Perbandingan Metode *Decision Tree, Naive Bayes,* dan *Induction Rule* untuk Klasifikasi Penyakit Diabetes

Dwy Laila Safitry[-1], Anisa Al Harani[-2], Erin Divayaning[-3], Faiqa Hadya Hanifa[-4], Dheva Fauzia Chema[-5], Iman Paryudi[-6]
Program Studi Teknik Informatika
Fakultas Teknik Universitas Pancasila
Jakarta, Indonesia
dwylailas@gmail.com[1], anisaalharani24@gmail.com[2], erindiva13@gmail.com[3],
faiqahdyhnf@gmail.com[4], chemafau@gmail.com[5], iman.paryudi@univpancasila.ac.id[6]

Abstrak — Teknik klasifikasi digunakan untuk memprediksi suatu nilai dari target variabel kategori. Metode klasifikasi menggunakan algoritma *Decision Tree*, *Naive Bayes*, dan *Induction Rule*. *Decision Tree* merupakan model prediksi menggunakan struktur pohon keputusan yang mengubah data dari proses pengambilan keputusan yang kompleks menjadi lebih sederhana sehingga memudahkan interpretasi informasi, pengambilan solusi, menemukan hubungan, dan identifikasi pola antara faktor-faktor yang mempengaruhi. *Naive Bayes* digunakan untuk mencari probabilitas dalam suatu *class*. *Induction Rule* bertujuan untuk mencari pola yang sering muncul diantara banyak transaksi dan akan menginduksi aturan yang kompleks. Metode-metode ini dapat digunakan sebagai acuan untuk mendiagnosa suatu jenis penyakit. Metode yang terbaik didapatkan dari penggunaan beberapa metode teknik klasifikasi. Penelitian ini mengimplementasikan konsep dan ilmu *data mining* tersebut dalam bidang kesehatan. Menggunakan *dataset* diagnosa penyakit diabetes yang diperoleh dari situs kaggle. Metode pemodelan dilakukan menggunakan *Decision Tree*, *Naïve Bayes*, dan *Induction Rule* melalui aplikasi *Orange*. Pemodelan diuji menggunakan *Confusion Matrix* dan *Cross Validation* untuk menunjukkan perbandingan dari ketiga metode klasifikasi yang diterapkan. Proses pengujian mengevaluasi kinerja pemodelan yang digunakan untuk memperoleh model dengan hasil akurasi yang maksimal. Penelitian ini membahas perbandingan akurasi dari penggunaan ketiga metode tersebut dan menghasilkan sepuluh aturan yang diketahui paling mempengaruhi hasil *outcome* diagnosa penyakit diabetes dari metode yang didapat paling baik dan efektif dalam proses klasifikasi.

Kata kunci: Data mining, Klasifikasi, Decision Tree, Naïve Bayes, Induction Rule, Diagnosa penyakit diabetes.

I. PENDAHULUAN

Data mining merupakan sekumpulan proses yang berguna untuk mengeksplorasi dan mencari nilai berupa informasi dan relasi yang tersimpan dalam suatu data. Penggalian pola informasi dilakukan terhadap data melalui cara mengekstraksi, juga mengetahui pola-pola, guna menghasilkan informasi baru yang lebih bermanfaat, berharga, dan menarik. *Data mining* digunakan dalam pengelolaan data besar dan membantu proses penyimpanan data transaksi. Diperlukan suatu proses data *preprocessing* sehingga tidak akan terjadi kondisi bahwa data berkualitas rendah akan menyebabkan kinerja *data mining* berkualitas rendah [1].

Klasifikasi merupakan teknik dalam *data mining* untuk menelusuri kemudian membangun sebuah model dari suatu data. Klasifikasi adalah proses penemuan model (atau fungsi) yang menggambarkan dan membedakan kelas data atau konsep yang bertujuan agar dapat digunakan untuk memprediksi kelas dari objek yang label kelasnya tidak diketahui [2]. Teknik klasifikasi digunakan untuk memprediksi suatu nilai dari target variabel kategori. Algoritma pada metode klasifikasi (*Decision Tree*, *Naive Bayes*, dan CN-2 *Rule Induction*) digunakan dalam pemodelan data klasifikasi penelitian ini.

Dalam penelitian ini penulis mengimplementasikan konsep dan ilmu data mining dalam bidang kesehatan dengan bantuan *tools*, yaitu aplikasi *Orange*. Dimana penulis meneliti tentang faktor-faktor pemicu penyakit diabetes.

II. DASAR TEORI

A. *Data Mining*

Proses penambangan untuk mendapatkan sebuah informasi dari suatu data merupakan pengertian dari *data mining*. Informasi-informasi yang telah didapatkan ini diperoleh dari berbagai macam proses teknik yang rumit, contohnya seperti menggunakan ilmu matematika, teknik statistik, *machine learning*, dan lainnya. Teknik tersebut yang akan mengidentifikasi, menganalisis, dan mengekstraksi informasi yang dapat dimanfaatkan dari suatu *database*. Hal yang penting dalam teknik *data mining*, yaitu aturan untuk menemukan pola frekuensi tinggi antar himpunan *itemset* yang disebut dengan fungsi *Association Rules*. *Data mining* sendiri berawal dari *machine learning* dan statistika, tetapi kemudian merambah ke bidang lain dalam ilmu komputer, yaitu seperti biologi, lingkungan, jaringan, finansial, dan lain sebagainya. Implementasi dari *data mining* juga

digunakan oleh berbagai bidang, sebagai contoh yaitu implementasi dalam bidang bisnis, bidang telekomunikasi, serta bidang pendidikan [3].

B. *Preprocessing Data*

Preprocessing data menurut [4], merupakan tahap awal yang harus dilakukan pada *data mining*. *Preprocessing data* memiliki tujuan untuk mempersiapkan data mentah sebelum diolah dan diproses menjadi data lain, memproses data untuk mendapatkan hasil yang lebih akurat, membuat nilai data menjadi lebih kecil tanpa merubah informasi yang terdapat didalamnya, serta pengurangan waktu perhitungan untuk *large scale problem*. *Preprocessing data* terdiri dari *data cleaning, data integration, data reduction,* dan *data transformation*. Menurut [5], *preprocessing data* memiliki tahap terakhir yaitu menghilangkan beberapa atribut yang tidak digunakan dalam tahap pemodelan data.

C. *Imputation*

Imputasi data dilakukan pada data yang isinya hilang atau memiliki masalah berdasarkan hasil proses pengecekan kembali kekonsistenan data dan kekonsistenan isi antar variabel. Maka dari itu, pengisian data yang hilang dengan nilai-nilai yang konsisten dengan data terdekat merupakan pengertian dari *imputation.* Secara umum, *imputation* dapat dilakukan kepada *outlier* sebagai salah satu upaya untuk meningkatkan kualitas suatu data [6]. Metode imputasi terbagi menjadi dua jenis, yaitu metode imputasi berbasis statistik dan metode imputasi berbasis *machine learning* [7]. Beberapa contoh dari metode imputasi berbasis *statistic*, yaitu *mean imputation*, *hot-deck imputation*, *metode regression*, dan *multiple imputation*. Metode *hot-deck imputation* sendiri merupakan salah satu metode imputasi yang sering digunakan. Metode ini adalah penyempurnaan dari metode *mean imputation* [8].

D. *Outlier*

Sering ditemukannya suatu nilai yang tidak konsisten dan berbeda dari data lain serta tidak mencerminkan karakteristik dari data secara umum pada *dataset* dinamakan *outlier*. *Outlier* atau anomali merupakan sehimpunan data yang dianggap memiliki karakter yang berbeda jika dibandingkan dengan mayoritas data lainnya. Menurut [9], tidak semua outlier dapat dihilangkan, karena terkadang *outlier* memiliki nilai yang penting dan berpengaruh dalam kumpulan objek. Analisis *outlier* juga dikenal dengan nama deteksi anomali atau deteksi deviasi (nilai atributnya objek tersebut, signifikan berbeda dengan nilai atribut objek lainnya). Terdapat beberapa penyebab data *outlier*, contohnya seperti:

1. Data berasal dari kelas yang berbeda.
2. Kesalahan saat pengumpulan data.
3. Variasi natural dari data itu sendiri.

E. *Select Relevant Feature*

Select Relevant Feature merupakan proses penghapusan *features* yang berlebihan dan tidak relevan dengan *dataset* yang sebenarnya. *Select Relevant Feature* digunakan untuk mempersingkat waktu dalam proses mengklasifikasi data, juga dapat meningkatkan akurasi karena *features* yang tidak relevan dapat memperburuk data serta mempengaruhi akurasi klasifikasi secara negatif. Dengan *Select Relevant Feature* dapat meningkatkan pemahaman dan biaya penanganan data agar lebih kecil [10].

F. *Confusion Matrix*

Confusion matrix merupakan metode yang digunakan untuk melakukan perhitungan akurasi pada konsep *data mining* [11]. *Confusion matrix* merupakan sebuah tabel yang menyatakan jumlah data uji yang benar diklasifikasikan dan jumlah data uji yang salah diklasifikasikan [12]. Berikut ini merupakan tabel dari *confusion matrix*:

		Observed	
		True	*False*
Predicted Class	*True*	*True Positive* (TP)	*False Positive* (FP)
	False	*False Negative* (FN)	*True Negative* (TN)

Tabel 1. Tabel *confusion matrix*.

Keterangan:

1. TP adalah *True Positive*, yang merupakan jumlah data positif yang terklasifikasi dengan benar oleh sistem.
2. TN adalah *True Negative*, yang merupakan jumlah data negatif yang terklasifikasi dengan benar oleh sistem.
3. FP adalah *False Positive*, yang merupakan jumlah data positif tetapi terklasifikasi dengan salah oleh sistem.
4. FN adalah *False Negative*, yang merupakan jumlah data negatif tetapi terklasifikasi dengan salah oleh sistem.

G. *Accuracy*

Accuracy merupakan persentase dari total data uji coba yang sudah benar dalam identifikasi [13]. *Accuracy* merupakan keakuratan data ketika sistem menerima input kemudian diolah menjadi sebuah informasi [14]. Dimensi dari *accuracy* mencakup pada ketepatan dari data yang dihasilkan oleh sistem informasi, contohnya seperti sistem informasi menyediakan informasi yang akurat, integritas dan keutuhan data yang dihasilkan, dan keterbatasan hak akses pada masing-masing *user*. Berikut ini adalah rumus *accuracy*:

$$A = \left(\frac{a+d}{total\ sample}\right)\ x\ 100\%$$

H. *Precision*

Precision merupakan bagian data yang diambil sesuai dengan informasi yang dibutuhkan [13]. Menurut [15], *precision* adalah sebuah ukuran yang mengukur tingkat proporsi jumlah dokumen yang dapat ditemukan kembali oleh sebuah proses pencarian dan dianggap relevan untuk kebutuhan pencarian informasi atau rasio jumlah dokumen relevan yang ditemukan dengan total jumlah dokumen yang ditemukan. Rumus *precision*, yaitu:

$$P = \left(\frac{d}{b+d}\right) x\ 100\%$$

I. *Recall*

Recall merupakan pengambilan data yang berhasil dilakukan terhadap bagian data yang relevan dengan *query* [13]. Menurut [15], *recall* merupakan proporsi jumlah dokumen yang dapat ditemukan kembali oleh sebuah proses pencarian informasi. Berikut ini adalah rumus *recall*:

$$R = \left(\frac{d}{c+d}\right) x\ 100\%$$

J. *Decision Tree*

Decision tree merupakan struktur data yang berbentuk seperti struktur pohon. Suatu sampel data yang belum diketahui kelasnya menggunakan *decision tree* sebagai metode untuk mengklasifikasikan sampel data sesuai dengan kelas-kelas yang ada. *Decision tree* memiliki ukuran *tree* yang relatif kecil sehingga relatif mudah dalam menginterpretasinya. *Decision tree* terdiri dari *internal node* (simpul percabangan), *leaf node* (simpul daun), dan *root node* (simpul akar). *Internal node* merupakan *node* yang menyatakan pengujian terhadap suatu atribut dimana setiap cabangnya menyatakan *output* dari suatu pengujian, *leaf node* merupakan *node* yang menyatakan distribusi kelas, sedangkan *root node* merupakan *node* yang letaknya berada di paling atas. *Internal node* akan memiliki satu *edge* (rusuk) masuk dan beberapa *edge* keluar, *leaf node* akan memiliki satu *edge* masuk tetapi tidak memiliki *edge* keluar, sedangkan *root node* akan memiliki beberapa *edge* yang keluar tetapi tidak memiliki *edge* masuk [16].

Decision tree memiliki beberapa karakteristik, yaitu:

1. Memberikan gambaran yang ekspresif dalam pembelajaran fungsi nilai diskrit.
2. Algoritma yang dimiliki *decision tree* cukup kuat untuk menangani masalah *overfitting*.
3. Atribut yang berlebihan pada *decision tree* tidak terlalu menghalangi kinerja dari keakurasiannya.

K. *Naive Bayes*

Naive Bayes Classifier (NBC) adalah metode pengklasifikasian data statistik untuk memprediksi probabilitas keanggotaan suatu kelas. Menurut [10], berdasarkan teori Bayes yang mengasumsikan bahwa atribut-atribut klasifikasi bersifat ketidakbergantungan atau independent dan tidak terdapat korelasi antara mereka. *Naive bayes* memiliki kelebihan dalam penggunaannya, yaitu hanya membutuhkan training data yang tidak besar untuk dapat menentukan perkiraan parameter yang digunakan dalam pengujian probabilistik. Pada saat klasifikasi, pendekatan dari *naive bayes* akan menghasilkan label kategori yang paling tinggi probabilitasnya.

L. *Induction Rule*

Induction rule merupakan metode yang bertujuan untuk mencari pola yang sering muncul diantara banyak transaksi, dimana setiap transaksinya terdiri dari beberapa *item*. Metode ini menghasilkan aturan yang menentukan besar hubungan antara X dan Y. Aturan tersebut akan digunakan untuk keperluan *support* dan *confidence* [17]. Pada metode *induction rule*, digunakan proses perhitungan algoritma *information gain* [18].

III. PEMROSESAN DATA

A. Deskripsi Data

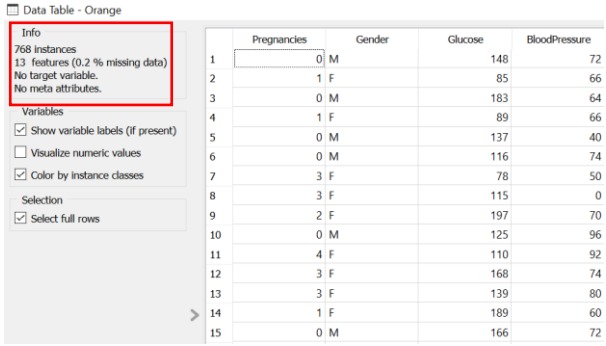
Dataset yang digunakan adalah *dataset* mengenai diagnosa penyakit diabetes yang dipublikasikan di situs *website* kaggle.com [19]. *Dataset* tersebut terdiri dari 768 data, dengan 13 atribut. Atribut-atribut tersebut seperti yang ditampilkan pada Tabel 2.

Tabel 2. Tabel Tipe Atribut

No	Atribut	Tipe
1	*Pregnancies*	Numerik
2	*Gender*	Kategorikal
3	*Glucose*	Numerik
4	*BloodPressure*	Numerik
5	*SkinThickness*	Numerik
6	*Insulin*	Numerik
7	*BMI*	Numerik
8	*DiabetesPedigreeFunction*	Numerik
9	*Age*	Numerik
10	*Outcome*	Kategorikal
11	*CalorieIntake*	Numerik
12	*Exercise*	Kategorikal
13	*SleepDuration*	Numerik

B. *Data Cleaning*

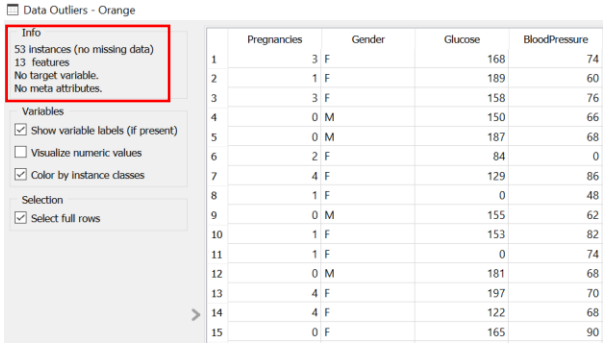
1. *Imputation*



Gambar 1. *Missing* data pada *dataset*.

Gambar 1 menunjukkan bahwa dari 768 data, terdeteksi bahwa data memiliki *missing* data sebesar 0,2%. Maka dari itu, perlu dilakukan *imputation*. *Imputation* yang dilakukan disini adalah dengan cara mengisi data menggunakan rata-rata atribut.

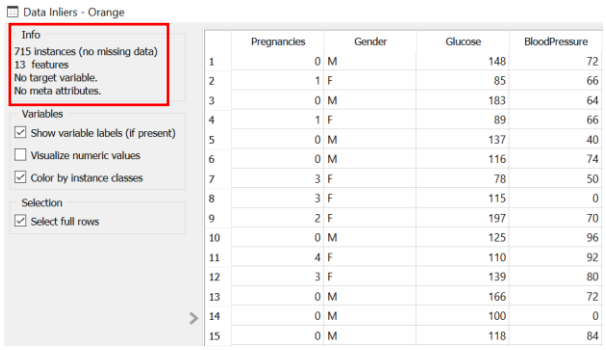
2. Deteksi *Outliers*



Gambar 2. Data *Outliers*

Gambar 2 menunjukan bahwa hasil deteksi *outliers*, didapatkan sebanyak 53 data yang termasuk kedalam data *outliers*.

3. Data *Inliers*

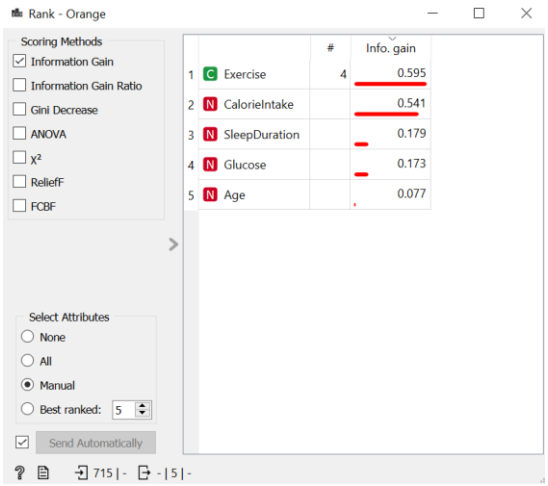


Gambar 3. Data *Inliers*

Gambar 3 menunjukkan data *inliers* dari *dataset* tersebut. Data *inliers* didapatkan dari hasil data mentah dikurangi data *outliers*. Data awal berjumlah 768 data kemudian dikurangi dengan 53 data (*outliers*) menjadi 715 data.

4. Data *Reduction*

Pada tahapan data *reduction*, digunakan fitur "*Select Relevant Feature*" dengan teknik *Information Gain*. *Feature* yang dipilih saat melakukan "*Select Relevant Feature*" adalah berjumlah lima *feature*. Kelima *feature* ini merupakan atribut yang paling berpengaruh untuk menentukan atribut dependen atau atribut *class*. Untuk melihat atribut apa saja yang paling berpengaruh terhadap atribut *class*, dapat menggunakan fitur "*Rank*" di aplikasi *Orange*.



Gambar 4. Tampilan urutan atribut berdasarkan nilai *information gain*.

Pada gambar 4 dapat dilihat bahwa atribut yang paling berpengaruh terhadap atribut *class*, yaitu atribut *Exercise* dengan nilai *information gain* sebesar 0.595, atribut *CalorieIntake* dengan nilai *information gain* sebesar 0.541, atribut *SleepDuration* dengan nilai *information gain* sebesar 0.179, atribut *Glucose* dengan nilai *information gain* sebesar 0.173, dan terakhir atribut *Age* dengan nilai *information gain* sebesar 0.077.

5. Data Akhir *Preprocessing*

Setelah melewati tahap pemrosesan data, data mengalami pengurangan jumlah, dari 768 data menjadi 715 data. Atribut juga mengalami pengurangan jumlah, dari 13 atribut menjadi 6 atribut (dimana 1 atribut merupakan atribut *class* atau dependen dan 5 atribut merupakan atribut independen). Pengurangan jumlah data terjadi karena adanya *outliers*, sedangkan pengurangan jumlah atribut terjadi karena melalui tahapan data *reduction* yang menggunakan "*Select Relevant Feature*" dengan teknik *information gain*. Selain itu, urutan atribut juga berubah menjadi didasarkan pada nilai *information gain* yang paling besar.

IV. PEMODELAN DATA

A. Deskripsi data

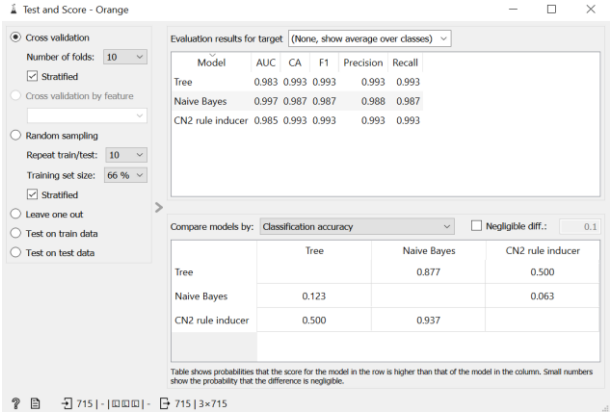
Dataset yang digunakan adalah *dataset* mengenai diagnosa penyakit diabetes yang telah melalui tahapan *preprocessing*. *Dataset* asli dipublikasikan di situs kaggle.com [19]. *Dataset* setelah melalui tahapan *preprocessing* memiliki 715 data, dengan 6 atribut. Atribut-atribut tersebut seperti yang ditampilkan pada tabel 3.

No	Atribut	Tipe
1	*Exercise*	Kategorikal
2	*CalorieIntake*	Numerik
3	*SleepDuration*	Numerik
4	*Glucose*	Numerik
5	*Age*	Numerik
6	*Outcome*	Kategorikal

Tabel 3. Tabel tipe atribut.

B. Hasil Pemodelan

Metode yang digunakan untuk pemodelan ini, yaitu *Decision Tree*, *Naive Bayes*, dan *Induction Rule*. Pemodelan dilakukan dengan menggunakan aplikasi *Orange*, seperti yang ditunjukan pada gambar 5.



Gambar 5. Hasil Pemodelan

Berdasarkan CA (*Classification Accuracy*) dari tiga metode tersebut, dapat disimpulkan bahwa:

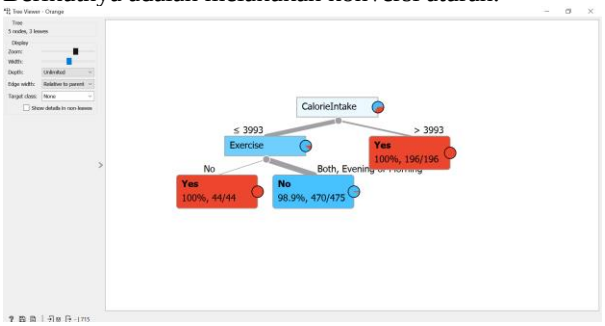
1. *Tree* lebih baik dari *Naive Bayes* karena probabilitasnya 0.877, dan *Tree* tidak lebih baik dari CN2 *Rule* karena probabilitasnya 0.500.
2. *Naive Bayes* tidak lebih baik dari *Tree* karena probabilitasnya 0.123, dan *Naive Bayes* tidak lebih baik dari CN2 *Rule* karena probabilitasnya 0.063.
3. CN2 *Rule* tidak lebih baik dari *Tree* karena probabilitasnya 0.500, dan CN2 *Rule* lebih baik dari *Naive Bayes* karena probabilitasnya 0.937.

Jadi metode yang terbaik untuk digunakan berdasarkan akurasinya adalah metode *Tree* dan metode CN2 *Rule Induction* karena probabilitasnya yang hampir mendekati 1. Oleh karena itu, hasil pemodelan yang akan dibahas lebih lanjut adalah pemodelan dengan menggunakan metode *Decision Tree* dan *Induction Rule*.

1. Pemodelan menggunakan metode *Decision Tree*

Untuk mengetahui nilai akurasi dari suatu model maka akan dilakukan evaluasi. Evaluasi dilakukan dengan menggunakan metode *Cross Validation*, didapatkan akurasi pada metode *Decision Tree* adalah sebesar 99,3%. AUC sebesar 98,3%. F1 *Score* sebesar 99,3%. *Precision* sebesar 99,3%. *Recall* sebesar 99,3%.

Berikutnya adalah melakukan konversi aturan.



Gambar 6. *Decision Tree*

Dari *Decision Tree* tersebut, dapat dikonversi menjadi 3 aturan, aturan-aturan tersebut sebagai berikut:

1. (*CalorieIntake* > 3993) → *Yes*
2. (*CalorieIntake* ≤ 3993, *Exercise* = *No*) → *Yes*
3. (*CalorieIntake* ≤ 3993, *Exercise* = *Both OR Exercise* = *Evening OR Exercise* = *Morning*) → *No*

Selanjutnya adalah dengan menggunakan *Confusion Matrix* akan dilakukan penentuan TP, FP, TN, FN, perhitungan *Precision*, *Recall*, F1 *Score*, *Accuracy*, dan AUC, serta perbandingan *Precision*, *Recall*, F1, *Accuracy*, dan AUC dari hasil pemodelan.

Gambar 7. *Confusion Matrix* pada metode *Decision Tree*

(a) Menentukan TP, FP, TN, FN.
TP: 470
FP: 5
TN: 240
FN: 0

(b) Menghitung *Precision*, *Recall*, F1, *Accuracy*, dan AUC.

Precision:

$$\frac{TP}{(TP+FP)}$$

$$\frac{470}{(470+5)} = \frac{470}{475} = 0{,}99 = 99\%$$

Recall:

$$\frac{TP}{(TP+FN)}$$

$$\frac{470}{(470+0)} = \frac{470}{470} = 1 = 100\%$$

F1 *Score*:

$$\frac{2 \times (Precision \times recall)}{(Precision + recall)}$$

$$\frac{2 \times (0{,}99 \times 1)}{(0{,}99 + 1)} = \frac{1{,}98}{1{,}99} = 0{,}994 = 99{,}4\%$$

Accuracy:

$$\frac{TP+TN}{(TP+TN+FP+FN)}$$

$$\frac{470+240}{(470+240+5+0)} = \frac{710}{715} = 0{,}993 = 99{,}3\%$$

AUC:

$$\frac{Recall+Specificity}{2}$$

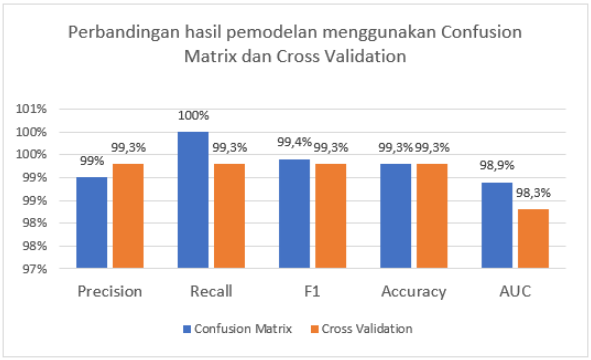
Karena nilai *specificity* belum diketahui, maka cari nilai *specificity* terlebih dahulu dengan cara:

$$\frac{TN}{(TN+FP)} = \frac{240}{(240+5)} = 0{,}979 = 97{,}9\%$$

Setelah nilai *specificity* diperoleh, kemudian hitung nilai AUC:

$$\frac{1+0{,}979}{2} = 0{,}989 = 98{,}9\%$$

(c) Membandingkan dengan *Precision*, *Recall*, F1, *Accuracy*, dan AUC hasil pemodelan.



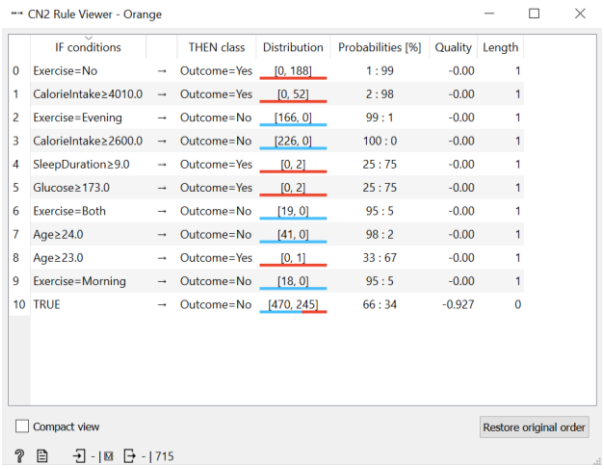
Gambar 8. Perbandingan hasil pemodelan menggunakan *Confusion Matrix* dan *Cross Validation*.

Pada gambar 8 menunjukkan bahwa perbedaan hasil nilai antara menggunakan *Confusion Matrix* dengan hasil pemodelan *Cross Validation* adalah terletak pada semua nilai kecuali nilai akurasinya, karena baik menggunakan *Confusion Matrix* maupun menggunakan *Cross Validation* akurasi yang didapatkan adalah sebesar 99,3%.

2. Pemodelan menggunakan metode *Induction Rule*

Untuk mengetahui nilai akurasi dari suatu model maka akan dilakukan evaluasi. Evaluasi dilakukan dengan menggunakan metode *Cross Validation*, didapatkan akurasi pada metode *Induction Rule* adalah sebesar 99,3%. AUC sebesar 98,5%. *F1 Score* sebesar 99,3%. *Precision* sebesar 99,3%. *Recall* sebesar 99,3%.

Berikutnya adalah melakukan konversi aturan.



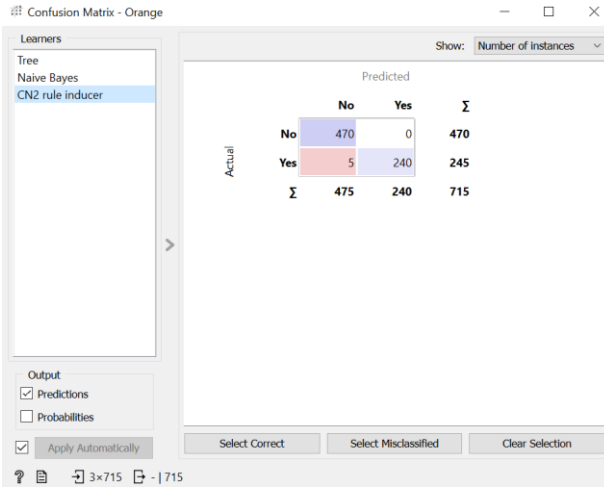
	IF conditions		THEN class	Distribution	Probabilities [%]	Quality	Length
0	Exercise=No	→	Outcome=Yes	[0, 188]	1 : 99	-0.00	1
1	CalorieIntake≥4010.0	→	Outcome=Yes	[0, 52]	2 : 98	-0.00	1
2	Exercise=Evening	→	Outcome=No	[166, 0]	99 : 1	-0.00	1
3	CalorieIntake≥2600.0	→	Outcome=No	[226, 0]	100 : 0	-0.00	1
4	SleepDuration≥9.0	→	Outcome=Yes	[0, 2]	25 : 75	-0.00	1
5	Glucose≥173.0	→	Outcome=Yes	[0, 2]	25 : 75	-0.00	1
6	Exercise=Both	→	Outcome=No	[19, 0]	95 : 5	-0.00	1
7	Age≥24.0	→	Outcome=No	[41, 0]	98 : 2	-0.00	1
8	Age≥23.0	→	Outcome=Yes	[0, 1]	33 : 67	-0.00	1
9	Exercise=Morning	→	Outcome=No	[18, 0]	95 : 5	-0.00	1
10	TRUE	→	Outcome=No	[470, 245]	66 : 34	-0.927	0

Gambar 9. Aturan dari *Induction Rule*

Dari *rule viewer* tersebut, dapat dikonversi menjadi 10 aturan, aturan-aturan tersebut sebagai berikut:

1. *IF* (*Exercise* = *No*) *THEN Outcome* = *Yes*
2. *IF* (*CalorieIntake* ≥ 4010) *THEN Outcome* = *Yes*
3. *IF* (*Exercise* = *Evening*) *THEN Outcome* = *No*
4. *IF* (*CalorieIntake* ≥ 2600) *THEN Outcome* = *No*
5. *IF* (*SleepDuration* ≥ 9) *THEN Outcome* = *Yes*
6. *IF* (*Glucose* ≥ 173) *THEN Outcome* = *Yes*
7. *IF* (*Exercise* = *Both*) *THEN Outcome* = *No*
8. *IF* (*Age* ≥ 24) *THEN Outcome* = *No*
9. *IF* (*Age* ≥ 23) *THEN Outcome* = *Yes*
10. *IF* (*Exercise* = *Morning*) *THEN Outcome* = *No*

Selanjutnya adalah dengan menggunakan *Confusion Matrix* akan dilakukan penentuan TP, FP, TN, FN, perhitungan *Precision*, *Recall*, F1 *Score*, *Accuracy*, dan AUC, serta perbandingan *Precision*, *Recall*, F1, *Accuracy*, dan AUC dari hasil pemodelan.



Gambar 10. *Confusion matrix* pada metode *Induction Rule*

(a) Menentukan TP, FP, TN, FN.
TP: 470
FP: 5
TN: 240
FN: 0

(b) Menghitung *Precision*, *Recall*, F1, *Accuracy*, dan AUC.

Precision:

$$\frac{TP}{(TP+FP)}$$

$$\frac{470}{(470+5)} = \frac{470}{475} = 0{,}99 = 99\%$$

Recall:

$$\frac{TP}{(TP+FN)}$$

$$\frac{470}{(470+0)} = \frac{470}{470} = 1 = 100\%$$

F1 *Score*:

$$\frac{2 \times (Precision \times recall)}{(Precision + recall)}$$

$$\frac{2 \times (0{,}99 \times 1)}{(0{,}99 + 1)} = \frac{1{,}98}{1{,}99} = 0{,}994 = 99{,}4\%$$

Accuracy:

$$\frac{TP+TN}{(TP+TN+FP+FN)}$$

$$\frac{470+240)}{(470+240+5+0)} = \frac{710}{715} = 0{,}993 = 99{,}3\%$$

AUC:

$$\frac{Recall+Specificity}{2}$$

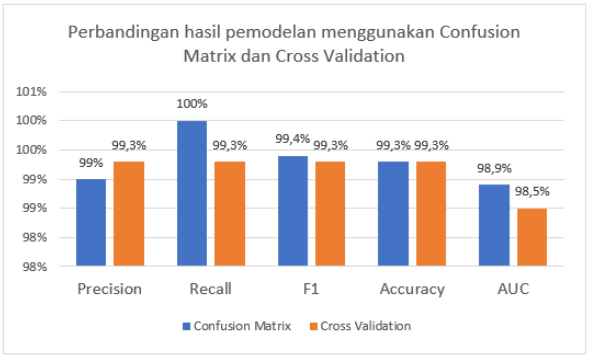
Karena nilai *specificity* belum diketahui, maka cari nilai *specificity* terlebih dahulu dengan cara:

$$\frac{TN}{(TN+FP)} = \frac{240}{(240+5)} = 0{,}979 = 97{,}9\%$$

Setelah nilai *specificity* diperoleh, kemudian hitung nilai AUC:

$$\frac{1+0{,}979}{2} = 0{,}989 = 98{,}9\%$$

(c) Membandingkan dengan *Precision*, *Recall*, F1 *Score*, *Accuracy*, dan AUC hasil pemodelan.



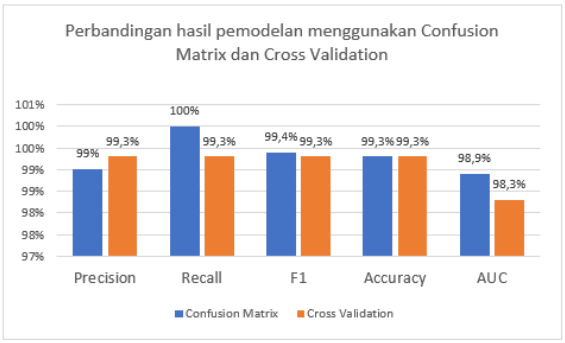
Gambar 11. Perbandingan hasil pemodelan menggunakan *Confusion Matrix* dan *Cross Validation*.

Pada gambar 11 menunjukkan bahwa perbedaan hasil nilai antara menggunakan *Confusion Matrix* dengan hasil pemodelan *Cross Validation* adalah terletak pada semua nilai kecuali nilai akurasinya, karena baik menggunakan *Confusion Matrix* maupun menggunakan *Cross Validation* akurasi yang didapatkan adalah sebesar 99,3%.

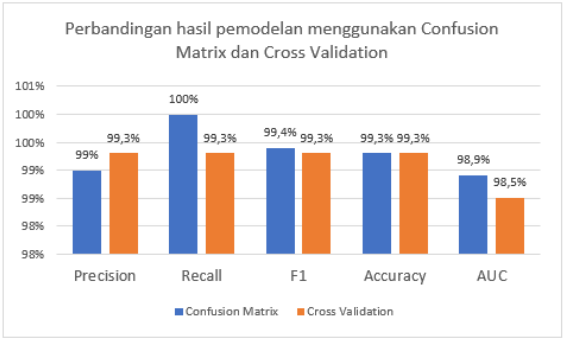
V. PEMBAHASAN

Pemrosesan data menghasilkan 715 data dari 768 data, serta 6 atribut yang terdiri dari satu atribut *class* atau dependen, dan lima atribut independen. Kelima atribut independen tersebut merupakan atribut yang paling berpengaruh dalam menentukan *class* dari sebuah data. Atribut-atribut tersebut antara lain; *Exercise*, *CalorieIntake*, *SleepDuration*, *Glucose*, dan *Age*, sementara atribut yang merupakan atribut *class* dari data tersebut adalah *Outcome*. Kemudian dari data dan atribut tersebut dilakukan pemodelan dengan menggunakan tiga metode, yakni *Decision Tree*, *Naive Bayes*, dan *Induction Rule*. Selanjutnya dari ketiga metode tersebut, dilakukan evaluasi pada masing-masing metode dengan tujuan untuk mendapatkan metode mana yang paling bagus untuk digunakan pada klasifikasi data tersebut, evaluasi yang digunakan adalah dengan teknik *Cross Validation*. Dapat dilihat pada gambar 5 menunjukan bahwa berdasarkan CA (*Classification Accuracy*) dari tiga metode tersebut, metode yang terbaik untuk digunakan berdasarkan akurasinya adalah metode *Tree* dan metode CN2 *Rule Induction* karena probabilitasnya yang hampir mendekati 1 sedangkan *Naive Bayes* memiliki probabilitas yang lebih rendah dibandingkan kedua metode lainnya, sehingga pembahasan mengenai evaluasi akan difokuskan pada pemodelan dengan menggunakan metode *Decision Tree* dan *Induction Rule*. Tujuan dari evaluasi ini adalah untuk mendapatkan nilai *Accuracy*, *Recall*, AUC, F1 *Score*, dan *Precision*.

Decision Tree



Induction Rule



Gambar 12. Perbandingan hasil pemodelan metode *Decision Tree* dan *Induction Rule*.

Berdasarkan gambar 12 dapat dilihat bahwa hasil perbandingan evaluasi menunjukan tidak ada perbedaan yang signifikan antara pemodelan menggunakan *Decision Tree* dan *Induction Rule*. Kedua pemodelan memiliki tingkat akurasi yang sama, yaitu 99,3%, dan skor evaluasi lainnya yang hampir identik. Namun jika melihat perbedaan kecil pada beberapa metrik evaluasi, AUC pada pemodelan menggunakan *Induction Rule* metoda *Cross Validation* sedikit lebih tinggi dibandingkan dengan AUC pada pemodelan menggunakan *Decision Tree* metoda *Cross Validation*. Oleh karena itu, dapat disimpulkan bahwa pemodelan menggunakan *Induction Rule* sedikit lebih baik dalam kinerja keseluruhan.

VI. KESIMPULAN

Pemodelan yang dilakukan menggunakan metode *Decision Tree*, *Naive Bayes*, dan *Induction Rule* pada *dataset* diagnosa penyakit diabetes dari situs Kaggle melalui tahap *preprocessing*, memproses 768 jumlah data dengan 13 jumlah atribut hingga menjadi 715 data dan 6 atribut. Dihasilkan bahwa pemodelan menggunakan *Induction Rule* merupakan metode yang paling baik untuk digunakan dalam teknik klasifikasi data dibandingkan dengan metode *Decision Tree* atau *Naive Bayes*.

Penggunaan metode *Induction Rule* pada pemodelan tersebut menghasilkan *Accuracy* sebesar 99,3%, AUC sebesar

98,5%, F1 *Score* sebesar 99,3%, *Precision* sebesar 99,3%, *Recall* sebesar 99,3%, dan 10 aturan. Aturan-aturan tersebut, yaitu sebagai berikut:

1. *IF* (*Exercise* = *No*) *THEN Outcome* = *Yes*
2. *IF* (*CalorieIntake* ≥ 4010) *THEN Outcome* = *Yes*
3. *IF* (*Exercise* = *Evening) THEN Outcome* = *No*
4. *IF* (*CalorieIntake* ≥ 2600) *THEN Outcome* = *No*
5. *IF* (*SleepDuration* ≥ 9) *THEN Outcome* = *Yes*
6. *IF* (*Glucose* ≥ 173) *THEN Outcome* = *Yes*
7. *IF* (*Exercise* = *Both*) *THEN Outcome* = *No*
8. *IF* (*Age* ≥ 24) *THEN Outcome* = *No*
9. *IF* (*Age* ≥ 23) *THEN Outcome* = *Yes*
10. *IF* (*Exercise* = *Morning*) *THEN Outcome* = *No*

Dari kesepuluh aturan tersebut, dapat diketahui bahwa atribut *Exercise*, *CalorieIntake*, *SleepDuration*, *Glucose*, dan *Age* mempengaruhi hasil diagnosa penyakit diabetes yang diperlihatkan dalam atribut *Outcome*. Jika *exercise = no* atau tidak adanya olahraga yang dilakukan maka hasil diagnosa diabetes positif. Sebaliknya, jika adanya olahraga yang dilakukan saat *morning*, *evening*, ataupun keduanya, hasil diagnosa adalah *no* atau negatif. Kemudian, dari aturan nomor 2 dan 4 didapat bahwa kalori yang dikonsumsi dibawah sama dengan 2600 kalori tidak mengakibatkan hasil diagnosa penyakit menjadi positif, namun asupan kalori yang dibawah sama dengan dan mendekati 4010 kalori menghasilkan diagnosa positif. Urutan dari aturan ini juga penting dalam mengambil keputusan hasil dari diagnosa penyakit. Urutan tersebut menunjukkan penyebab yang paling mempengaruhi hasil diagnosa penyakit diabetes, yaitu olahraga yang dilakukan, jumlah kalori yang dikonsumsi, durasi tidur, kadar glukosa dalam tubuh, dan umur.

DAFTAR PUSTAKA

[1] Arhami, M., Kom, M., & Muhammad Nasir, S. T. (2020). *Data Mining-Algoritma dan Implementasi*. Penerbit Andi.

[2] Han, & Kamber. (2006). Data Mining Concepts and technique. San Francisco: Diane Cerra.

[3] Sudarsono, B. G., Leo, M. I., Santoso, A., & Hendrawan, F. (2021). Analisis Data Mining Data Netflix Menggunakan Aplikasi Rapid Miner. *JBASE-Journal of Business and Audit Information Systems*, *4*(1).

[4] Nasution, D. A., Khotimah, H. H., & Chamidah, N. (2019). Perbandingan Normalisasi Data untuk Klasifikasi Wine Menggunakan Algoritma K-NN. *CESS (Journal of Computer Engineering, System and Science)*, *4*(1), 78-82.

[5] Paryudi, I., Ashari, A., & Mustofa, K. (2022). The performance of personality-based recommender system for fashion with demographic data-based personality prediction. *International Journal of Advanced Computer Science and Applications*, *13*(1).

[6] Badan Pusat Statistik. Survei Sosial Ekonomi Nasional (SUSENAS) Kor 2017. (2017). Jakarta: BPS.

[7] Jerez, J.M., dan Molina, I., (2010). Missing data imputation using statistical and machine learning methods in a real breast cancer problem. Artificial intelligence in medicine, 105-115.

[8] Hendrawati, Triyani. (2015). Kajian Metode Imputasi Dalam Menangani Missing data. Prosiding Seminar Nasional Matematika dan Pendidikan Matematika UMS. Surakarta: Universitas Muhammadiyah Surakarta.

[9] Utami, D. S., & Saputro, D. R. S. (2018). Pengelompokkan Data yang Memuat Pencilan dengan Kriteria Elbow dan Koefisien Silhouette (Algoritme K-Medoid). Konferensi Nasional Penelitian Matematika Dan Pembelajaran (KNPMP) III, 448–456.

[10] Rahmansyah, A., Dewi, O., Andini, P., Ningrum, T. H. P., & Suryana, M. E. (2018, August). Membandingkan Pengaruh Feature Selection Terhadap Algoritma Naïve Bayes dan Support Vector Machine. In *Seminar Nasional Aplikasi Teknologi Informasi (SNATi)*.

[11] Pratiwi, B. P., Handayani, A. S., & Sarjana, S. (2020). Pengukuran Kinerja Sistem Kualitas Udara Dengan Teknologi Wsn Menggunakan Confusion Matrix. *Jurnal Informatika Upgris*, *6*(2).

[12] Normawati, D., & Prayogi, S. A. (2021). Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter. *J-SAKTI (Jurnal Sains Komputer dan Informatika)*, *5*(2), 697-711.

[13] Novianti, B., Rismawan, T., & Bahri, S. (2016). Implementasi Data Mining Dengan Algoritma C4. 5 Untuk Penjurusan Siswa (Studi Kasus: Sma Negeri 1 Pontianak). *Coding Jurnal Komputer dan Aplikasi*, *4*(3).

[14] Saputra, A., & Kurniadi, D. (2019). Analisis Kepuasan Pengguna Sistem Informasi E-Campus Di Iain Bukittinggi Menggunakan Metode Eucs. *Voteteknika (Vocational Teknik Elektronika dan Informatika)*, *7*(3), 58-66.

[15] Pendit, P. L. (2008). Perpustakaan Digital dari A sampai Z. Jakarta: Citra Karya Karsa Mandiri.

[16] Qadrini, L., Seppewali, A., & Aina, A. (2021). Decision Tree dan Adaboost pada Klasifikasi Penerima Program Bantuan Sosial. Jurnal Inovasi Penelitian, 2(7), 1959-1966.

[17] Nasrum, N., Zulkarnain, Z., & Nurdiansah, N. (2022). Sistem Pelaporan Gangguan Jaringan Telkom Dengan Metode Apriori dan Generalized Rule Induction. *Dipanegara Komputer Sistem Informasi*, *16*(1), 66-70.

[18] Purwanto, A., & Darmadi, E. A. (2018). Perbandingan Minat Siswa Smu Pada Metode Klasifikasi Menggunakan 5 Algoritma. *ikraith-informatika*, *2*(1), 43-47.

[19] Kaggle Repository, "Diabetes Dataset." https://www.kaggle.com/datasets/dslearner0406/diabetes-dataset.