ANALISIS PERBANDINGAN KLASIFIKASI PREDIKSI PENYAKIT HEPATITIS DENGAN MENGGUNAKAN ALGORITMA *K-NEAREST NEIGHBOR*, NAÏVE BAYES DAN *NEURAL NETWORK*

Sulastri[1], Kristophorus Hadiono[2], Muchamad Taufiq Anwar[3]
[1,2,3]Fakultas Teknologi Informasi, Universitas Stikubank
e-mail: [1]sulastri@edu.unisbank.ac.id, [2]kristophorus.hadiono@edu.unisbank.ac.id,
[3]taufiq@edu.unisbank.ac.id

Abstrak

Hepatitis merupakan penyakit yang diderita oleh banyak orang, bahkan bisa menyebabkan kematian. Prediksi awal dapat mencegah kematian tersebut yaitu dengan mengumpulkan data pasien hepatitis yang dilihat dari faktor - faktornya. Faktor-faktor tersebut antara lain Protime, Alk Phosphat, Albumin, Bilirubin dan Usia. Untuk mengolah data tersebut, dibutuhkan Data Mining. Salah satu metode data mining yang digunakan pada penelitian ini adalah klasifikasi.

Tujuan penelitian ini yaitu bagaimana memprediksi hidup atau meninggalnya pasien penyakit hepatitis dengan tingkat akurasi dan mencari atribut paling berpengaruh terhadap prediksi hidup atau meninggalnya pasien penyakit hepatitis dengan menggunakan algoritma Algoritma *K-Nearest Neighbor*, Naïve Bayes Dan *Neural Network* dan kemudian membandingkan ketiga hasil analisis dari ketiga algoritma tersebut.

Dari hasil analisis 20 atribut dilakukan 3 kali percobaan dengan algoritma Naïve Bayes didapat model klasifikasi dengan tingkat akurasi yang terbaik yaitu 76.92 %, tingkat error 23.01% dan atribut *Acites* dan *Spider* merupakan atribut yang berpengaruh terhadap keputusan hidup atau meninggalnya pasien yang terkena penyakit hepatitis.Dengan menggunakan Algoritma Neural Network didapat model klasifikasi dengan tingkat akurasi yang terbaik yaitu 82,97%, tingkat error 17.03% dan atribut yang paling berpengaruh yaitu *anorexia*, *spiders* dan *protime*. Dengan menggunakan algoritma *K-Nearest Neighbor* didapat model klasifikasi dengan tingkat akurasi terbaik yaitu 93%, tingkat error 7% dan atribut yang paling berpengaruh terhadap penderita penyakit hepatitis yaitu *Albumin*.

Kata Kunci: Hepatitis, Data Mining, *K-Nearest Neighbor*, *Naïve Bayes, Neural Network*.

1. PENDAHULUAN

Dalam dunia kesehatan, diagnosis penyakit menjadi hal yang sangat sulit dilakukan. Namun demikian catatan rekam medis telah menyimpan gejala-gejala penyakit pasien dan diagnosis penyakitnya. Hal seperti ini tentu sangat berguna bagi para ahli kesehatan. Mereka dapat menggunakan catatan rekam medis yang sudah ada sebagai bantuan untuk mengambil keputusan tentang diagnosis penyakit pasien [1].

Salah satu penyakit yang cukup banyak penderitanya adalah hepatitis. Secara umum hepatitis didefinisikan sebagai suatu penyakit yang ditandai dengan suatu peradangan yang terjadi pada organ tubuh seperti hati. Hepatitis diakibatkan berbagai faktor dimana tiap faktor mempunyai karakter khas, maka timbullah berbagai macam hepatitis yang berbeda satu sama lain [2]. Untuk itu perlu adanya prediksi awal penentuan hepatitis. Prediksi awal ini perlu dilakukan karena banyak yang menyepelekan penyakit hepatitis. Hal ini dapat dilakukan dengan pengumpulan data pasien ataupun data hasil cek kesehatan.

Dengan banyaknya data yang diperoleh ataupun dikumpulkan, tidaklah mudah untuk mengolah data tersebut menjadi informasi yang bermanfaat. Maka dari itu, dibutuhkan sebuah ilmu yang dapat mengolah data tersebut. Data mining merupakan suatu ilmu yang mempelajari

bagaimana data itu diolah sehingga menghasilkan sebuah informasi yang dapat dimanfaatkan untuk berbagai bidang terutama bidang kesehatan. Banyak pemanfaatan data mining untuk melakukan prediksi terhadap suatu penyakit. Salah satu contohnya yaitu penyakit hepatitis. Untuk memprediksi penyakit ini, digunakanlah salah satu teknik dari data mining yaitu klasifikasi.

Klasifikasi adalah sebuah teknik pengelompokkan data ke dalam beberapa kategori yang sudah ditentukan. Dalam klasifikasi, data yang diperoleh terlebih dahulu dilakukan pengolahan dengan menggunakan variabel yang ada untuk menentukan data tersebut termasuk kategori yang mana. Dalam penelitian ini, peneliti menggunakan Algoritma *K- Nearest Neighbor,* Naïve Bayes dan *Neural Network,* kemudian membandingkan prediksi seorang pasien hepatitis akan meninggal atau hidup dengan melihat variabel yang berpengaruh dan juga hasil akurasinya.

2. METODE PENELITIAN

Penelitian ini dilakukan dengan menggunakan data yang diambil dari UCI Machine Learning berupa data pasien hepatitis dengan Data berjumlah 155 record dan terdiri dari 19 variabel penjelas dan 1 variabel respon.. Berikut adalah variabel datanya:

Tabel 1. Variabel Data

Variable	Keterangan	
Class	variabel respon	1=die, 2=live
Age	usia pasien	10-80
Sex	jenis kelamin pasien	1=male, 2=female
Steroid	senyawa organik lemak sterol tidak terhidrolisis yang dapat dihasil reaksi penurunan dari terpena atau skualena.	1=no, 2= yes
Antivirals	obat yang menghambat atau merusak replikasi virus.	1=no, 2=yes
Fatigue	suatu kelelahan yang terjadi pada syaraf dan otot-otot	1=no, 2=yes
Malaise	lemas, lesu, letih, dan merasa sakit.	1=no, 2=yes
anorexia	gangguan makan yang ditandai dengan rasa takut yang berlebihan bila berat badan bertambah, dan gangguan persepsi pada bentuk tubuh.	1=no, 2=yes
liver big	penyakit yang disebabkan oleh berbagai faktor yang merusak hati, seperti virus dan penggunaan alkohol.	1=no, 2=yes
liver firm		1=no, 2=yes
spleen palpable	kerusakan organ jaringan limfatik	1=no, 2=yes
Spiders	sekumpulan pembuluh darah abnormal dekat permukaan kulit	1=no, 2=yes
Ascites	penumpukan cairan di rongga perut	1=no, 2=yes
Varices	pembuluh darah vena yang membengkak dan tampak dekat dari permukaan kulit.	1=no, 2=yes
Bilirubin	senyawa pigmen berwarna kuning yang merupakan produk katabolisme enzimatik biliverdin oleh biliverdin reduktase.	0.39-4.00
alk phosphate	untuk mengukur tingkat enzim fosfatase alkali dalam darah.	33-250
Sgot	enzim yang biasanya ditemukan pada hati (liver), jantung, otot, ginjal, hingga otak.	13-500
Albumin	protein utama yang terdapat dalam darah manusia yang diproduksi oleh organ hati.	2.1-6.0
Protime	disintesis oleh hati dan merupakan prekursor tidak aktif dalam proses pembekuan.	10-90

Histology	pemeriksaan contoh sampel jaringan pada pasien hepatitis	1=no, 2=yes

Berikut adalah proses Knowlegde Data Mining Discovery (KDD):

a. *Data Selection*

Data yang akan digunakan dalam bentuk excel, untuk bisa diolah menggunakan Bahasa R, format data dirubah menjadi CSV untuk diimport ke RStudio.

class	age	sex	steroid	antivirals	fatigue	malaise	anorexia	liver_big	liver_firm	spleen_palpable	spiders	ascites	varices	bilirubin	alk_phosphate	sgot	albumin	protime	histology
2	30	2	1	2	2	2	2	1	2	2	2	2	2	1.00	85	18	4.0	?	1
2	50	1	1	2	1	2	2	1	2	2	2	2	2	0.90	135	42	3.5	?	1
2	78	1	2	2	1	2	2	2	2	2	2	2	2	0.70	96	32	4.0	?	1
2	31	1	?	1	2	2	2	2	2	2	2	2	2	0.70	46	52	4.0	80	1
2	34	1	2	2	2	2	2	2	2	2	2	2	2	1.00	?	200	4.0	?	1
2	34	1	2	2	2	2	2	2	2	2	2	2	2	0.90	95	28	4.0	75	1
1	51	1	1	2	1	2	1	2	2	1	1	2	2	?	?	?	?	?	1
2	23	1	2	2	2	2	2	2	2	2	2	2	2	1.00	?	?	?	?	1
2	39	1	2	2	1	2	2	2	1	2	2	2	2	0.70	?	48	4.4	?	1
2	30	1	2	2	2	2	2	2	2	2	2	2	2	1.00	?	120	3.9	?	1
2	39	1	1	1	2	2	2	1	1	2	2	2	2	1.30	78	30	4.4	85	1
2	32	1	2	1	1	2	2	2	1	2	1	2	2	1.00	59	249	3.7	54	1
2	41	1	2	1	1	2	2	2	1	2	2	2	2	0.90	81	60	3.9	52	1
2	30	1	2	2	1	2	2	2	1	2	2	2	2	2.20	57	144	4.9	78	1
2	47	1	1	1	2	2	2	2	2	2	2	2	2	?	?	60	?	?	1
2	38	1	1	2	1	1	1	2	2	2	2	1	2	2.00	72	89	2.9	46	1
2	66	1	2	2	1	2	2	2	2	2	2	2	2	1.20	102	53	4.3	?	1
2	40	1	1	2	1	2	2	2	1	2	2	2	2	0.60	62	166	4.0	63	1
2	38	1	2	2	2	2	2	2	2	2	2	2	2	0.70	53	42	4.1	85	2
2	38	1	1	1	2	2	2	1	1	2	2	2	2	0.70	70	28	4.2	62	1

Gambar 1. Data Mentah

b. *Pre- processing / Cleaning*

Pada tahap ini data yang akan dipakai akan dilakukan *cleaning* karena masih terdapat data yang kosong atau tidak memiliki nilai. Proses *cleaning* tersebut menggunakan rumus *average* atau rata-rata dari kolom yang terdapat record yang kosong.

class	age	sex	steroid	antivirals	fatigue	malaise	anorexia	liver_big	liver_firm	spleen_palpable	spiders	ascites	varices	bilirubin	alk_phosphate	sgot	albumin	protime	histology
2	30	2	1	2	2	2	2	1	2	2	2	2	2	1	85	18	4	66	1
2	50	1	1	2	1	2	2	1	2	2	2	2	2	0,9	135	42	3,5	46	1
2	78	1	2	2	1	2	2	2	2	2	2	2	2	0,7	96	32	4	52	1
2	31	1	2	1	2	2	2	2	2	2	2	2	2	0,7	46	52	4	80	1
2	34	1	2	2	2	2	2	2	2	2	2	2	2	1	135	200	4	100	1
2	34	1	2	2	2	2	2	2	2	2	2	2	2	0,9	95	28	4	75	1
1	51	1	1	2	1	2	1	2	2	1	1	2	2	1	75	20	4	31	1
2	23	1	2	2	2	2	2	2	2	2	2	2	2	1	95	60	3	90	1
2	39	1	2	2	1	2	2	2	1	2	2	2	2	0,7	71	48	4,4	62	1
2	30	1	2	2	2	2	2	2	2	2	2	2	2	1	150	120	3,9	57	1
2	39	1	1	1	2	2	2	1	1	2	2	2	2	1,3	78	30	4,4	85	1
2	32	1	2	1	1	2	2	2	1	2	1	2	2	1	59	249	3,7	54	1
2	41	1	2	1	1	2	2	2	1	2	2	2	2	0,9	81	60	3,9	52	1
2	30	1	2	2	1	2	2	2	1	2	2	2	2	2,2	57	144	4,9	78	1
2	47	1	1	1	2	2	2	2	2	2	2	2	2	0,9	72	60	3,5	46	1
2	38	1	1	2	1	1	1	2	2	2	2	1	2	2	72	89	2,9	46	1
2	66	1	2	2	1	2	2	2	2	2	2	2	2	1,2	102	53	4,3	50	1
2	40	1	1	2	1	2	2	2	1	2	2	2	2	0,6	62	166	4	63	1
2	38	1	2	2	2	2	2	2	2	2	2	2	2	0.70	53	42	4.1	85	2

Gambar 2. Data yang telah Cleaning

c. *Transformation*

Setelah dilakukan cleaning, selanjutnya dilakukan tahap transformasi atau mengubah data. Pada penelitian ini, data yang dipakai berupa numerik sehingga dilakukan perubahan dengan merubah data numeric menjadi data huruf menggunakan fungsi IF pada excel.

age	sex	steroid	antivirals	fatigue	malaise	anorexia	liver_big	liver_firm	spleen_pa	spiders	ascites	varices	bilirubin	alk_phosp	sgot	albumin	protime	histology	class
30	2	1	2	2	2	2	1	2	2	2	2	2	1	85	18	4	66	1	LIVE
50	1	1	2	1	2	2	1	2	2	2	2	2	0.9	135	42	3.5	46	1	LIVE
78	1	2	2	1	2	2	2	2	2	2	2	2	0.7	96	32	4	52	1	LIVE
31	1	2	1	2	2	2	2	2	2	2	2	2	0.7	46	52	4	80	1	LIVE
34	1	2	2	2	2	2	2	2	2	2	2	2	1	135	200	4	100	1	LIVE
34	1	2	2	2	2	2	2	2	2	2	2	2	0.9	95	28	4	75	1	LIVE
51	1	1	2	1	2	1	2	2	1	1	2	2	1	75	20	4	31	1	DIE
23	1	2	2	2	2	2	2	2	2	2	2	2	1	95	60	3	90	1	LIVE
39	1	2	2	1	2	2	2	1	2	2	2	2	0.7	71	48	4.4	62	1	LIVE
30	1	2	2	2	2	2	2	2	2	2	2	2	1	150	120	3.9	57	1	LIVE
39	1	1	1	2	2	2	1	1	2	2	2	2	1.3	78	30	4.4	85	1	LIVE
32	1	2	1	1	2	2	2	1	2	1	2	2	1	59	249	3.7	54	1	LIVE
41	1	2	1	1	2	2	2	1	2	2	2	2	0.9	81	60	3.9	52	1	LIVE
30	1	2	2	1	2	2	2	1	2	2	2	2	2.2	57	144	4.9	78	1	LIVE
47	1	1	1	2	2	2	2	2	2	2	2	2	0.9	72	60	3.5	46	1	LIVE
38	1	1	2	1	1	1	2	2	2	2	1	2	2	72	89	2.9	46	1	LIVE
66	1	2	2	1	2	2	2	2	2	2	2	2	1.2	102	53	4.3	50	1	LIVE
40	1	1	2	1	2	2	2	1	2	2	2	2	0.6	62	166	4	63	1	LIVE
38	1	2	2	2	2	2	2	2	2	2	2	2	0.7	53	42	4.1	85	2	LIVE
38	1	1	1	2	2	2	1	1	2	2	2	2	0.7	70	28	4.2	62	1	LIVE
22	2	2	1	1	2	2	2	2	2	2	2	2	0.9	48	20	4.2	64	1	LIVE
27	1	2	2	1	1	1	1	1	1	1	2	2	1.2	133	98	4.1	39	1	LIVE

Gambar 3. Data yang Telah Ditransformasi

d. Data Mining

Data yang telah ditranformasi kemudian diklasifikasi dengan menggunakan algoritma *K-Neurest Neighbor, Naïve Bayes* dan *Neural Network*. Proses pengolahan data tersebut menggunakan software RStudio. Pada penilitian ini akan dilakukan percobaan sebanyak 3 kali dengan perbadingan training set dan testing set yang berbeda dan kemudian dianalisa dengan menggunakan 3 algoritma yang berbeda. Perbandingannya adalah sebagai berikut :

Tabel 2. Pembagian Data Training dan Data Testing

	Traning set	Testing set
Percobaan 1	70 %	30 %
Percobaan 2	75 %	25 %
Percobaan 3	80 %	30 %

e. *Interpretation / Evaluation*

Dalam tahap ini akan terlihat hasil pola yang terbentuk dari proses data mining mengunakan *Algoritma K- Neurest Neighbor, Naïve Bayes dan Neural Network*, sehingga informasi yang didapatkan lebih dipahami pembaca, ditampilkan dalam bentuk grafik gambar.

3. HASIL DAN PEMBAHASAN

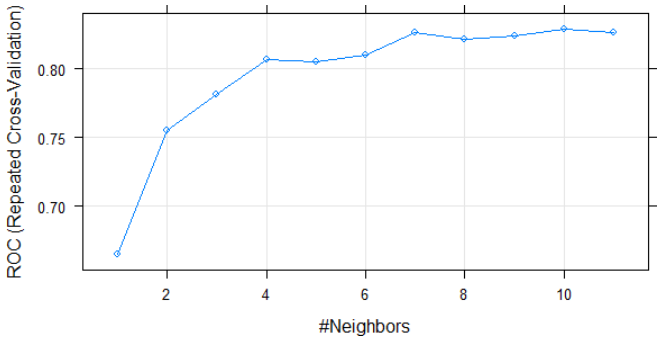
a. Implementasi Dengan Algoritma K-Nearest Neighbor

Pengolahan data mining menggunakan Bahasa R. Proses awal dimulai dengan menginstal *package* yang digunakan untuk mendukung proses perhitungan. Pertama *package caret* atau *Classification* dan *Regress Training. Package* memiliki fungsi untuk merampingkan proses pelatihan model untuk masalah klasifikasi dan regresi yang kompleks sampai contoh-contoh data (training maupun testing), ringkasan informasi data, parameter kontrol, plot, evaluasi model, dan lain-lain. Kedua *package pROC. Package* ini digunakan untuk memvisualisasikan, menghaluskan, dan membandingkan karakteristik operasi penerima (kurva ROC) area di bawah kurva (AUC). Ketiga *package mlbench*. Digunakan untuk mengetahui kumpulan masalah tentang tolak ukur pembelajaran mesin dan dunia nyata termasuk beberapa set data dari repository UCI.

Memasukan data yang akan digunakan dalam perhitungan. Pada R data yang dibaca berupa format .csv. Data yang sudah dimasukkan dibuat data sampel atau pembagian data(data training dan data testing). Dari fungsi data sampel ini digunakan library *caret* yaitu untuk membuat data training dan data testing. Pada penelitian ini mengguanakan tiga percobaan pembagian data yaitu 70% data training atau didapatkan 108 data training dari 155 record data, 75% data training atau didapatkan 1116 data training dari 155 record data dan 80% data training atau didapatkan 124 data training dari 155 record data. Pada algoritma K-Nearest Neighbor pembagian data dilakukan dengan syarat data training harus lebih banyak dari data testing atau artinya data training dan data testing tidak boleh memiliki nilai yang sama (50% data training dan 50% data testing). Dari pembagian data didapatkan hasil akurasi yang tinggi dengan perbandingan data 80% data training dan 20% data testing (124 record data training dan 31 record data testing).

Menghitung nilai K tertinggi dengan jumlah K = 11. Nilai K didapatkan melalui akar dari nilai training yang terbanyak dari percobaan yang sudah dilakukan dalam penelitian. Selanjutnya memvisualisasikan algoritma *K- Nearest Neighbor* menggunakan package *pROC* kemudian ditampilkan berupa gambar diagram. Pada gambar diagram terdapat dua keterangan tabel yaitu vertikal berupa ROC (*Repeated Cross-Validation*) dan horizontal berupa *#Neighbors*. Pada keterangan ROC (*Repeated Cross-Validation*) terdapat 3 set lipatan yang didapatkan yaitu mulai dari 0,70 yang selanjutnya berkelipatan 0,05 hingga didapatkan angka

tertinggi 0,85. Pada keterangan *#Neighbors* jarak antar nilai K yaitu berkelipatan 2 dengan nilai K = 11. Dapat diartikan bahwa digram tersebut yaitu hasil dari nilai ROC yang sudah dilakukan perhitungan pada fungsi algoritma *K-Nearest Neighbor*. Titik awal pada diagram yaitu K=1 bernilai 0,66 dan divisualisasikan pada diagram berada dibawah angka 0,70 pada nilai ROC. Titik kedua pada diagram yaitu K=2 bernilai 0,76 dan divisualisasikan pada diagram berada dibawah angka 0,80 dan diatas angka 0,75 pada nilai ROC. Dan titik yang ketiga yaitu K=3 yang bernilai 0,78 divisualisasikan pada diagram berada dibawah angka 0,80 dan diatas 0,75 pada nilai ROC, hingga didapatkan nilai K tertinggi pada nilai ROC yaitu pada K=10 yang bernilai 0,83. (lihat gambar41)



Gambar 4. Hasil perhitungan *K-Nearest Neighbors* berupa diagram

Mencari variabel data yang sangat berpengaruh pada data yang digunakan. Variabel data merupakan penyebab utama pada penderita penyakit Hepatitis. Dari hasil yang tampil terdapat variabel yang memiliki nilai yang sangat tinggi dan didapatkan angka 100 yaitu pada variabel albumin dan terdapat variabel yang sangat tidak berpengaruh karena didapatkan angka 0 pada variabel liver_firm. Variabel yang mendapatkan angka diatas 0 merupakan variabel yang berpengaruh terhadap penderita penyakit hepatitis. Syarat variabel itu bisa digunakan atau berpengaruh pada perhitungan yaitu variabel harus bernilai 20 atau lebih.(lihat gambar 5)

```
ROC curve variable importance

               Importance
albumin            100.00
bilirubin           81.51
spiders             72.09
ascites             59.86
histology           55.59
malaise             55.12
protime             52.95
age                 52.37
fatigue             51.14
varices             37.62
spleen_palpable     26.21
steroid             24.87
alk_phosphate       24.63
anorexia            19.43
sex                 18.14
sgot                15.80
antivirals          10.42
liver_big            8.25
liver_firm           0.00
```

Gambar 5. Variabel yang berpengaruh

Melakukan prediksi dengan menggunakan dua level kelas berupa “DIE” dan “LIVE”. Prediksi menggunakan data baru dengan menggunakan data testing yang akan menjadi acuan. Penelitian ini menggunakan data testing 10% atau 31 record data dari 155 record. Hasil dari fungsi prediksi data ini akan digunakan pada fungsi *Confusion matrix*. (lihat gambar 6)

```
> pred <- predict(fit, newdata = data.test3)
> pred
 [1] LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE
[14] LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE DIE  LIVE LIVE LIVE LIVE
[27] LIVE DIE  LIVE LIVE DIE
Levels: DIE LIVE
```

Gambar 6. Hasil prediksi

Menjalankan fungsi *Confusion matrix*. Didapat hasil pada referensi prediksi 16 data testing dengan *class* DIE dan LIVE (3 data DIE dan 28 data LIVE). Data testing yang dijalankan dengan fungsi *Confusion matrix* selanjutnya akan masuk kedalam tabel "DIE" dan "LIVE". Dari hasil tersebut terdapat data eror yang mana pada tabel "DIE" berisi 1 data "LIVE", pada tabel "LIVE" berisi 1 data "DIE" hasil itu dinyatakan data eror karena tidak sesuai. Pada perhitungan menggunakan fungsi ini mendapatkan nilai akurasi 0,93 atau dalam bentuk porsentase 0,93%. Sensitivitas dari data tersebut didapatkan nilai 0,96 sebagai proporsi hasil positif dari jumlah sampel yang sebenarnya positif. Spesifisitas bernilai 0,66 atau jumlah sampel yang sebenarnya negatif. Nilai prediktif positif bernilai 0,96, hasil dapat didefinisikan sebagai persen dari nilai positif yang sebenarnya. Nilai prediktif negative bernilai 0,66 dan didefinisikan sebagai persen nilai negatif yang sebenarnya. Hasil tersebut dinyatakan sangat baik karena melebihi syarat batas nilai akurasi sebesar 0,75%.

Selain itu telah dilakukan dua kali pengujian dengan menggunakan data training sebanyak 108 data dan data testing sebanyak 47 data, dengan data training sebanyak 116 data dan data testing sebanyak 39 data. Hasil ditunjukkan berupa tabel.(lihat tabel 3)

Tabel 3. Hasil Pengukuran Pengujian

Pengujian	Data Training	Data Testing	Akurasi
Percobaan 1	108	47	0,85%
Percobaan 2	116	39	0,84%
Percobaan 3	124	31	0,93%

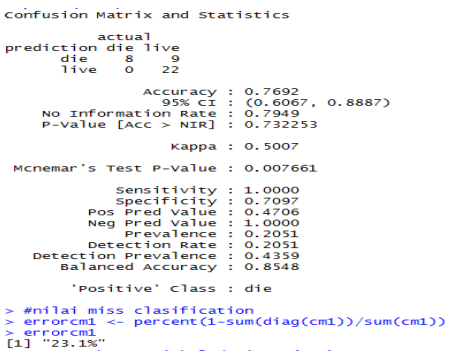
b. Implementasi dengan Algoritma Naïve Bayes

```
myformula1<-
class~age+sex+steroid+antivirals+fatigue+malaise+anorexia+liver_big+
liver_firm+spleen_palp
able+spiders+ascites+varices+bilirubin+alk_phosphate+sgot+albumin+pr
otime+histology model3 <- naiveBayes(myformula1, data = train3)
model3
```

Gambar 7. Hasil Pemodelan *Naive Bayes*

Dari pemodelan *naïve bayes* yang dilakukan didapatkan hasil probabilitas dari setiap attribute untuk kasus DIE dan LIVE dengan data yang digunakan sebanyak 116 data. Kemudian hasil prediksi tersebut dilanjutkan dengan menghitung tingkat akurasi menggunakan metode *counfusion matrix*. Berikut merupakan hasil perhitungan counfusion matrix dengan menggunakan package caret dan package scales untuk menghitung tingkat eror.

```
cm3 = table (prediksi3, test3$class, dnn =
list("prediction","actual"))
cmt3 = confusionMatrix(cm3)
print(cmt3)
```



```
Confusion Matrix and Statistics

          actual
prediction die live
      die    8    9
      live   0   22

               Accuracy : 0.7692
                 95% CI : (0.6067, 0.8887)
    No Information Rate : 0.7949
    P-Value [Acc > NIR] : 0.732253

                  Kappa : 0.5007

 Mcnemar's Test P-Value : 0.007661

            Sensitivity : 1.0000
            Specificity : 0.7097
         Pos Pred Value : 0.4706
         Neg Pred Value : 1.0000
             Prevalence : 0.2051
         Detection Rate : 0.2051
   Detection Prevalence : 0.4359
      Balanced Accuracy : 0.8548

       'Positive' Class : die

> #nilai miss clasification
> errorcm1 <- percent(1-sum(diag(cm1))/sum(cm1))
> errorcm1
[1] "23.1%"
```

Gambar 8. Hasil akurasi

Dari hasil Gambar 5 didapat hasil akurasi sebesar 76.92 atau 76.92% dan tingkat eror sebesar 23.1 atau 23.1% yang berarti bahwa model tersebut dapat memprediksi data baru secara benar dengan tingkat keberhasilan sangat tinggi.

Selain itu telah dilakukan juga dau kali pengujian dengan menggunakan data *training* 109 data dan data *testing* 46 data, dan juga menggunakan data *training* 124 data dan data *testing* 31 data. Hasilnya ditunjukan pada tabel 4, dibawah ini :

Tabel 4. hasil percobaan

Kegiatan	Data *Training*	Data *Testing*	Akurasi %	*Error Rate %*
Percobaan 1	93 data	62 data	76.06%	23.04%
Percobaan 2	109 data	46 data	76.92%	23.01%
Percobaan 3	124 data	31 data	74.19%	25.08%

Dari tabel 3 menunjukan bahwa tiga kali percobaan menggunakan algortima *naive bayes* memliki hasil akurasi yang berbeda dan percobaan 2 memiliki tingkat akurasi tertinggi yaitu sebesar 76.92% dengan tingkat error sebesar 23.01%.

c. Implementasi Dengan Algoritma Neural Network

Proses pengolahan data mining ini menggunakan Bahasa R melalui software RStudio. Proses awal yaitu menginstall package *nnet*, *NeuralNetTools* dan *partykit*. *Package nnet* memiliki fungsi untuk menghitung data dengan rumus *Neural Network*. *Package NeuralNetTools* digunakan untuk menampilkan atau memvisualisasi model dari *Neural Network*. *Package partykit* digunakan untuk menampilkan model pohon keputusan. Memasukkan data yang akan digunakan. Data yang akan diimpor ke RStudio terlebih dahulu diubah ke dalam bentuk .csv. Setelah diimpor, kemudian membuat data sampel acak dengan fungsi set.seed(8) untuk menghasilkan sample acak yang sama setiap saat dan menjaga konsistensi serta meningkatkan akurasi. Setelah membuat sampel acak, kemudian membuat pembagian data training dan data testing. Penelitian ini menggunakan 3 kali percobaan dengan pembagian data training dan testing masing-masing sebesar 70%-30%,75%-25%, dan 80%-20%.

Melakukan perhitungan dengan fungsi nnet pada RStudio. Pertama install fungsi nnet. Kemudian masukkan formula atau rumus nnet. Pada iterasi maksimal 100 belum diperoleh hasil yang diharapkan. Untuk itu dilakukan percobaan kedua dan seterusnya sampai pada percobaan keempat dengan maksimal iterasi 400. Pada percobaan ini didapat hasil yang diharapkan yaitu *converged*. Apabila sudah mencapai *converged* berarti iterasi berhenti dan nilai dari perhitungan terseut sudah tetap atau tidak berubah dari nilai iterasi sebelumnya.

```
> hepa<-nnet(class~.,data=data.train,size=3,decay=5e-4,maxit=400)
# weights:  64
initial  value 107.795920
iter  10 value 59.523739
iter  20 value 54.797722
iter  30 value 38.629814
iter  40 value 32.876371
iter  50 value 23.533446
iter  60 value 22.519210
iter  70 value 18.116466
iter  80 value 16.805530
iter  90 value 14.194084
iter 100 value 14.131779
iter 110 value 14.105984
iter 120 value 14.085306
iter 130 value 14.080934
iter 140 value 14.078672
iter 150 value 14.078140
iter 160 value 14.077958
final  value 14.077813
converged
```

Gambar 9. Hasil perhitungan yang sudah diharapkan dengan rumus nnet

Menampilkan data seleksi dari data testing dan menampilkan data hasil prediksi. Aktual merupakan variabel class dengan jumlah data sebanyak data testing yaitu sebanyak 47. Untuk prediksi yaitu hasil prediksi dari data testing dengan jumlah data yang sama dengan data testing.

```
> aktual<-data.test$class
> aktual
 [1] LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE
 LIVE
[17] LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE DIE  LIVE LIVE DIE  LIVE LIVE
 DIE
[33] LIVE LIVE LIVE LIVE LIVE LIVE LIVE LIVE DIE  LIVE DIE  DIE  LIVE LIVE LIVE
Levels: DIE LIVE
> prediksi<-predict(hepa,data.test,type = "class")
> prediksi
 [1] "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "DIE"  "LIV
E" "DIE"
[13] "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "LIVE" "DIE"  "LIVE" "LIVE" "LIVE" "LIV
E" "LIVE"
[25] "LIVE" "LIVE" "LIVE" "LIVE" "DIE"  "LIVE" "LIVE" "DIE"  "LIVE" "DIE"  "LIV
E" "LIVE"
[37] "LIVE" "DIE"  "LIVE" "LIVE" "LIVE" "LIVE" "DIE"  "DIE"  "LIVE" "LIVE" "DIE
"
```

Gambar 10. Data seleksi dari data testing dan data hasil prediksi

Menampilkan tabel hasil prediksi. Hasil persentase ditentukan dengan seberapa akurat pr ediksi terhadap data testing, yaitu prediksi DIE yang cocok dengan aktual DIE harus lebih banyak begitu juga prediksi LIVE yang cocok dengan aktual LIVE. Pada gambar 10 diatas didapat hasil prediksi DIE yang cocok dengan aktual DIE sebanyak 4 dari 6 data sedangkan hasil prediksi LIVE yang cocok dengan aktual LIVE sebanyak 35 dari 41 data.

```
> tabelhasil<-table(aktual,prediksi)
> tabelhasil
        prediksi
aktual DIE LIVE
  DIE    4    2
  LIVE   6   35
```

Gambar 11. Tabel hasil data testing dengan data prediksi

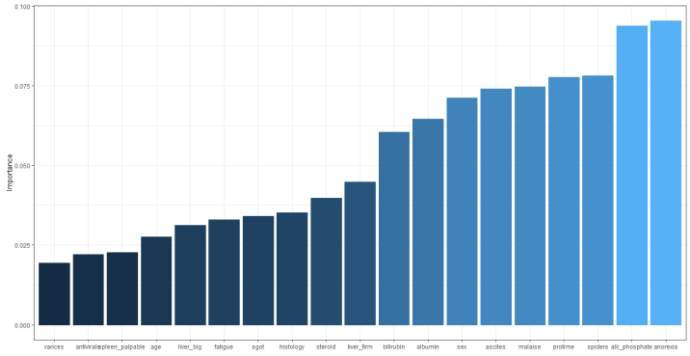
Menghitung persentase perhitungan *Algoritma Neural Network* dan menghitung jumlah error pada data yang digunakan. Dilakukan perhitungan matriks pada tabelhasil dengan fungsi *diag* yang kemudian dijumlah dengan fungsi *sum.* Hasil tersebut kemudian dibagi dengan banyaknya jumlah data testing sehingga didapat persentase sebesar 0.8297 atau 82,97%. Untuk error dilakukan perhitungan rata-rata dari data diluar data aktual yang dicocokkan dengan data prediksi sehingga mendapatkan hasil sebesar 0.1702. Hasil ini merupakan hasil yang cukup bagus mengingat jumlah data yang tidak begitu banyak dan dengan jumlah variabel yang cukup sedikit.

```
> sum(diag(table(aktual,prediksi)))/47
[1] 0.8297872
> error<-mean(aktual!=prediksi)
> error
[1] 0.1702128
```

Gambar 12. Persentase hasil perhitungan dan jumlah error

Menampilkan visualisasi *Neural Network*. Sebelum menampilkan visualisasi, pertama harus menginstall package *NeuralNetTools* terlebih dahulu. Pada package tersebut terdapat beberapa fungsi yang dipakai yaitu *plotnet, summary,* dan *garson.* Fungsi *plotnet* digunakan

untuk menampilkan visualisasi *Neural Network*. Fungsi *summary* digunakan untuk mengetahui bobot dari visualisasi *Neural Network.* Sedangkan *garson* digunakan untuk menampilkan grafik variabel paling berpengaruh. Pada penelitian ini, variabel paling dominan dengan fungsi *garson* yaitu *anorexia*.



Gambar 13. Grafik variabel dominan dengan fungsi *garson*

Tabel 5. Hasil Pengukuran Pengujian Dengan Neural Network

Pengujian	Data Training	Data Testing	Akurasi
Percobaan 1	108	47	82,97%
Percobaan 2	116	39	82,05%
Percobaan 3	124	31	77,41%

4. KESIMPULAN DAN SARAN

Berdasarkan percobaan yang telah dilakukan didapat bahwa dengan menngunakan algoritma Naïve Bayes didapat model klasifikasi dengan tingkat akurasi yang terbaik pada percobaan 2 yaitu 76.92 % dan tingkat error 23.01%. Dari tiga kali percobaan dihasilkan atribut *Acites* dan *Spider* merupakan atribut yang berpengaruh terhadap keputusan hidup atau meninggalnya pasien yang terkena penyakit hepatitis yang diikuti dengan atribut *varices, malaise, spider, albumin, anorexia, age, alk_phospate.*Dengan menggunakan Algoritma Neural Network didapat model klasifikasi dengan tingkat akurasi yang terbaik pada percobaan 1yaitu 82,97% dengan tingkat error 17.03%. Atribut yang paling berpengaruh dari masing-masing percobaan yaitu *anorexia, spiders* dan *protime*. Dengan menggunakan algoritma *K-Nearest Neighbor* didapat model klasifikasi dengan tingkat akurasi terbaik pada percobaan 3 yaitu 93% dan tingkat error 7%. Atribut yang paling berpengaruh terhadap penderita penyakit hepatitis yaitu *Albumin*.

Saran yang diberikan untuk penelitian lebih lanjut yaitu sebagai berikut dataset yang digunakan dalam penelitian sebaiknya memiliki jumlah dataset yang lebih banyak agar hasil yang diperoleh lebih maksimal.

DAFTAR PUSTAKA

[1] RI, K. K. (2014) ‘InfoDATIN: Situasi dan Analisi Hepatitis’, *Pusat Data dan Informasi*, p. 8. doi: 24427659.

[2] Lestari, M. E. I. (2014) ‘Penerapan Algoritma Klasifikasi Nearest Neighnor (K-NN) Untu Mendeteksi Penyakit Jantung’, Factor Exacta, 7, pp. 366–371.

[3] Krisandi, N., Helmi. and Prihandono, B. (2013) ‘Algoritma K-Nearest Neighbor dalam Klasifikasi Data Hasil Produktsi Kelapa Sawit PT. MINAMAS Kecamatan Paridu’, Teknologi Informasi & pendidikan, Vol. 2, no.1, pp. 34-35.

[4] Merluarini, B., Safitri, D. and Hoyyi, A. (2014) 'Perbandingan Analisis Klasifikasi Menggunakan Metode K-Nearest Neighbor (K-NN) dan Multivariate Adaptive Regression Spline (Mars) pada data Akreditasi Sekolah Dasar Negeri di Kota Semarang', Jurnal Gaussian, Vol. 3, no. 3, pp. 314-317.

[5] Leidiyana H. E. N. N. Y. (2013). 'Penerapan Algoritma K-Nearest Neighbor Untuk Penentuan Resiko Kredit Kepemilikan Kendaraan Bermotor', Jurnal Penelitian Ilmu Komputer, 1(1), 65-76.

[6] Septiani, W. D. (2017) Komparasi Metode Klasifikasi Data Mining Algoritma C4.5 Dan Naive Bayes Untuk Prediksi Penyakit Hepatitis, 13(1), pp. 76–84.

[7] Erawati, W. (2015) Prediksi Penyakit Hati Dengan Menggunakan Model Algoritma Neural Network, *Techno Nusa Mandiri*, XII(2), pp. 21–26.

[8] Shukla, A., Tiwari, R., & Kala, R. (2010) Real Life Applications of Soft Computing, Taylor and Francis Group, United States of America.