Analisis Algoritma *Decesion Tree, KNN* dan *Naïve Bayes* pada *Dataset* Penyakit Jantung

Ermandy Astama Putra dan Raden Supriyanto

Universitas Gunadarma
Jl. Margonda Raya No. 100, Depok 16424, Jawa Barat
E-mail: ermandyastama@gmail.com, supriyanto.r@gmail.com

Abstrak

Jantung merupakan organ vital manusia, namun terkadang memiliki beberapa masalah. Betapa pentingnya peran jantung sehingga jika mengalami masalah berpengaruh dalam kelangsungan hidup. Maka perlu adanya suatu pemanfaatan yang lebih baik dalam mendeteksi gejala penyakit jantung, salah satu teknik yang dapat digunakan yaitu *data mining* dengan teknik *classification*. Penelitian ini melakukan analisis komparasi algoritma *Decesion Tree, KNN* dan *Naïve Bayes* pada *dataset* penyakit jantung. Penggunaan *tools* membantu menghitung nilai akurasi maupun *AUC* dari tiga algoritma tersebut. Algoritma *Decesion Tree* memiliki nilai akurasi 84.77%, *KNN* memiliki nilai akurasi 94.92% dan *Naïve Bayes* memiliki nilai akurasi 81.64%. Perhitungan tersebut diuji berdasarkan nilai *recall* maupun presisi. Nilai *AUC* ditarik dari konversi ROC bahwa algoritma *Decesion Tree* memiliki nilai AUC 0.936, *KNN* memiliki nilai *AUC* 0.990 dan *Naïve Bayes* memiliki nilai *AUC* 0.901. Berdasarkan hasil tersebut disimpulkan bahwa penelitian *dataset* penyakit jantung lebih baik menggunakan algoritma *KNN* dibandingkan dengan *Decesion Tree* dan *Naïve Bayes*.

Kata kunci : *Decesion, Tree, KNN, Naïve Bayes, Penyakit Jantung, Klasifikasi*.

Pendahuluan

Jantung adalah organ tubuh manusia yang berongga dan berotot yang berperan untuk memompa darah yang membawa oksigen dan makanan ke seluruh bagian tubuh. Jantung merupakan organ vital manusia, namun terkadang jantung memiliki beberapa masalah. Dengan demikian betapa sangat pentingnya peran organ jantung tersebut sehingga jika suatu saat mengalami masalah tentunya akan sangat berpengaruh dalam kelangsungan hidup manusia. Penyakit jantung merupakan pembunuh nomor satu didunia. Setiap tahunnya lebih dari dua juta orang meninggal karena penyakit jantung. Penyakit jantung merupakan gangguan yang terjadi pada sistem pembuluh darah. Hal ini menyebabkan jantung dan peredaran darah tidak dapat bekerja sebagaimana biasanya.

Berdasarkan hal tersebut maka perlu adanya suatu pemanfaatan yang lebih baik dalam mendeteksi gejala penyakit jantung dan dapat diketahui apa saja faktor resikonya. Perkembangan teknologi informasi yang begitu pesat khususnya dalam bidang medis membuat banyak peneliti untuk mulai mengembangkan analisa dan model untuk melakukan deteksi dini dari berbagai macam penyakit salah satunya penyakit jantung. Dari permasalahan tersebut salah satu teknik yang bisa digunakan dalam melakukan pendeteksian yaitu *data mining*. Teknik *data mining* yang dapat digunakan yaitu teknik *classification*. Teknik tersebut dapat membantu untuk mengidentifikasi gejala yang timbul terhadap pasien apakah terdiagnosis gejala penyakit jantung atau tidak. Prediksi yang akurat dari penyakit jantung tersebut akan membantu dalam memberikan perawatan yang lebih baik kepada penderitanya. Pemanfaatan teknik *data mining* untuk memprediksi penyakit jantung telah dilakukan oleh peneliti terdahulu.

Penelitian yang dilakukan oleh [1] mengimplementasikan perbandingan tingkat akurasi sebelum dan sesudah reduksi dengan menggunakan algoritma *C5.0* dan algoritma *Naïve Bayes Classifier (NBC)*. Dalam hasil penelitiannya bahwa bahwasannya algoritma *Naïve Bayes Classifier (NBC)* dapat menangani proses klasifikasi pada dataset penyakit jantung dengan kinerja lebih baik dibandingkan dengan algoritma *C5.0*. Penelitian dilakukan oleh [2] mengimplementasikan beberapa algoritma klasifikasi yaitu *Decision Tree, Naïve Bayes, k-Nearest Neighbour, Random Forest* dan *Decison Stump* menggunakan uji parametrik dengan *t-test* agar dapat menghasilkan perbandingan metode yang lebih baik untuk *dataset* laki- laki penderita penyakit jantung. Hasil penelitian menun-

DOI : http://dx.doi.org/10.32409/jikstik.22.2.3353

jukkan bahwa algoritma *Random Forest* dan *Decision Stump* melakukan performa terbaik dalam melakukan klasifikasi pada *dataset* penyakit jantung dibandingkan algoritma *Decesion Tree, Naïve Bayes* dan *KNN*. Penelitian yang dilakukan oleh [3]. Pada penelitian tersebut menggunakan algoritma *KNN (k-NN)* berbasis *Forward Selection* untuk meningkatkan akurasi dalam diagnosis penyakit jantung koroner. Hasil penelitian menunjukkan bahwa algoritma *Forward Selection-kNN* memiliki akurasi yang lebih baik dari pada *k-NN*. Penelitian yang dilakukan oleh [4]. Dalam penelitian tersebut menekankan pada analisis klasifikasi diagnosa penyakit jantung menggunakan teknik *data mining*. Pada penelitian menggunakan algoritma *Multilayer Perception, Naïve Bayes, Random Forest* dan *Decission tree* untuk dilakukan komparasi menentukan presentase akurasi yang terbaik dalam memprediksi penyakit jantung. Dari hasil analisis yang dilakukan terhadap 4 algoritma menghasilkan algoritma *Multilayer Perception* memiliki akurasi sebesar 85.18%, *Naïve Bayes* memiliki akurasi sebesar 87.20%, *Random Forest* memiliki akurasi sebesar 83.72% dan *Decission tree* menghasilkan akurasi sebesar 84.26%. Dari hasil penelitian yang dilakukan bahwa algoritma *Naïve Bayes* memiliki akurasi tertinggi dari pada algoritma lainnya sehingga merupakan algoritma yang terbaik dalam memprediksi penyakit jantung. Penelitian dilakukan oleh [5]. Dalam penelitian tersebut menekankan pada analisis prediksi penyakit jantung menggunakan tiga metode *data mining* yaitu *Random Forest, Decission Tree* dan *Naïve Bayes*. Tujuan dari penelitian tersebut untuk membandingkan ketiga metode tersebut, metode mana yang menghasilkan prediksi dengan akurasi terbaik. *Dataset* yang digunakan yaitu *dataset* data *public UCI Repository* yang terdiri dari 270 *dataset* dengan 13 atribut. Kemudian peneliti membagi *dataset* menjadi dua bagian yaitu 80% sebagai *data training* dan 20% *data testing*. Hasil yang diperoleh menunjukkan bahwa algoritma *Random Forest* lebih baik dengan presisi 81% dalam prediksi penyakit jantung dibandingkan dengan algoritma *Naïve Bayes* dan *Decission Tree*.

Selanjutnya Penelitin dilakukan oleh [6]. Penelitian tersebut melakukan komparasi pada teknik *data mining* dengan menggunakan tiga algoritma yaitu algortima *Support Vector Machine, k-Nearest Neighboor* dan *Naïve Bayes*. *Dataset* yang digunakan bersumber dari *database Clevland UCI Machine Learning* yang terdiri dari 302 *dataset* dan 14 Atribut. Data uji sebesar 30% sedangkan *data training* sebesar 70%. Hasil akurasi yang diperoleh dengan algoritma *Support Vector Machine* menghasilkan akurasi sebesar 77.7%, *k- Nearest Neighboor* sebesar 76.67% dan *Naïve Bayes* sebesar 86.6%. Dari penelitian tersebut dihasilkan bahwa algoritma *Naïve Bayes* menghasilkan akurasi terbaik dari algoritma lainnya yang diuji. Penelitian dilakukan oleh [7]. Dalam penelitian tersebut *dataset* yang digunakan berasal dari *dataset public Clevland* yang berjumlah 303 data dan memiliki 13 atribut. Algoritma yang digunakan untuk melakukan komparasi yaitu *RIPPER, Decission Tree, Artificial Neural Network* dan *Support Vector Machine*. Hasil penelitian diperoleh bahwa algoritma *Support Vector Machine* memiliki akurasi tertinggi dengan presentase sebesar 84.12%. Penelitian dilakukan oleh [8]. algoritma yang digunakan yaitu *Repeated Incremental Pruning to Produce Error Reduction (RIPPER), Decision Tree, Artificial Neural Network, Naïve Bayes, Support Vector Machine* dan *k-Nearest Neighbor*. Penelitian tersebut melakukan komparasi berdasarkan nilai *Sensitivity, Specificity, Accuracy, Error Rate, True Positive Rate* dan *False Positive Rate*. Berdasarkan hasil pengujian didapatkan bahwa algoritma *Support Vector Machine* merupakan algoritma yang terbaik dalam memprediksi penyakit jantung dibandingkan dengan algoritma *Repeated Incremental Pruning to Produce Error Reduction (RIPPER), Decision Tree, Artificial Neural Network, Naïve Bayes* dan *k-Nearest Neighbor*. Penelitian dilakukan oleh [9]. Dalam penelitian tersebut dataset menggunakan data yang diambil dari Universitas di California, Irvine (UCI) *Machine Learning Repositori. Dataset* yang digunakan memiliki 14 atribut yang digunakan untuk mendiagnosa penyakit jantung yaitu *age, sex, cp, trestbps, col, fbs, restecg, thalach, exang, oldpeak, slope, ca, thal* dan target. Penelitian ini membandingkan tiga algoritma yaitu *Naive Bayes, Logistic Regression* dan *Support Vector Machine (SVM)* yang bertujuan untuk mengetahui tingkat akurasi terbaik dari dataset yang digunakan untuk memprediksi penyakit jantung. Penelitian ini menghasilkan akurasi terbaik sebesar 87% yang dihasilkan dengan metode *Naive Bayes*. Penelitian dilakukan oleh [10]. Dalam penelitian tersebut *dataset* menggunakan data yang diambil dari *repository UCI* pada link *https://archive.ics.uci.edu/ml/datasets/Heart +Disease. Dataset* yang digunakan memiliki 668 record dengan 14 atribut yaitu *age, sex, cp, trestbps, chol, restecg, fbs, thalach, exang, oldpeak, slope, ca, thal* dan target. Penelitian ini menggunakan tiga algoritma yaitu *Decision Tree, Naive Bayes* dan *Neural Networ*k dengan membandingkan tingkat akurasi terbaik dari tiga algortima tersebut. Berdasarkan hasil penelitian dapat disimpulkan bahwa algoritma *Decision Tree* memiliki tingkat akurasi terbaik yaitu sebesar 98,54% dibandingkan dengan algoritma *Naïve Bayes* sebesar 85,01% dan algoritma *Neural Network* sebesar 81,83%.

Pada penelitian ini, digunakan *dataset public* yang bersumber dari *https://kaggle.com/johnsmith-88/heart-disease-dataset.* Kemudian ditambahkan data lokal yang bersumber dari rumah sakit. Sumber *dataset* terdiri atas dengan 14 atribut yaitu *Age, Sex, Chest Pain Type, Resting Blood Pressure, Serum Cholestoral, Fasting Blood Sugar, Resting*

Electrocardiographic results, Maximum Heart rate achieved, Exercise induced angina, oldpeak, slope of peak, number of major vessel, thal, dan *output. Dataset* yang diuji berjumlah 1517 data. Hasil penelitian ini diharapkan dapat membantu tenaga medis, pasien maupun masyarakat dalam mengambil langkah terhadap deteksi dini penyakit jantung yang mana salah satu penyakit dengan jumlah resiko kematian tertinggi di dunia.

Metode Penelitian

Metode penelitian ini menjelaskan tahapan yang akan digunakan dalam proses penelitian secara sistematis sehingga penelitian yang dilakukan dapat terarah dengan baik.

Identifikasi Masalah

Untuk permasalahan yang timbul yaitu pada saat mendiagnosa pasien yang beresiko penyakit jantung dengan gejala-gejala yang timbul pada *dataset* yang diteliti. Target dari *dataset* tersebut adalah untuk membedakan pasien yang sakit (terdiagnosa penyakit jantung) dengan pasien yang sehat (tidak terdiagnosa penyakit jantung).

Tujuan Penelitian

Tujuan dari penelitian ini yaitu melakukan prediksi mengenai pasien yang terdiagnosa penyakit jantung dengan menggunakan metode klasifikasi, dengan menggunakan algoritma *Decesion Tree, KNN* dan *Naïve Bayes.* Urgensi dari penelitian yang dilakukan dapat membantu pihak tenaga medis maupun pasien dalam mendiagnosa gejala yang dialami apakah terdiagnosa penyakit jantung atau tidak.

Studi Pustaka

Studi pustaka bertujuan untuk mengetahui teori yang mendukung dalam penelitian yang dikerjakan, selain itu menjadi referensi terkait permasalahan yang dialami dalam proses penelitian dan dalam melakukan proses diagnosa pasien yang beresiko penyakit jantung menggunakan *data mining* dengan metode klasifikasi. Teori pendukung ini yang kemudian dijadikan dasar maupun referensi.

Pengumpulan Data

Pada penelitian ini, digunakan *dataset public* yang bersumber dari *https://kaggle.com/johnsmith88/-heart-disease-dataset.* Pengolahan data penelitian menggunakan aplikasi *RapidMiner Studio* untuk mempermudah dalam perhitungan akurasi masing-masing algoritma sehingga dapat dilakukan perbandingan.

Analisis Kebutuhan Sistem

Dalam melakukan proses penelitian maupun pengujian terhadap algoritma dibutuhkan perangkat lunak (*software*) dan perangkat keras (*hardware*) yang berfungsi untuk memudahkan pengujian terhadap *dataset* yang diuji. *Spesifikasi hardware* diantaranya *Processor Intel Core i5-5005u, 2.0Ghz, RAM 4Gb, Harddisk 500 GB.* Sedangkan spesifikasi *software* diantaranya *Microsoft Windows 10 64bit, Microsoft Office 2010, RapidMiner Studio* versi 9.9.002.

Dataset

Data yang digunakan dalam penelitian ini bersumber dari *Kaggle.com* dengan format *file .csv* dengan ukuran 37 KB. *Dataset* ini memiliki 14 atribut dan jumlah data sebanyak 1517 data yang terdiri dari 2 hasil proses yaitu 0 / Sehat (Tidak terdiagnosa penyakit jantung) dan 1 / Sakit (Terdiagnosa penyakit jantung). Hasil target *dataset* digunakan peneliti sebagai perbandingan untuk mendapatkan akurasi terbaik dari algoritma yang diujikan. Atribut *dataset* dapat dilihat pada Tabel 1.

Tabel 1: Atribut Dataset

Atribut	**Nilai**
Age	29–77
Sex	Laki-laki, Perempuan
Chest Pain Type (CP)	*Typical angina, Atypical angina, Non-anginal pain, Asymptomatic pain*
Trestbps	94-200
Cholesterol	126-564
FBS	*True, False*
Resting Electrocardiology	Normal, Abnormal, *Hypertrofi Ventrikel*
Max Heart Rate	71-202
Exercise Induced Angina	*True, False*
Oldpeak	0-4
Slope	*Down, Flat, Upsloping*
Number of vessels Colored	0-4
Thal	Normal, Cacat Tetap, Cacat Reversibel
Target	*No* (Sehat), *Yes* (Sakit)

Dari Tabel 1 dapat dijelaskan bahwa *Age* menunjukkan pasien pada *dataset* berusia 29 hingga 77 tahun. *Sex* : menunjukkan pasien pada *dataset* berjenis kelamin Laki-laki dan Perempuan. *Chest Pain Type (CP)* : merupakan tipe nyeri dada yang diderita pasien. Atribut ini memiliki 4 nilai yaitu *Typical angina* diberi nilai Tipe 1, *Atypical angina* diberi nilai Tipe 2, *Non- anginal pain* diberi nilai Tipe 3 dan *Asymptomatic* pain diberi nilai 4. *Trestbps* yaitu tekanan darah pasien ketika dalam keadaan istirahat. Satuan yang dipakai adalah *mm Hg. Cholesterol* yaitu kadar kolesterol dalam darah pasien, dengan satuan *mg/dl. Fbs* yaitu kadar gula darah pasien, atribut *fbs* ini hanya memiliki 2 nilai yaitu 1 jika kadar gula darah pasien lebih dari 120 mg/dl, dan 0 jika kadar gula darah pasien kurang dari sama dengan 120 mg/dl. *Resting Elec-*

trocardiographic yaitu kondisi *ECG* pasien ketika dalam keadaan istirahat. Atribut ini memiliki 3 nilai yaitu nilai 0 untuk keadaan normal, nilai 1 untuk keadaan ST-T *wave abnormality* (abnormal) yaitu keadaan dimana gelombang *inversions* T dan atau ST meningkat maupun menurun lebih dari 0,5 mV, dan nilai 2 untuk keadaan dimana *Hypertrofy Ventrikel. Max Heart Rate* yaitu denyut jantung target yang umumnya dinyatakan sebagai persentase (71-202) persen) dari detak jantung aman maksimum seseorang. *Exercise Induced Angina* adalah keluhan umum pasien jantung, terutama saat berolahraga dalam cuaca dingin dengan nilai *True* atau *False. Oldpeak* yaitu penurunan ST akibat olah raga. *Slope* yaitu *slope* dari puncak ST setelah berolah raga. Atribut ini memiliki 3 nilai yaitu 1 untuk *upsloping*, 2 untuk flat, dan 3 untuk *down. Number of vessels colored* yaitu jumlah pembuluh darah besar dengan penyempitan >50% (0,1,2,3, atau 4). *Thal* yaitu detak jantung pasien. Atribut ini memiliki 3 nilai yaitu 3 untuk normal, 6 untuk *fixed defect*, dan 7 untuk r*eversal defect.* Target yaitu hasil diagnosa penyakit jantung dengan 0 (*No*) dan 1 (*Yes*).

Preprocessing

Tahap *preprocessing* data merupakan tahapan awal setelah *dataset* ditentukan. Pada tahap ini dataset akan dilakukan pengecekan dan pembersihan pada *dataset* sehingga fitur yang dilakukan uji coba hanya yang relevan untuk penelitian. Tahapan *preprocessing* yaitu *Data Cleaning* dan *Diskretisasi Data. Data Cleanin*g digunakan untuk membersihkan data yang tidak relevan, menghapus data kosong, data duplikat, data *missing value* dan *data noise.* Sedangkan Diskterisasi Data adalah proses pengkategorian atau pengelompokkan nilai berdasarkan masing-masing atribut.

Split Data

Untuk menguji tingkat akurasi dari hasil klasifikasi dalam penelitian ini menggunakan metode persentase *split*, dimana *dataset* dibagi menjadi dua bagian yaitu 75% (1138 data) sebagai *data training* dan 25% (379 data) sebagai *data testing.*

Pelatihan dan Pengujian Algoritma Klasifikasi

Pada pengujian ini menggunakan tiga algoritma yaitu *Decesion Tree, KNN d*an *Naïve Bayes* untuk diketahui hasil akurasinya, kemudian dibandingkan dari ketiga algoritma tersebut yang memiliki akurasi terbaik dalam pengujian *dataset* untuk memprediksi penyakit jantung. Hasil pelatihan nantinya akan di uji dengan *confusion matrix* dan analisa kurva *ROC* untuk mengetahui tingkat akurasi dan *error* yang terdapat dalam masing-masing pengujian algoritma.

Decesion Tree

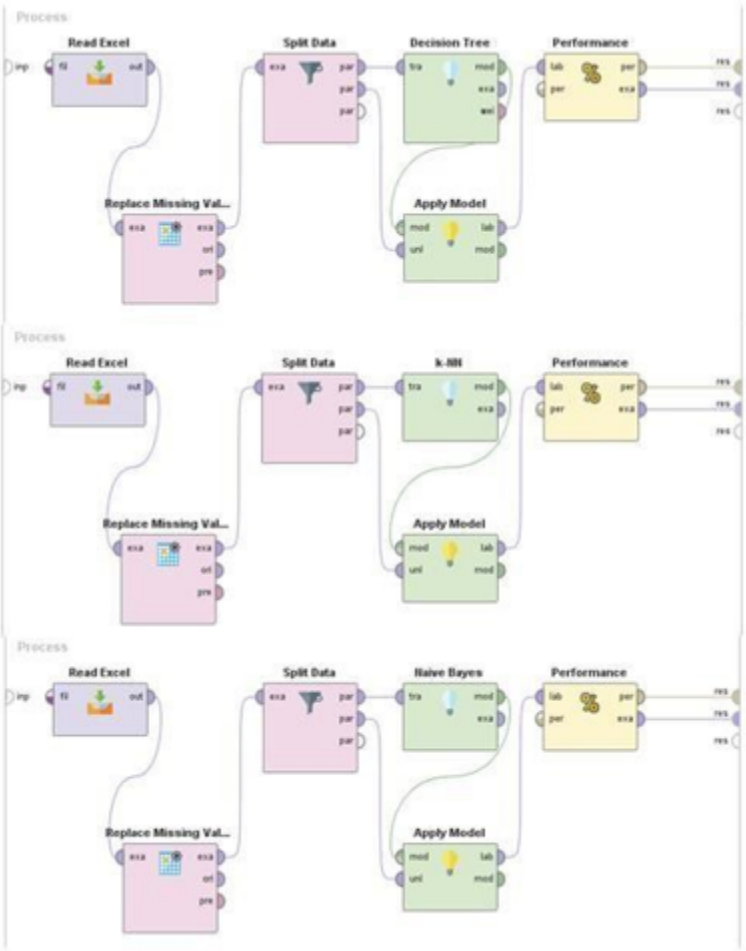
Decission Tree akan memprediksi kategori dari *class* yang diuji untuk kemudian dikelompokkan kedalam kategori atau kelas yang tersedia. Metode yang berbentuk stuktur pohon ini melakukan klasifikasi dimana tiap *node* yang disebut sebagai *leaf node* yang menunjukkan nilai atribut target atau contoh kelas dan *node* keputusan yang menentukan beberapa pengujian yang akan dilakukan. Salah satu algoritma dalam pembentukan Decission tree yaitu Algoritma *C4.5* dimana merupakan perkembangan dari algoritma *ID3.* Perhitungan dengan menggunakan Algoritma *C4.5* juga merupakan pengembangan dari algoritma *Decission tree* yaitu dengan menghitung *Entropi, Info Gain, Split* dan *Gain Ratio.* Entropi merupakan parameter untuk mengukur tingkat keberagaman dari kumpulan data, sedangkan nilai *Info Gain* untuk menentukan variabel yang dijadikan akar dari pohon keputusan yang dibuat, *Split* sebagai pembagi dari *Gain* yang akan menghasilkan *Gain Ratio* dan *Gain Ratio* merupakan ukuran untuk mengatasi masalah pada atribut yang memiliki nilai bervariasi.

KNN

K-Nearest Neighbor adalah salah satu algoritma yang digunakan dalam masalah pengklasifikasian. Algoritma *k-Nearest Neighbor* termasuk dalam algoritma *supervised learning* dimana hasil dari *instance* yang baru diklasifikasikan berdasarkan mayoritas dari kategori k-tetangga terdekat. Tujuan dari algoritma ini adalah untuk mengklasifikasikan obyek baru berdasarkan atribut dan sampel dari *data training.* Algoritma *k-Nearest Neighbor* menggunakan *Neighborhood Classification* sebagai nilai prediksi dari nilai *instance* yang baru. Prinsip kerja *k-Nearest Neighbor* yaitu mencari jarak terdekat antar data yang akan dievaluasi dengan tetangga terdekat dalam data training. Algoritma *k-Nearest Neighbor* merupakan salah satu algoritma paling sederhana untuk memecahkan masalah klasifikasi dan sering menghasilkan hasil yang kompetitif dan signifikan.

Naive Bayes

Model ini biasanya disebut dengan teknik *Bayes* yang merupakan sebuah teknik prediksi berbasis probabilistik sederhana yang berdasarkan pada penerapan teorema *Bayes* dengan asumsi yang *independent* (tidak ketergantungan) yang kuat. Hipotesis dalam teorema *Bayes* yaitu label kelas yang menjadi pemetaan dalam klasifikasi, sedangkan bukti menjadi fitur pendukung. Klasifikasi *Naïve Bayes* bekerja berdasarkan teori probabilitas yang memandang semua fitur dari berbagai data sebagai bukti dalam probabilitas.



Gambar 1: Pelatihan dan Pengujian Algoritma

Berikut penjelasan pada Gambar 1 :

1. *Read Excel*, berfungsi untuk memasukan *dataset* yang akan diuji dalam format *Excel*.

2. *Replace Missing Value*, merupakan tahap *preprocessing* yang berfungsi untuk menghilangkan *missing value, noise, data outlier* dan sebagainya.

3. *Split Data*, merupakan metode pengujian klasifikasi menggunakan metode *Split Percentage* yang berfungsi untuk membagi *dataset* menjadi dua bagian yaitu dengan perbandingan 75% *data training* dan 25% *data testing*.

4. *Decision Tree, k-NN* dan *Naïve Bayes*, pengujian algoritma yang dilakukan

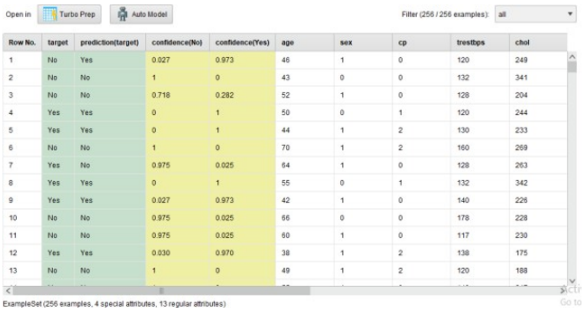
5. *Apply Model*, berfungsi untuk dapat membaca data yang akan diklasifikasikan

6. *Performance*, berfungsi untuk melihat nilai akurasi dari algoritma yang diujikan

Hasil Pengujian dan Analisa Perbandingan Algoritma Klasifikasi

Decesion Tree

Pengujian *dataset* dengan algoritma klasifikasi pertama menggunakan algoritma *Decesion Tree*. *Dataset* yang digunakan untuk pengujian berjumlah 1517 data. Setelah dilakukan pengujian dengan *split percentage* dengan perbandingan 75% (1138 data) sebagai *data training* dan 25% (379 data) sebagai *data testing* didapatkan hasilnya seperti pada Gambar 2.



Gambar 2: Hasil Prediksi Algoritma *Decesion Tree*

Pada Gambar 2 berdasarkan hasil pengujian yang dilakukan dengan *split percentage* dengan persentase 25% *data testing* maka diperoleh hasil prediksi dari *data testing* sebanyak 256 data yang dilakukan secara acak. Target merupakan hasil nilai yang berasal dari dataset yang diujikan dan *Prediction* (Target) merupakan hasil nilai prediksi menggunakan algoritma *Decesion Tree.* Hasil nilai *Prediction*(Target) diperoleh berdasarkan perbandingan antara nilai *Confidence(No)* dan *Confidence(Yes)* mana yang paling besar. Kemudian untuk hasil nilai akurasi dapat dilihat pada Gambar 3.

accuracy: 84.77%

	true No	true Yes	class precision
pred. No	113	27	80.71%
pred. Yes	12	104	89.66%
class recall	90.40%	79.39%	

Gambar 3: Hasil Pengujian Algoritma *Decesion Tree*

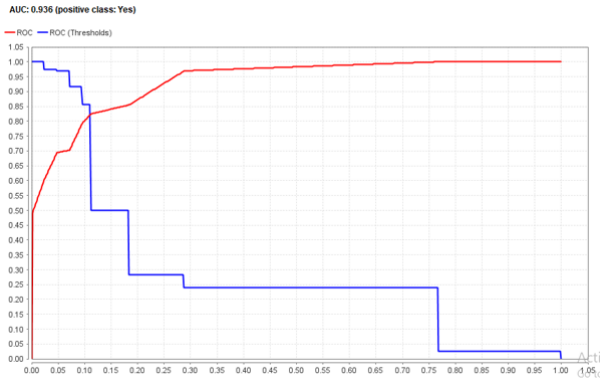
Pada Gambar 3 berdasarkan hasil pengujian diperoleh bahwa akurasi yang diperoleh dengan menggunakan model *Decesion Tree* yaitu sebesar 84,77% dengan nilai *error* sebesar 15,23%. Untuk mengetahui hasil prediksi *error* dapat menggunakan *confussion matrix* yang dapat dilihat darihasil *Performance Vector.*

PerformanceVector

```
PerformanceVector:
accuracy: 84.77%
ConfusionMatrix:
True:   No      Yes
No:     113     27
Yes:    12      104
precision: 89.66% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     113     27
Yes:    12      104
recall: 79.39% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     113     27
Yes:    12      104
AUC (optimistic): 0.955 (positive class: Yes)
AUC: 0.936 (positive class: Yes)
AUC (pessimistic): 0.918 (positive class: Yes)
```

Gambar 4: *Performance Vector Algoritma Decesion Tree*

Pada Gambar 4 berdasarkan hasil dari *Performance Vector* dapat diketahui nilai *precision* sebesar 89,66%, nilai *recall* 79,39%.

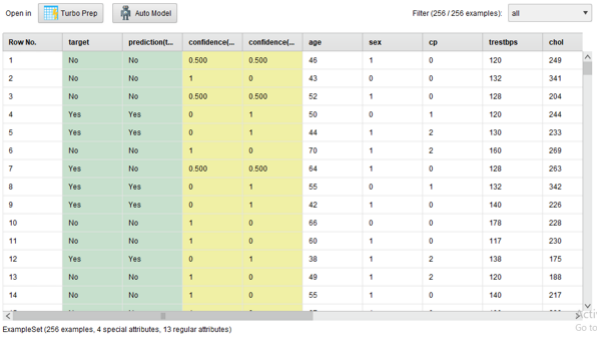


Gambar 5: Kurva *ROC* Algoritma *Decesion Tree*

Pada Gambar 5 berdasarkan tampilan hasil *AUC* dengan menggunakan kurva *ROC* bahwa nilai *AUC* sebesar 0.936 atau 93,6%.

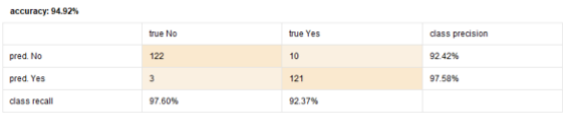
KNN

Pengujian *dataset* dengan algoritma klasifikasi kedua menggunakan algoritma *KNN. Dataset* yang digunakan untuk pengujian berjumlah 1517 data. Setelah dilakukan pengujian dengan *split percentage* dengan perbandingan 75% *data training* dan 25% *data testing* didapatkan hasilnya seperti pada Gambar 6.



Gambar 6: Hasil Prediksi Algoritma *KNN*

Pada Gambar 6 berdasarkan hasil pengujian yang dilakukan dengan *split percentage* dengan persentase 25% data testing maka diperoleh hasil prediksi dari *data testing* sebanyak 256 data yang dilakukan secara acak. Target merupakan hasil nilai yang berasal dari *dataset* yang diujikan dan *Prediction* (Target) merupakan hasil nilai prediksi menggunakan algoritma *KNN.* Hasil nilai *Prediction*(Target) diperoleh berdasarkan perbandingan antara nilai *Confidence(No)* dan *Confidence(Yes)* mana yang paling besar. Kemudian untuk hasil nilai akurasi dapat dilihat pada Gambar 7.



accuracy: 94.92%

	true No	true Yes	class precision
pred. No	122	10	92.42%
pred. Yes	3	121	97.58%
class recall	97.60%	92.37%	

Gambar 7: Hasil Pengujian Algoritma *KNN*

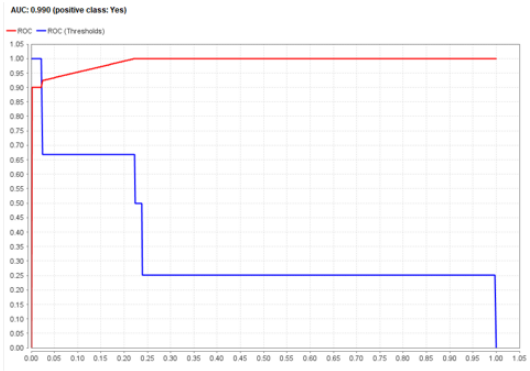
Berdasarkan pada Gambar 7 hasil pengujian diperoleh bahwa akurasi yang diperoleh dengan menggunakan model *KNN* yaitu sebesar 94,92% dengan nilai *error* sebesar 5,08%. Untuk mengetahui hasil prediksi *error* dapat menggunakan *confussion matrix* yang dapat dilihat dari hasil *Performance Vector*.

PerformanceVector

```
PerformanceVector:
accuracy: 94.92%
ConfusionMatrix:
True:   No      Yes
No:     122     10
Yes:    3       121
precision: 97.58% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     122     10
Yes:    3       121
recall: 92.37% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     122     10
Yes:    3       121
AUC (optimistic): 0.998 (positive class: Yes)
AUC: 0.990 (positive class: Yes)
AUC (pessimistic): 0.982 (positive class: Yes)
```

Gambar 8: *Performace Vector Algoritma KNN*

Berdasarkan pada Gambar 8 hasil dari *Performance Vector* dapat diketahui nilai *precision* sebesar 97,58%, nilai *recall* sebesar 92,37%.

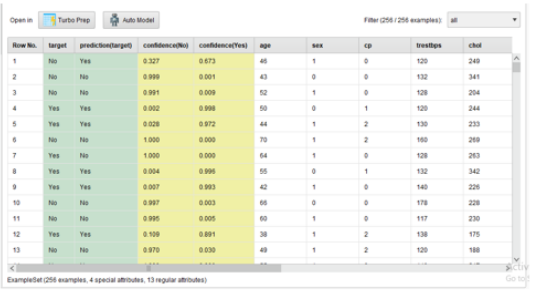


Gambar 9: Kurva *ROC* Algoritma *KNN*

Pada Gambar 9 merupakan tampilan hasil *AUC* dengan menggunakan kurva *ROC* bahwa nilai *AUC* sebesar 0.990 atau 99%.

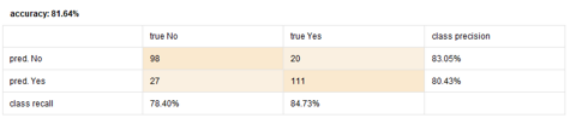
Naïve Bayes

Pengujian *dataset* dengan algoritma klasifikasi ketiga menggunakan algoritma *Naïve Bayes*. *Dataset* yang digunakan untuk pengujian berjumlah 1517 data. Setelah dilakukan pengujian dengan *split percentage* dengan perbandingan 75% *data training* dan 25% *data testing* didapatkan hasilnya seperti pada Gambar 10.



Gambar 10: Hasil Prediksi Algoritma *Naïve Bayes*

Berdasarkan pada Gambar 10 hasil pengujian yang dilakukan dengan *split percentage* dengan persentase 25% *data testing* maka diperoleh hasil prediksi dari *data testing* sebanyak 256 data yang dilakukan secara acak. Target merupakan hasil nilai yang berasal dari *dataset* yang diujikan dan *Prediction* (Target) merupakan hasil nilai prediksi menggunakan algoritma *Naïve Bayes*. Hasil nilai *Prediction*(Target) diperoleh berdasarkan perbandingan antara nilai *Confidence(No)* dan *Confidence(Yes)* mana yang paling besar. Kemudian untuk hasil nilai akurasi dapat dilihat pada Gambar 11.



accuracy: 81.64%

	true No	true Yes	class precision
pred. No	98	20	83.05%
pred. Yes	27	111	80.43%
class recall	78.40%	84.73%	

Gambar 11: Hasil Pengujian Algoritma *Naïve Bayes*

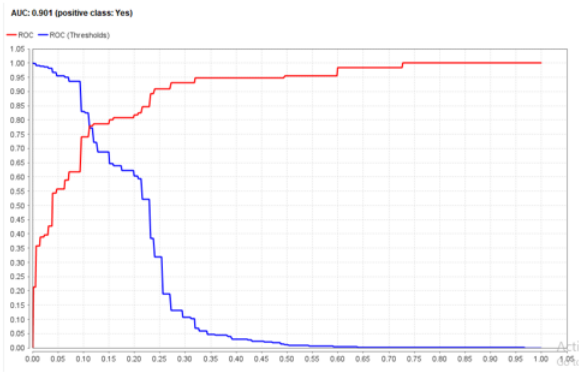
Berdasarkan pada Gambar 11 hasil pengujian diperoleh bahwa akurasi yang diperoleh dengan menggunakan model *Naïve Bayes Classifier* yaitu sebesar 81,64% dengan nilai *error* sebesar 18,36%. Untuk mengetahui hasil prediksi dapat menggunakan *confussion matrix* yang dapat dilihat dari hasil *Performace Vector*.

PerformanceVector

```
PerformanceVector:
accuracy: 81.64%
ConfusionMatrix:
True:   No      Yes
No:     98      20
Yes:    27      111
precision: 80.43% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     98      20
Yes:    27      111
recall: 84.73% (positive class: Yes)
ConfusionMatrix:
True:   No      Yes
No:     98      20
Yes:    27      111
AUC (optimistic): 0.901 (positive class: Yes)
AUC: 0.901 (positive class: Yes)
AUC (pessimistic): 0.901 (positive class: Yes)
```

Gambar 12: *Performace Vector* Algoritma *Naïve Bayes*

Berdasarkan hasil dari *Performance Vector* dapat diketahui nilai *precision* sebesar 80,43%, nilai *recall* sebesar 84,73%.



Gambar 13: Kurva *ROC* Algoritma *Naive Bayes*

Pada Gambar 13 merupakan tampilan hasil *AUC* dengan menggunakan kurva *ROC* bahwa nilai *AUC* sebesar 0.901 atau 90,1%

Hasil dan Pembahasan

Hasil dari setiap model algoritma yang dibuat dapat memunculkan hasil prediksi, nilai probabilitas, nilai akurasi, nilai *recall*, nilai presisi serta kurva *ROC (Receiver Operating Characteristic)* pada kedua algoritma tersebut.

Decesion Tree

Pada tahap ini akan dilakukan sebuah perhitungan dengan menggunakan Algoritma *Decission tree* dengan mengambil *dataset* yang diujikan menggunakan *tools RapidMiner Studio*. *Decission tree* merupakan teknik pengklasifikasian sederhana dalam *data mining* dengan pemodelan pohon keputusan. Langkah-langkah pengerjaan serta hasil perhitungan dari algoritma *Decesion Tree* dengan *Rapidminer Studio* menggunakan bantuan *Microsoft Excel*.

Setelah dilakukan perhitungan dengan *Entropy, Info Gain, Split* dan *Gain Ratio* dari masing-masing atribut, maka langkah selanjutnya menentukan atribut yang dijadikan akar dalam pohon keputusan. Pada perhitungan atribut *Info Gain* dengan nilai tertinggi terdapat pada atribut *thal*, sehingga thal dijadikan sebagai akar dalam pohon keputusan, hal ini sesuai dengan hasil pemodelan *Decission tree* menggunakan *tools* seperti Gambar 14.

Gambar 14: Hasil Pemodelan Algoritma *Decesion Tree*

KNN

Pada tahap ini akan dilakukan sebuah perhitungan dengan menggunakan model Algoritma *KNN* dengan mengambil *dataset* yang diujikan dengan menggunakan *tools RapidMiner Studio*. *KNN* merupakan algoritma yang berfungsi untuk melakukan klasifikasi suatu data berdasarkan data pembelajaran (*training dataset*) yang diambil dari k tetangga terdekatnya (*nearest neighbour*). Dengan k merupakan banyaknya tetangga terdekat.

Konsep dasar dari *KNN* yaitu mencari jarak terdekat antara data yang akan dievaluasi dengan k tetangga terdekatnya dalam data pembelajaran. Perhitungan jarak dilakukan dengan konsep *Euclidean*. Jarak *Euclidean (Euclidean Distance)* adalah formula untuk mencari jarak antara dua (2) titik dalam ruang dua dimensi. Jumlah kelas yang paling banyak dengan jarak terdekat tersebut akan menjadi kelas dimana data evaluasi tersebut berada.

Penggunaan algoritma *KNN* perlu ditentukan banyaknya k tetangga terdekat yang digunakan untuk melakukan klasifikasi data baru. Banyaknya k sebaiknya merupakan angka ganjil, misalnya k = 1, 3, 5 dan seterusnya. Penentuan nilai k dipertimbangkan berdasarkan banyaknya data yang ada dan ukuran dimensi yang dibentuk oleh data. Semakin banyak data yang ada, angka yang dipilih sebaiknya semakin rendah. Sebaliknya semakin besar ukuran dimensi data, angka k yang dipilih sebaiknya semakin tinggi. Berikut langkah-langkah pengerjaan serta hasil penghitungan dari Algoritma *KNN* dengan *tools RapidMiner Studio*:

1. Menentukan parameter k (jumlah tetangga paling dekat) k = 3.
2. Menghitung kuadrat jarak *Euclidean (Euclidean Distance)* objek terhadap data mining yang diberikan.

$$rumus : d_1 = \sqrt{\sum_{i=1}^{p} (x_1 - x_2)^2} \qquad (1)$$

   Keterangan :

   d = jarak

   i = variabeldatap = dimensi data

   x_1= sampel data

   x_2= data uji

   Dengan perhitungan menggunakan *Microsoft Excel* Seperti pada Gambar 15.
3. Menghitung hasil langkah no 2 secara *ascending* (berurutan dari nilai kecil ke besar), lihat Gambar 16.
4. Mengumpulkan kategori Y (klasifikasi *nearest neighbour* berdasarkan nilai k), lihat Gambar 17.
5. Kategori *nearest neighbor* yang paling mayoritas maka dapat diprediksikan kategori objek, lihat Gambar 18.
6. Berdasarkan data diatas, maka perbandingannya adalah 1 (Sehat) < 4 (Sakit). Maka dapat disimpulkan bahwa inputan *data testing* tersebut masuk dalam kelas Sakit (terdiagnosis penyakit jantung).

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	euclidean
2	46	1	0	120	249	0	0	144	0	0.8	2	0	3	No	19.21171518
3	43	0	0	132	341	1	0	136	1	3	1	0	3	No	101.4278561
4	52	1	0	128	204	1	0	156	1	1	1	0	0	No	41.40060386
5	50	0	1	120	244	0	1	162	0	1.1	2	0	2	Yes	0
6	44	1	2	130	233	0	1	179	1	0.4	2	0	2	Yes	23.44120304
7	70	1	2	160	269	0	1	112	1	2.9	1	1	3	No	71.65361121
8	64	1	0	128	263	0	1	105	1	0.2	1	1	3	Yes	62.26403456
9	55	0	1	132	342	0	1	166	0	1.2	2	0	2	Yes	98.93942591
10	42	1	0	140	226	0	1	178	0	0	2	0	2	Yes	32.36062422
11	66	0	0	178	228	1	1	165	1	1	1	2	3	No	62.40200317
12	60	1	0	117	230	1	1	160	1	1.4	2	2	3	No	17.8350778
13	38	1	2	138	175	0	1	173	0	0	2	4	2	Yes	73.27489338
14	49	1	2	120	188	0	1	139	0	2	1	3	3	No	60.6614375
15	55	1	0	140	217	0	1	111	1	5.6	0	0	3	No	61.50812954
16	67	1	0	100	299	0	0	125	1	0.9	1	2	2	No	71.35853138
17	29	1	1	130	204	0	0	202	0	0	2	0	2	Yes	61.18995016
18	59	1	3	170	288	0	0	159	0	0.2	1	0	3	No	67.34099791
19	47	1	2	138	257	0	0	156	0	0	2	0	2	Yes	23.28540315
20	52	1	1	134	201	0	1	158	0	0.8	2	1	2	Yes	45.46526146
21	46	1	2	150	231	0	1	147	0	3.6	1	0	2	No	36.32148125
22	38	1	2	138	175	0	1	173	0	0	2	4	2	Yes	73.27489338
23	62	0	0	124	209	0	1	163	0	0	2	0	2	Yes	37.25869026
24	65	1	0	110	248	0	0	158	0	0.6	2	2	1	No	19.11151485
25	52	1	3	118	186	0	0	190	0	0	1	0	1	Yes	64.53843816

Gambar 15: Menghitung Kuadrat Jarak *Euclidean*

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	euclidean	rank
2	46	1	0	120	249	0	0	144	0	0.8	2	0	3	No	19.21171518	18
3	43	0	0	132	341	1	0	136	1	3	1	0	3	No	101.4278561	251
4	52	1	0	128	204	1	0	156	1	1	1	0	0	No	41.40060386	113
5	50	0	1	120	244	0	1	162	0	1.1	2	0	2	Yes	0	1
6	44	1	2	130	233	0	1	179	1	0.4	2	0	2	Yes	23.44120304	40
7	70	1	2	160	269	0	1	112	1	2.9	1	1	3	No	71.65361121	210
8	64	1	0	128	263	0	1	105	1	0.2	1	1	3	Yes	62.26403456	175
9	55	0	1	132	342	0	1	166	0	1.2	2	0	2	Yes	98.93942591	250
10	42	1	0	140	226	0	1	178	0	0	2	0	2	Yes	32.36062422	74
11	66	0	0	178	228	1	1	165	1	1	1	2	3	No	62.40200317	177
12	60	1	0	117	230	1	1	160	1	1.4	2	2	3	No	17.8350778	8
13	38	1	2	138	175	0	1	173	0	0	2	4	2	Yes	73.27489338	218
14	49	1	2	120	188	0	1	139	0	2	1	3	3	No	60.6614375	164
15	55	1	0	140	217	0	1	111	1	5.6	0	0	3	No	61.50812954	171
16	67	1	0	100	299	0	0	125	1	0.9	1	2	2	No	71.35853138	209
17	29	1	1	130	204	0	0	202	0	0	2	0	2	Yes	61.18995016	167
18	59	1	3	170	288	0	0	159	0	0.2	1	0	3	No	67.34099791	194
19	47	1	2	138	257	0	0	156	0	0	2	0	2	Yes	23.28540315	39
20	52	1	1	134	201	0	1	158	0	0.8	2	1	2	Yes	45.46526146	130
21	46	1	2	150	231	0	1	147	0	3.6	1	0	2	No	36.32148125	94
22	38	1	2	138	175	0	1	173	0	0	2	4	2	Yes	73.27489338	218
23	62	0	0	124	209	0	1	163	0	0	2	0	2	Yes	37.25869026	96
24	65	1	0	110	248	0	0	158	0	0.6	2	2	1	No	19.11151485	16
25	52	1	3	118	186	0	0	190	0	0	1	0	1	Yes	64.53843816	183

Gambar 16: Menghitung Hasil Langkah 2 Secara *Ascending*

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	euclidean	rank	knn
50	0	1	120	244	0	1	162	0	1.1	2	0	2	Yes	0	1	Yes
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes
52	1	0	128	255	0	1	161	1	0	2	1	3	No	14.00749799	5	No
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes

Gambar 17: Mengumpulkan Kategori Y

age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	target	euclidean	rank	knn	klasifikasi
50	0	1	120	244	0	1	162	0	1.1	2	0	2	Yes	0	1	Yes	Sakit
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes	Sakit
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes	Sakit
52	1	0	128	255	0	1	161	1	0	2	1	3	No	14.00749799	5	No	Sehat
56	1	1	120	240	0	1	169	0	0	0	0	2	Yes	10.35422619	2	Yes	Sakit

Gambar 18: Kategori *Nearest Neighbor* Mayoritas

Naïve Bayes

Pada tahap ini akan dilakukan sebuah perhitungan dengan menggunakan model Algoritma *Naïve Bayes* dengan mengambil *dataset* yang diuji coba dengan menggunakan *tools RapidMiner Studio*. *Naïve Bayes* merupakan teknik pengklasifikasian sederhana dalam *data mining* yang menghitung probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai *dataset* yang diberikan. Berikut langkah-langkah pengerjaan serta hasil penghitungan dari Algoritma *Naïve Bayes* dengan *tools RapidMiner Studio*:

Menghitung P(Ci) jumlah kelas dari keterangan berdasarkan klasifikasi yang dibentuk.

Hasil = Sakit (Kelas Hasil = "Sakit"), P(Y=Sakit) = 131/256 = 0.512 (Jumlah data hasil "Sakit" pada *data training* dibagi dengan jumlah keseluruhan *data training*)

Hasil = Sehat (Kelas Hasil ="Sehat"), P(Y=Sehat) = 125/256 =0.488 (Jumlah data hasil "Sehat" pada *data training* dibagi dengan jumlah keseluruhan *data training*).

Menghitung P(X|Ci) jumlah kasus yang sama dengan kelas yang sama. Untuk atribut kuantitatif perhitungan yang dilakukan menggunakan *Microsoft Excel* yang mana hasilnya dapat dilihat pada Tabel 2.

Setelah didapatkan nilai *mean* dan standar deviasi dari atribut kuantitatif langkah selanjutnya yaitu menghitung nilai probabilitasnya dengan menggunakan sampel data untuk diuji menggunakan model algoritma *Naïve Bayes*. Nilai probabilitas dapat dilihat pada Tabel 3.

Sampel data tersebut memiliki hasil "Sakit". Kemudian untuk melakukan perhitungan probabilitas pada atribut kuantitif dapat dilakukan menggunakan rumus *Gaussian*. Hasil perhitungan probabilitas dari atribut kuantitatif dapat dilihat pada Tabel 4.

Tabel 2: Nilai Mean dan Standar Deviasi Setiap Atribut

	Kelas	
Age	Sakit	Sehat
Nilai *Mean*	54.230	54.117
Standar Deviasi	9.011	8.994
	Kelas	
Sex	Sakit	Sehat
Nilai *Mean*	0.75	0.75
Standar Deviasi	0.434	0.434
	Kelas	
Cp	Sakit	Sehat
Nilai *Mean*	0.960	0.945
Standar Deviasi	1.059	1.058
	Kelas	
Trestbps	Sakit	Sehat
Nilai *Mean*	130.325	130.203
Standar Deviasi	17.903	17.820
	Kelas	
Chol	Sakit	Sehat
Nilai *Mean*	247.837	248.141
Standar Deviasi	47.740	47.830
	Kelas	
Fbs	Sakit	Sehat
Nilai *Mean*	0.171	0.176
Standar Deviasi	0.377	0.381
	Kelas	
Restecg	Sakit	Sehat
Nilai *Mean*	0.587	0.578
Standar Deviasi	0.509	0.510
	Kelas	
Thalach	Sakit	Sehat
Nilai *Mean*	150.548	150.359
Standar Deviasi	23.040	22.972
	Kelas	
Exang	Sakit	Sehat
Nilai *Mean*	0.349	0.355
Standar Deviasi	0.478	0.480
	Kelas	
Oldpeak	Sakit	Sehat
Nilai *Mean*	1.059	1.065
Standar Deviasi	1.247	1.243
	Kelas	
Slope	Sakit	Sehat
Nilai *Mean*	1.369	1.367
Standar Deviasi	0.646	0.643
	Kelas	
Ca	Sakit	Sehat
Nilai *Mean*	0.825	0.816
Standar Deviasi	1.115	1.110
	Kelas	
Thal	Sakit	Sehat
Nilai *Mean*	2.369	2.363
Standar Deviasi	0.614	0.630

Tabel 3: Sampel *Data Testing*

Age	50
Sex	0
Cp	1
Trestbps	120
Chol	224
Fbs	0
Restecg	1
Thalach	162
Exang	0
Oldpeak	1.1
Slope	1
Ca	0
Thal	3
Target	Yes

Tabel 4: Hasil Probabilitas Atribut Kuantitatif

Atribut	Probabilitas	
	Sakit	Sehat
Age = 50	0.116	0.116
Sex = 0	0.243	0.243
Cp = 1	0.392	0.392
Ttrestbps = 120	0.071	0.071
Chol = 244	0.055	0.055
Fbs = 0	0.593	0.589
Restecg = 1	0.384	0.381
Thalach = 162	0.075	0.075
Exang = 0	0.450	0.448
Oldpeak = 1.1	0.353	0.353
Slope = 2	0.321	0.320
Ca = 0	0.295	0.296
Thal = 2	0.432	0.432

Kemudian setelah didapatkan hasil probabilitas atribut tipe numerik (kuantitatif) langkah selanjutnya yaitu menghitung dengan persamaan probabilitas akhir yang diuji. Berikut perhitungannya:

Perhitungan untuk Kelas = "Sakit"

=P(Sakit|F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, Probabilitas Kelas) = P(Sakit) x μ^{13} P(Fi)

*=P(Sakit)*P(age=50|Sakit)*P(sex=0|Sakit)*
**P(cp=1|Sakit)*P(trestbps=120|Sakit)*
**P(chol=244|Sakit)*P(fbs=0|Sakit)*
**P(restecg=1|Sakit)*P(thalach=162|Sakit)*
**P(exang=0|Sakit)*P(oldpeak=1.1|Sakit)*
**P(slope=2|Sakit)*P(ca=162|Sakit) *P(thal=2|Sakit)*
*=0.116*0.243*0.392*0.071*0.055*0.593*0.384*0.075*
**0.450*0.353*0.321*0.295*0.432*0.520*
=2.49444E-09

Perhitungan untuk Kelas = "Sehat"

=P(Sehat|F1, F2, F3, F4, F5, F6, F7, F8, F9, F10, F11, F12, F13, Probabilitas Kelas) = P(Sehat) x μ^{13}P(Fi)

*=P(Sehat)*P(age=50|Sehat)*P(sex=0|Sehat)*
**P(cp=1|Sehat)*P(trestbps=120|Sehat)*
**P(chol=244|Sehat)*P(fbs=0|Sehat)*
**P(restecg=1|Sehat)*P(thalach=162|Sehat)*
**P(exang=0|Sehat)*P(oldpeak=1.1|Sehat)*
**P(slope=2|Sehat)*P(ca=162|Sehat) *P(thal=2| Sehat)*
*=0.116*0.243*0.392*0.071*0.055*0.589*0.381*
**0.075*0. 448*0.353*0.320*0.296*0.4232*0.480*
=2.2718E-09

Berdasarkan hasil probabilitas akhir yang telah diuji bahwa nilai semua atribut untuk kelas Hasil = Sakit > Hasil = Sehat. Maka dapat ditentukan untuk solusi sampel *data testing* yang diuji termasuk dalam klasifikasi Hasil = "Sakit". Terbukti bahwa hasil sampel *data testing* sesuai dengan hasil pengujian.

Analisa Hasil Komparasi

Setelah melakukan pengujian terhadap tiga algoritma yaitu *Decesion Tree, KNN, Naïve Bayes* menggunakan *confussion matrix* dan *AUC*, maka dapat dibuat perbandingan terhadap tiga algoritma tersebut untuk memprediksi penyakit jantung yang dapat dilihat pada Tabel 5.

Tabel 5: Analisa Komparasi

Algoritma	Akurasi	*AUC*
Decesion Tree	84.77%	0.936
KNN	94.92%	0.990
Naïve Bayes	81.64%	0.901

Dari Tabel 5 dapat dilihat dari segi akurasi bahwa algoritma *KNN* lebih baik dibandingkan dengan algoritma *Decesion Tree* dan *Naïve Bayes*. Algoritma *KNN* memiliki nilai akurasi 94.92%, sedangkan algoritma *Decesion Tree* memiliki nilai akurasi sebesar 84.77% dan algoritma *Naïve Bayes* memiliki nilai akurasi 81.64%. Dengan demikian algoritma *KNN* lebih baik dibandingkan dengan algoritma *Decesion Tree* dan algoritma *Naïve Bayes*. Kemudian untuk nilai *AUC* yang ditarik dari kurva *ROC* menunjukkan bahwa algoritma *KNN* memiliki nilai sebesar 0.990, sedangkan algoritma *Decesion Tree* memiliki nilai sebesar 0.936 dan algoritma *Naïve Bayes* memiliki nilai sebesar 0.901.

Penutup

Berdasarkan hasil penelitian dan pengujian terhadap dataset diagnosa penyakit jantung maka dapat ditarik kesimpulan bahwa algoritma *KNN* memiliki akurasi lebih baik dibandingkan dengan algoritma *Decesion Tree* dan *Naïve Bayes*. Penggunaan *tools* dapat membantu dalam menghitung nilai akurasi maupun *AUC* dari tiga algoritma yang dibandingkan. Kemudian beberapa saran dari peneliti diharapkan baik, yaitu penggunaan algoritma klasifikasi lain seperti *SVM, Random Forest* dan lain sebagainya dapat dilakukan untuk dibandingkan dengan *KNN* agar dapat melihat algoritma mana yang lebih akurat dalam memprediksi penyakit jantung. Kemudian penggunaan algoritma yang dikombinasikan dengan metode lain seperti metode *PCA*, *Forward Selection* dan lain sebagainya.

Daftar Pustaka

[1] D. P. Utomo dan Mesran, "Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung", J. Media Inform. Budidarma, Vol. 4, No. 2, pp. 437–444, 2020.

[2] R. Annisa, "Analisis Komparasi Algoritma Klasifikasi Data Mining Untuk Prediksi Penderita Penyakit Jantung", J. Tek. Inform. Kaputama, Vol. 3, No. 1, pp. 22-28, 2019

[3] A. N. Safriandono, "Algoritma KNN Berbasis Forward Selection untuk Mendiagnosis Penyakit Jantung Koroner", Komputaki, Vol. 3, No. 1, pp. 1–16, 2017.

[4] M. Kumar, S. Shambhu and A. Sharma, "Classification of Heart Diseases Patients using Data Mining Techniques", Int. J. Res. Electron. Comput. Eng., Vol. 6,No. 3, pp. 1495–1499, 2018

[5] H. B. F. David and S. A. Belcy, "Heart Disease Prediction Using Data Mining Techniques", J. Soft Comput., Vol. 09, No. 01, pp. 1817–1823, 2018.

[6] S. Anitha and N. Sridevi, "Heart Disease Prediction Using Data Mining Techniqies", J. Anal. Comput., Vol. 13, No. 2, pp. 48–55, 2019.

[7] Sundas Naqeeb Khan, Nazri Mohd Nawi, Asim Shahzad, Arif Ullah, Muhammad Faheem Mushtaq, Jamaluddin Mir and Muhammad Aamir, "Comparative Analysis for Heart Disease Prediction", JOIV : International Journal on Informatics Visualization, Vol. 1, No. 4, pp. 227–231, 2017.

[8] S. Shylaja and R. Muralidharan, "Comparative Analysis of Various Classification and Clustering Algorithms for Heart Disease Prediction System", Int. J. Biometrics Bioinforma., Vol. 10, No. 4, pp. 74–77, 2018.

[9] I. E. Putri, D. Rahmawati and Y. Azhar, "Comparison of Data Mining Classification Methods to Detect Heart Disease", J. PILAR Nusa Mandiri, Vol. 16, No. 2, pp. 213–218, 2020.

[10] P. Shetgaonkar and S. Aswale, "Heart Disease Prediction Using Data Mining Techniques", Int. J. Eng. Res. Technol., Vol. 10, No. 2,pp. 281-286, 2021

[11] Kuncahyo Setyo Nugroho, "Confusion Matrix untuk Evaluasi Model pada Supervised Learning Contoh: Untuk Pemodelan Klasifikasi Biner", diakses daring pada https://medium.com/@ksnugroho/confusion-matrix- untuk-evaluasi-model-pada-unsupervised-machine- learning-bc4b1ae9ae3f, 2019.

[12] Anonym, "Heart Disease Dataset", Public Health Dataset, diakses daring pada https://www.kaggle.com/johnsmith88/heart-disease-dataset, 2022.