pemangkasan) untuk mengatasi *overfitting* pada pohon. Pemangkasan yang dilakukan dengan menggunakan *cost complexity pruning*. Pada analisis *python*, pemangkasan dilakukan menggunakan fungsi *cost_complexity_pruning_path*. Pada fungsi tersebut terdapat nilai *ccp_alpha* (*Cost complexity Pruning – Alpha*) yang merupakan parameter pada fungsi pemangkasan. Pohon keputusan yang dihasilkan pada ketiga perbandingan *data training* dan *data testing* menghasilkan bentuk pohon yang berbeda-beda. Dari pohon-pohon yang dihasilkan, dipilih pohon keputusan yang memiliki nilai ketepatan klasifikasi tertinggi yakni dengan perbandingan 80%:20%. Pohon tersebut dilakukan pemangkasan dan diperoleh pohon keputusan optimal C5.0 pada Gambar 1 dengan menggunakan nilai *ccp_alpha* terbaik sebesar 0.0206.



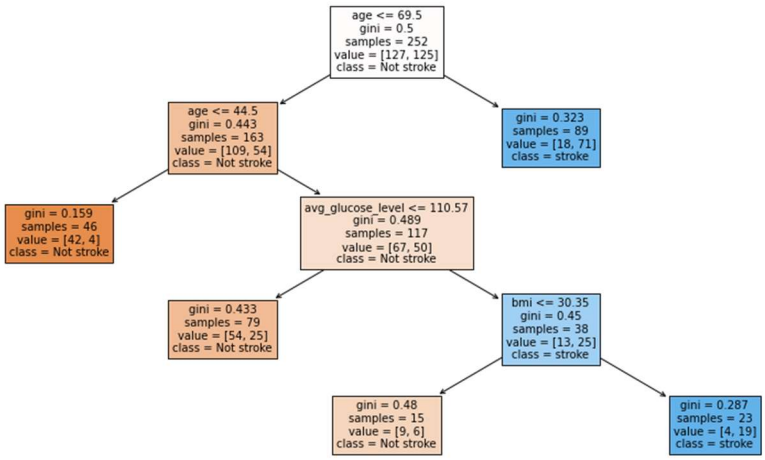
Gambar 1. Diagram Pohon Klasifikasi Optimal Algoritma C5.0

Pada Gambar 1 terdapat 1 *root node*, 2 *decision node*, dan 4 *terminal node*. *Terminal node* ke-1 memberikan hasil bahwa orang yang berusia kecil atau sama dengan 44.5 tahun tidak berisiko mengalami *stroke*. *Terminal node* ke-2 memberikan hasil bahwa orang yang berusia lebih dari 44.5 tahun dan kecil atau sama dengan 76.5 tahun serta memiliki rata-rata kadar glukosa dalam darah kecil atau sama dengan 116.24 mg/dL tidak berisiko mengalami *stroke*. *Terminal node* ke-3 memberikan hasil bahwa orang yang berusia lebih dari 44.5 tahun dan kecil atau sama dengan 76.5 tahun serta memiliki rata-rata kadar glukosa dalam darah besar dari 116.24 mg/dL berisiko mengalami *stroke*. Sementara itu, *terminal node* 4 memberikan hasil bahwa orang yang berusia lebih dari 76.5 tahun berisiko mengalami *stroke*. Sehingga disimpulkan variabel usia dan rata-rata kadar glukosa dalam darah paling berpengaruh terhadap risiko terjadinya *stroke*.

Pembentukan pohon keputusan pada CART dilakukan dengan menghitung nilai indeks *gini* calon simpul kiri dan kanan, dilanjutkan dengan menghitung nilai *goodness of split* tiap pemilah, dimana pemilah dengan nilai tertinggi akan menjadi pemilah utama. Proses menghitung indeks *gini* dan *goodness of split* terus dilakukan dan berhenti apabila semua amatan pada simpul telah homogen dan dilanjutkan dengan pelabelan pada simpul berdasarkan kelas dengan jumlah amatan terbanyak. Sama halnya dengan C5.0, analisis CART dengan *python* dilakukan dengan mengimpor *library DecisionTreeClasiifier* dari *sklearn.tree.* Kriteria yang digunakan adalah nilai *gini*. Pohon keputusan CART menggunakan ketiga pembagian *data training* dan *data testing* menghasilkan bentuk pohon yang berbeda. Pohon yang dipilih adalah pohon keputusan yang memiliki ketepatan klasifikasi tertinggi yakni dengan perbandingan 80%:20%. Pohon tersebut dilakukan *post-pruning* dengan nilai *ccp_alpha* terbaik sebesar 0.0127 sehingga diperoleh pohon klasifikasi optimal CART pada Gambar 2.

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X



Gambar 2. Diagram Pohon Klasifikasi Optimal Algoritma CART

Pada Gambar 2 terdapat 1 *root node*, 3 *decision node*, dan 5 *terminal node* dimana terminal node ke-1 memberikan hasil bahwa orang yang berusia kecil atau sama dengan 44.5 tahun tidak berisiko mengalami *stroke*. *Terminal node* ke-2 memberikan hasil bahwa orang yang berusia kecil atau sama dengan 69.5 tahun dan besar dari 44.5 tahun, serta memiliki rata-rata kadar glukosa dalam darah kecil atau sama dengan 110.57 mg/dL tidak berisiko mengalami *stroke*. *Terminal node* ke-3 memberikan hasil bahwa orang yang berusia kecil atau sama dengan 69.5 tahun dan besar dari 44.5 tahun, memiliki rata-rata kadar glukosa dalam darah besar dari 110.57 mg/dL, serta memiliki BMI kecil atau sama dengan 30.35 kg/m^2 tidak berisiko mengalami *stroke*. *Terminal node* ke-4 memberikan hasil bahwa orang yang berusia kecil atau sama dengan 69.5 tahun dan besar dari 44.5 tahun, memiliki rata-rata kadar glukosa dalam darah besar dari 110.57 mg/dL, serta memiliki BMI besar dari 30.35 kg/m^2 berisiko mengalami *stroke* Sementara itu, *terminal node* 5 memberikan hasil bahwa orang yang berusia lebih dari 69.5 tahun berisiko mengalami *stroke*. Sehingga disimpulkan variabel usia, BMI, dan rata-rata kadar glukosa dalam darah paling berpengaruh terhadap risiko terjadinya *stroke*.

Keakuratan model dalam melakukan klasifikasi dapat dilihat menggunakan *confusion matrix*. Berikut tabel perbandingan ketepatan klasifikasi menggunakan algoritma C5.0 dan CART dalam mengklasifikasikan penyakit *stroke*.

Tabel 2. Perbandingan Ketepatan Klasifikasi Algoritma C5.0 dan CART

Pengujian	Algoritma CART			Algoritma C5.0		
	Akurasi	Presisi	*Recall*	Akurasi	Presisi	*Recall*
70%:30%	0.796	0.867	0.709	0.769	0.826	0.691
80%:20%	0.792	0.818	0.750	0.778	0.794	0.750
90%:10%	0.750	1.000	0.550	0.778	0.875	0.700
Rata-rata	0.779	0.895	0.670	0.775	0.832	0.714

Pada Tabel 2 diperoleh nilai akurasi dan presisi algoritma CART lebih tinggi, sedangkan untuk nilai *recall* algoritma C5.0 memiliki nilai yang lebih tinggi. Rata-rata akurasi algoritma CART dan C5.0 masing-masing diperoleh sebesar 0.779 dan 0.775. Hal ini menunjukkan bahwa terdapat 77.9% dan 77.5% amatan yang diprediksi secara benar dari keseluruhan data. Rata-rata presisi masing-masing algoritma sebesar 0.895 dan 0.832, menunjukkan bahwa terdapat 89.5% dan 83.2% amatan yang benar diprediksi mengalami *stroke* dari keseluruhan amatan yang diprediksi mengalami *stroke*. Dan rata-rata *recall* masing-masing algoritma sebesar 0.670 dan 0.714, menunjukkan bahwa terdapat 67% dan 71.4% data yang benar diprediksi mengalami *stroke* dari keseluruhan amatan yang mengalami *stroke*.

Data kategori mengalami *stroke* dan tidak mengalami *stroke* memiliki proporsi yang telah seimbang. Dari nilai akurasi pada data yang telah diperoleh menunjukkan bahwa model algoritma CART lebih baik dalam memprediksi amatan secara keseluruhan. Pada nilai presisi terlihat bahwa model algoritma CART lebih baik dalam memprediksi

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

kategori *stroke* dari seluruh amatan yang diprediksi *stroke*. Sementara itu, pada nilai *recall* terlihat bahwa model algoritma C5.0 lebih baik dalam memprediksi kategori *stroke* dari seluruh amatan yang mengalami *stroke*.

IV. KESIMPULAN

Berdasarkan hasil dan pembahasan yang disajikan mengenai perbandingan algoritma C5.0 dan CART diperoleh bahwa nilai rata-rata ketepatan klasifikasi pada algoritma CART dan C5.0 tidak jauh berbeda. Rata-rata nilai akurasi pada CART sedikit lebih tinggi dengan nilai masing-masing algoritma sebesar 77.9% dan 77.5%. Hal ini menunjukkan bahwa CART lebih baik dalam melakukan prediksi amatan secara keseluruhan. Rata-rata nilai presisi juga menunjukkan nilai CART lebih tinggi dengan nilai masing-masing algoritma sebesar 89.5% dan 83.2%. Ini berarti bahwa CART lebih baik dalam melakukan prediksi amatan mengalami *stroke* dari keseluruhan prediksi mengalami *stroke*. Namun, pada *recall* algoritma C5.0 memiliki nilai lebih tinggi dengan nilai masing-masing algoritma sebesar 67% dan 71.4%. Hal ini berarti C5.0 lebih baik dalam melakukan prediksi amatan mengalami *stroke* dari keseluruhan amatan yang mengalami *stroke*.

DAFTAR PUSTAKA

Amalda, R. N. (2022). Implementasi Algoritma C5.0 dalam Menganalisa Kelayakan Penerima Keringanan UKT Mahasiswa ITK, *Teorema*, Vol. 7, No. 1, hal. 101-116.

Bahri, S., & Lubis, A. (2020). Metode Klasifikasi Decision Tree untuk Memprediksi Juara English Premier League, *Jurnal Sintaksis*, Vol. 2, No. 1, hal. 63-70.

Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and Regression Tree*. New York: Chapman and Hall.

Joseph, V. R. (2022). Optimal Ratio for Data Splitting, *The ASA Data Sci Journal*, Vol. 15, No. 4, hal. 531-538.

Joshi, A. P. & Patel, B. V. (2020). Data Preprocessing: the Techniques for Preparing Clean and Quality Data for Data Analytics Process, *Oriental Journal of Computer and Technology*, Vol. 13, No. 2-3, hal. 78-81.

Kaggle. Stroke Prediction Dataset. Website: https://www.kaggle.com/datasets/fedesoriano/stroke-prediction-dataset/download?datasetVersionNumber=1

Kusrini & Luthfi, E. T. (2009). Algoritma Data Mining. Yogyakarta: CV. ANDI

Lewis, R. J. (2000). *An Introduction to Classification and Regression Tree (CART) Analysis*. California: UCLA Medical Center.

Mardika, Z. W., Mukid, M. A., & Yasin, H. (2016). Pembentukan Pohon Klasifikasi Biner dengan Algoritma CART: Studi Kasus Kredit Macet di PD. BPR-BKK Purwokerto Utara, *Jurnal Gaussian*, Vol. 5, No. 3, hal. 583-592.

Patil, N., Lathi, R., & Chitre, V. (2012). Comparison of C5.0 & CART Classification Algorithm Using Pruning Technique, *International Journal of Engineering Researh & Technology (IJERT)*, Vol. 1, No. 4, hal. 1-5.

Prabawati, N. I., Widodo, & Duskarnaen, M. F. (2019). Kinerja Algoritma Classification and Regression Tree (CART) dalam Mengklasifikasi Lama Masa Studi Mahasiswa yang Mengikuti Organisasi di Universitas Negeri Jakarta, *Jurnal Pinter*, Vol. 3, No. 2, hal. 139-145.

Pratiwi, F. E., & Zain, I. (2014). Klasifikasi Pengangguran Terbuka Menggunakan CART (Classification and Regression Tree) di Provinsi Sulawesi Utara, *Jurnal Sains dan Seni Pomits*, Vol.3, No. 1, hal. 54-59.

Pratiwi, R., Hayati, M. N., & Prangga, S. (2020). Perbandingan Klasifikasi Algoritma C5.0 dengan CART (Studi Kasus: Data Sosial Kepala Keluarga Masyarakat Desa Teluk Baru Kecamatan Muara Ancalong Tahun 2019), *Jurnal Ilmu Matematika dan Terapan,* Vol. 14, No. 2, hal. 267-278.

Puspita, D., Aminah, S., & Arif, A. (2021). Application of C4.5 Algorithm for Credit Eligibility Prediction. *Journal of Informatics and Telecommunication Engineering,* Vol. 4, No. 2, hal. 1-9.

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144

ISSN(*Print*) : 3025-5511
ISSN(*Online*): 2985-475X

Setio, P. B. N., Saputro, D. R. S., & Winarno, B. (2020). Seleksi Fitur pada Supervised Learning: Klasifikasi Prestasi Belajar Mahasiswa Saat dan Pasca Pandemi COVID-19. *Jurnal Nasional Teknologi dan Sistem Informasi*, Vol. 9, No. 1, hal. 21-32.

Sofyan, F. M. A., Riyandoro, A. P., Maulana, D. F, & Jajam, J. H. (2023). Penerapan Data Mining dengan Algoritma C5.0 Untuk Prediksi Penyakit Stroke . *Jurnal Teknologi Sistem informasi dan Sistem Komputer TGD*, Vol. 6, No. 2, hal. 619-625.

Subarkah, P. (2020). Penerapan Algortima Klasifikasi Classification and Regression Tree (CART) untuk Diagnosis Penyakit Diabetes Retinopathy. *Jurnal MATRIK*, Vol. 19, No. 2, hal. 294-301.

Suwaryo, P. A., Widodo, W. T., & Setyaningsih, E. (2019). Faktor Risiko yang Mempengaruhi Kejadian Stroke. *Jurnal Keperawatan*, Vol. 11, No. 4, hal. 251-260.

Word Health Organization (WHO). 2020. The Top of Causes of Death. Website: https://www.who.int/news-room/fact-sheets/detail/the-top-10-causes-of-death

Yomeldi, H., Azmy, M. R., & Pranita, R. (2007). Analisis Kecenderungan Keterlambatan Pembayaran Pengecekan Kapal di Pelabuhan Regional Riau. *Jurnal Sistem Cerdas*, Vol. 02, No. 02, hal. 92-98.

Yusuf, Y. (2007). Perbandingan Performansi Algoritma Decision Tree C5.0, CART, dan CHAID: Kasus Prediksi Status Resiko Kredit di Bank X. *SNATI*, B-59-B-62.

DOI: https://doi.org/10.24036/ujsds/vol2-iss1/144