Jurnal Teknik Informatika (JUTIF)
Vol. 3, No. 1, Februari 2022, hlm. 133-140

DOI: https://doi.org/10.20884/1.jutif.2022.3.1.139
p-ISSN: 2723-3863
e-ISSN: 2723-3871

THE CONCEPT OF NAIVE BAYES AND ITS SIMPLE USE FOR PREDICTION FINAL SCORE OF STUDENT EXAMINATION USING R LANGUAGE

Aslan Alwi[*1], Munirah[2]

[1,2]Program Studi Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Ponorogo, Indonesia
Email: [1]elangbijak4@gmail.com, [2]munirah.mt@gmail.com

(Naskah masuk: 01 Februari 2022, Revisi: 12 Februari 2022, diterbitkan: 25 Februari 2022)

Abstract

In this paper, we try to explain how to formulate the derivation of the Naive Bayes concept and apply it to a simple case. This is because usually users only use existing formulas or tools that are already available in a programming language regardless of where the formulas are implemented in the available tools come from. To familiarize users with understanding the state of art rather than a formulation, in this study we try to combine the concept and application of the Naive Bayes model formulation. Starting with the elaboration of the concept of derivation of the Naive Bayes formula, then we take a case study to begin to provide an overview of the implementation of the formula. In this study, we apply Naive Bayes to predict learning outcomes before ending at the end of the semester. The dataset was constructed using daily scores from student activity and quizzes. The calculation of this algorithm is enough to use the R language with case sampling in 4 classes of language theory and automata even semester 2017-2018 at the Department of Informatics Engineering, Faculty of Engineering, University of Muhammadiyah Ponorogo with a dataset size of 99 records (99 students) which are divided into 70 records for training data and the rest for test data. The final result is that the prediction accuracy is 78.6%, with the conclusion that the use of the Naive Bayes concept is good enough to be used to predict in helping decision making and evaluation.

Keywords: *Accuracy, Naive Bayes, Prediction, R Language, Scores.*

KONSEP NAIVE BAYES DAN PENGGUNAANNYA SECARA SEDERHANA UNTUK PREDIKSI NILAI AKHIR UJIAN MAHASISWA MENGGUNAKAN BAHASA R

Abstrak

Pada paper ini, kami berusaha menjelaskan bagaimana rumus penurunan konsep *naive bayes* dan menerapkannya pada kasus sederhana. Hal ini dikarenakan biasanya pengguna hanya menggunakan formula yang sudah ada atau tools-tools yang sudah tersedia di dalam sebuah bahasa pemrograman tanpa menghiraukan dari mana perolehan rumus yang diimplementasikan dalam tools yang tersedia tersebut. Untuk membiasakan agar para pengguna memahami *state of art* dari pada sebuah perumusan, maka dalam penelitian ini kami mencoba untuk mengkombinasikan antara konsep dan penerapan terhadap model perumusan naive bayes. Diawali dengan penjabaran daripada konsep penurunan rumus naive bayes, lalu kami mengambil studi kasus untuk mulai memberikan gambaran terhadap implementasi daripada rumus tersebut. Pada penelitian ini, kami menerapkan *naive bayes* untuk memprediksi hasil pembelajaran sebelum berakhir di penghujung semester. Dataset dibangun menggunakan nilai-nilai harian dari keaktifan dan quis mahasiswa. Perhitungan algoritma ini cukup menggunakan bahasa R dengan pengambilan sampel kasus di 4 kelas mata kuliah teori bahasa dan automata semester genap tahun 2017-2018 di Jurusan Teknik Informatika, Fakultas Teknik, Universitas Muhammadiyah Ponorogo dengan ukuran dataset sebanyak 99 rekord (99 mahasiswa) yang terbagi atas 70 rekord untuk data pelatihan dan sisanya untuk data pengujian. Hasil akhir diperoleh ketepatan prediksi sebanyak 78,6%, dengan kesimpulan bahwa penggunaan konsep naive bayes cukup baik untuk digunakan memprediksi dalam membantu pengambilan keputusan dan evaluasi.

Kata kunci: *Akurasi, Bahasa R, Naive Bayes, Nilai, Prediksi.*

1. PENDAHULUAN

Naive bayes biasanya digunakan orang untuk menyaring email apakah berisi spam atau bukan.

Penggunaan naive bayes dalam beberapa penelitian di bidang analisis sentimen dilakukan oleh Das dan Kolya [1], Atika dan Suhadi [2], Kusuma dan Nugroho [3], serta Kosasih dan Alberto [4]. Pengunaan lain naive bayes di bidang klasifikasi antara lain penelitian yang dilakukan oleh Hsu dkk [5], Nirmala dkk [6], Basuki dkk [7], Putro dkk [8], Bakhtiar [9], Nugroho dkk [10], Muhathir dkk [11] dan Dikananda dkk [12]. Selain itu, naive bayes juga digunakan pada penelitian di bidang prediksi antara matakuliah dan dosen pengampu oleh Munirah dan Desriyanti [13] dan juga di bidang optimasi yang dilakukan oleh Romli dkk [14] serta Tempola dkk [15].

2. METODE PENELITIAN

Bayes classifier adalah sebuah cara melakukan klasifikasi dengan menggunakan teorema bayes. Teorema bayes yang memiliki bentuk dasar sebagai berikut sebagaimana dinyatakan oleh rumus (1).

$$P(Y|X) = \frac{P(X|Y).P(Y)}{P(X)} \quad (1)$$

Probabilitas P($Y|X$) adalah probabilitas posterior, P($X|Y$) adalah probabilitas *conditional*, P(Y) adalah probabilitas prior dan P(X) probabiltas *evidence*. Rumus (1) dapat ditulis lebih luas dan spesifik dengan memperluas variabel X menjadi $X_1,X_2,X_3,...,X_n$, dimana setiap variabel ke-i, X_i adalah variabel *explanatory* atau variabel yang hendak digunakan untuk melakukan klasifikasi atau prediksi. Adapun variabel Y disebut sebagai variabel kelas atau variabel target karena nilai variabel Y menjadi kelas-kelas yang menggolongkan dataset.

Secara umum rumus (1) dapat ditulis dalam bentuk $X_1,X_2,X_3,...,X_n$, dan Y sebagai berikut:

$$P(Y|X1, X2, X3, \dots, Xn) = \frac{P(X1,X2,X3,\dots,Xn|\,Y).P(Y)}{P(X1,X2,X3,\dots,Xn)} \quad (2)$$

Dengan mengambil nilai probabilitas maksimum dari posterior maka klasifikasi dilakukan. Misal sebuah rekord dengan nilai-nilai atribut ($X_1,X_2,X_3,...,X_n$) maka rekord tersebut dapat diklasifikasikan ke dalam nilai Y yang memiliki probabilitas posterior yang maksimum. Yaitu P(Y | $X_1,X_2,X_3,...,X_n$) maksimum. Proses klasifikasi ini dapat ditulis dalam rumus berikut:

$$argmax\ P(Y|X1, X2, X3, \dots, Xn) = argmax \frac{P(X1,X2,X3,\dots,Xn|Y).P(Y)}{P(X1,X2,X3,\dots,Xn)} \quad (3)$$

Akan tetapi untuk menghitung probabilitas P($Y|X_1,X_2,X_3,...,X_n$) diperlukan banyak kombinasi hitungan untuk P($X_1,X_2,X_3,...,X_n|Y$). Secara umum, jumlah kemungkinan probabilitas untuk menghitung P($X_1,X_2,X_3,...,X_n|Y$) jika nilai setiap variabel hanya ada dua kemungkinan nilai yaitu $2.2^{(n-1)}$. Jika terdapat k kemungkinan nilai maka jumlah kemungkinan komputasi adalah $k.k^{(n-1)}$. Tetapi jika mengunakan assumsi bahwa setiap variabel di dalam $X_1,X_2,X_3,...,X_n$, Y adalah independen satu sama lain (hubungan yang independen atau hubungan yang naif, sehingga disebut *naive bayes*), maka jumlah kemungkinan komputasi untuk menghitung probabilitas P($X_1,X_2,X_3,...,X_n|Y$) adalah $2n$ saja. Jika terdapat k kemungkinan nilai bagi setiap X_i maka jumlah kombinasi kemungkinan komputasinya adalah kn.

Berikut ini adalah sebuah argumen mengapa jumlah komputasinya demikian. Misalkan untuk setiap X_i memiliki kemungkinan nilai u_i dan *non*-u_i (termasuk Y yaitu u_y dan *non*-u_y) yaitu terdapat dua kemungkinan nilai saja untuk setiap variabel. Jika demikian maka banyaknya kombinasi P($X_1,X_2,X_3,...,X_n|Y$) adalah banyaknya susunan kombinasi biner (biner karena 2 kemungkinan yaitu u_i dan *non*-u_i) dari $X_1,X_2,X_3,...,X_n$ dalam n posisi ditambah dengan kemungkinan nilai Y secara umum keseluruhan dalam n+1 posisi yang mungkin, yaitu 2^{n+1} atau 2.2^n.

Akan tetapi kita hanya tertarik untuk menghitung probabilitas P($X_1,X_2,X_3,...,X_n|Y$) untuk suatu nilai Y tertentu (Y tetap), sehingga kemungkinan nilai Y pada saat itu hanyalah satu. Ini berarti jumlah kemungkinan komputasi menjadi 2.2^{n-1} saja.

Secara umum jika terdapat k_i jumlah kemungkinan nilai untuk setiap X_i dan k_{max} adalah suatu jumlah maksimum diantara k_i maka maksimum jumlah komputasi $k_{max}.k_{max}^{n-1}$. Selanjutnya berdasarkan prinsip independen x independen y jika hanya jika P($x|y$)=P(x), diperoleh:

$$P(X_1,X_2,X_3,...,X_n|Y) = P(X_1).P(X_2).P(X_3)...P(X_{n-1}).P(X_n) \quad (4)$$

Misalkan bahwa untuk setiap variabel terdapat hanya 2 kemungkinan nilai yaitu u_i dan *non*-u_i maka jumlah kemungkinan untuk setiap hitungan P(X_i) adalah 2 kemungkinan yaitu P(X_i=u_i) dan P(X_i=*non*-u_i). Karena ada n buah perkalian kemungkinan yaitu P(X_1)P(X_2)P(X_3)...P(X_{n-1})P(X_n), maka jumlah seluruh kemungkinan komputasi adalah $2n$. Secara umum jika terdapat k_i kemungkinan nilai untuk setiap X_i dan k_{max} adalah yang terbesar maka jumlah maksimum seluruh kemungkinan komputasi adalah $2k_{max}$. Dengan demikian diperoleh bentuk asumsi *naive bayes* sebagai berikut:

$$\text{argmax}\ \mathrm{P}(Y|X1, X2, X3, \dots, Xn) = \frac{\text{argmax}\ \mathrm{P}(X1)\mathrm{P}(X2)\mathrm{P}(X3)\dots\mathrm{P}(Xn-1)\mathrm{P}(Xn).\mathrm{P}(Y)}{\mathrm{P}(X1,X2,X3,\dots,Xn)} \quad (5)$$

Persamaan (4) menjadi lebih mudah dihitung karena hanya memiliki $2n$ jumlah komputasi probabilititas dibanding persamaan (2) dengan jumlah komputasi sebanyak 2.2^{n-1} kemungkinan komputasi.

Akan tetapi kita dapat membuat sebuah asumsi *naive* baru yaitu bahwa walaupun $X_1,X_2,X_3,...,X_n$ independen satu sama lain, tetapi $X_1,X_2,X_3,...,X_n$ adalah tidak independen terhadap Y. Ini berarti

$P(X_1,X_2,X_3,...,X_n|Y) \neq P(X_1,X_2,X_3,...,X_n)$. Asumsi ini tertulis sebagai berikut:

$$P(X_1,X_2,X_3,...,X_n|Y) = P(X_1|Y)P(X_2|Y)P(X_3|Y)...P(X_{n-1}|Y)P(X_n|Y) \quad (6)$$

Untuk memahami bagaimana asumsi ini diturunkan adalah sebagai berikut:

Dari asumsi independen, yaitu bahwa X independen terhadap Z jika hanya jika $P(X|Z)=P(X)$. Ini berarti dari definisi *conditional probability* diperoleh:

$$P(X,Z)=P(X|Z).P(Z)=P(Z|X).P(X)=P(X).P(Z) \quad (7)$$

Sehingga asumsi independen dapat diperluas menjadi X dan Z saling independen jika hanya jika $P(X,Z)=P(X).P(Z)$. Tetapi jika kita membawa ke dalam cara pandang himpunan, maka dapat ditulis sebagai $P(X,Z)=P(X\cap Z)$.

Tetapi $P(X\cap Z)=P(X\cap Z\cap U)$, U adalah himpunan universal dimana $P(U)=1$ dan $P(X\cap Z\cap U)= P(X,Z,U)=P(X,Z|U).P(U)$ berdasarkan definisi *conditionl probability*. Diperoleh:

$$P(X,Z)=P(X\cap Z)=P(X\cap Z\cap U)=P(X,Z,U)=P(X,Z|U).P(U)=P(X,Z|U) \quad (8)$$

Tetapi kemudian $P(X,Z)=P(X,Z,U)=P(X\cap Z\cap U)$ $=P(X\cap U\cap Z\cap U)=P((X,U), (Z,U))$ karena X, Z independen maka $X\cap U$ independen $Z\cap U$ dan juga (X,U) independen (Z,U). Tetapi berdasarkan asumsi independen $P((X,U), (Z,U))=P(X,U).P(Z,U)$ dan $P(X,U)=P(X|U).P(U)=P(X|U)$ dan $P(Z,U)=P(Z|U).P(U)=P(Z|U)$.

Berdasarkan definisi *conditional probability* dan $P(U)=1$ dan (9) diperoleh $P(X,Z|U)=P(X|U).P(Z|U)$. Persamaan ini dapat diperluas untuk setiap $Y\subseteq U$. Berdasarkan pandangan himpunan, setiap himpunan X_i yang ada di dalam Y dapat melihat secara relatif bahwa Y sebagai himpunan universalnya. Secara umum yaitu bahwa setiap irisan X_i yang ada di dalam Y dapat melihat bahwa Y adalah himpunan universalnya, atau bahwa setiap himpunan $X_i\cap Y$ dapat melihat Y secara relatif sebagai himpunan universalnya. Sehingga secara relatif, kita dapat menetapkan $P(Y)=1$ (untuk semua nilai Y) bagi semua probabilitas relatif X_i terhadap Y. Karena itu, persamaan dapat ditulis ulang di dalam cara pandang probabilitas relatif sebagai:

$$P(X_1,X_2|Y)=P(X_1|Y).P(X_2|Y) \quad (9)$$

Selanjutnya jika $X_1,X_2,X_3,...,X_n$ saling independen tetapi tidak terhadap Y, maka dengan menggunakan induksi, secara umum. dapat diperluas menjadi:

$$P(X_1,X_2,X_3,...,X_n|Y)= \prod P(X_i|Y),\ i=1,2,3,...,\ n \quad (10)$$

Selanjutnya, persamaan *Bayes classifier* dengan menggunakan asumsi *naive bayes* pada persamaan (4) dapat ditulis menjadi:

$$argmax\ P(Y|X1,X2,X3,\ldots,Xn) = argmax \frac{P(Y).\prod P(Xi|Y)}{P(X1,X2,X3,\ldots,Xn)} \quad (11)$$

Dengan hitungan yang sama seperti pembahasan di atas sebelumnya, jumlah kemungkinan komputasi bagi (12) adalah $2n$ atau $2k_{max}$.

3. HASIL DAN PEMBAHASAN

Untuk melihat bagaimana cara menghitung probabilitas persamaan (12), pada bagian ini inShaa Allah akan didemonstrasikan sebuah contoh yang diharapkan dapat menunjukkan dengan jelas cara perhitungan probabilitas (12). Misalkan terdapat sebuah dataset sebagai berikut:

Tabel 1. Contoh dataset

id	X_1	X_2	X_3	Y
1.	A	D	W	sukses
2.	B	D	G	gagal
3.	B	D	G	sukses
4.	B	C	W	gagal
5.	A	C	G	gagal
6.	A	C	G	gagal
7.	A	C	W	sukses

Hitung argmax $P(Y|X_1,X_2,X_3)$ sebagai *max* {$P(Y=sukses\ |X_1,X_2,X_3)$, $P(Y=gagal\ |X_1,X_2,X_3)$ }. Yaitu:

argmax $P(Y|X_1,X_2,X_3)$ = *max* {$P(Y=sukses\ |X_1,X_2,X_3)$, $P(Y=gagal\ |X_1,X_2,X_3)$ }

Kemudian hitung satu persatu $P(Y=sukses\ |X_1,X_2,X_3)$ dan $P(Y=gagal\ |X_1,X_2,X_3)$. Yaitu dihitung berdasarkan persamaan (12) sehingga menjadi:

$P(Y=sukses|X_1,X_2,X_3)=P(X_1|Y=sukses).P(X_2|Y=sukses).P(X_3|Y=sukses).P(Y=sukses)/P(X_1,X_2,X_3)$ dan $P(Y=gagal|X_1,X_2,X_3)=P(X_1|Y=gagal).P(X_2|Y=gagal).P(X_3|Y=gagal).P(Y=gagal)/P(X_1,X_2,X_3)$

Selanjutnya hitung satu persatu kemungkinan probabilitas untuk setiap nilai X_i yaitu sebagai berikut:

$P(X_1=A|Y=sukses)$ = $P(X_1=A,Y=sukses)/P(Y=sukses)$

$P(X_1=A,Y=sukses)$ = 2/7 = jumlah rekord dimana X_1=A dan $Y=sukses$ dibagi total rekod

$P(Y=sukses)$ = 3/7 = jumlah rekord yang memiliki kolom $Y=sukses$ dibagi total rekod

Sehingga:

$P(X_1=A|Y=sukses)$ = $P(X_1=A,Y=sukses)/P(Y=sukses)=2/3$

Selanjutnya dengan cara yang sama:

$P(X_1=B|Y=sukses)$ = $P(X_1=B,Y=sukses)/P(Y=sukses)=1/3$

dan

$P(X_2=D|Y=sukses)$ = $P(X_2=D,Y=sukses)/P(Y=sukses)=2/3$

$P(X_2=C|Y=sukses)$ = $P(X_2=C,Y=sukses)/P(Y=sukses)=1/3$

$P(X_3=W|Y=sukses)$ = $P(X_3=W,Y=sukses)/P(Y=sukses)=2/3$

P(X_3=G|Y=*sukses*) = P(X_3=G,Y=*sukses*)/P(Y=*sukses*)=1/3

Sehingga misalkan untuk kemungkinan kombinasi X_1=A, X_2=C, X_3=W dan Y=*sukses.*

P(X_1=A, X_2=C, X_3=W) = 1/7 = jumlah rekord yang memiliki kolom X_1=A, X_2=C, X_3=W dbagi jumlah rekord keseluruhan.

Juga P(Y=*sukses*)= 3/7 = jumlah rekord dengan kolom Y = *sukses* dibagi jumlah kolom yang memuat nilai *sukses* atau *gagal*.

Dengan demikian, untuk kemungkinan kombinasi X_1=A, X_2=C, X_3=W, maka persamaan 4.12 dapat dihitung sebagai berikut:

P(Y=*sukses*|X_1=A,X_2=C,X_3=W)=P(X_1=A|Y=*sukses*). P(X_2=C|Y=*sukses*).P(X_3=W|Y=*sukses*).P(Y=*sukses*)/ P(X_1=A,X_2=C,X_3=W)

=(2/3.1/3.2/3.3/7)/(1/7)

= 4/9

Selanjutnya hitung seluruh kemungkinan kombinasi, lalu pilih nilai probabilitas yang paling tinggi di antara seluruh kemungkinan itu sebagai nilai *argmax* yang diinginkan.

	A	C	D	E	F	G	H	I	J	K	L	M	N	O
1	NIM	quis1	quis2	quis3	quis4	quis5	quis6	quis7	quis8	Jumlah quis	Total quis	Absen	UTS	UAS
2	16532695	100	0	0	0	0	0	0	0	10	80	30	90	100
3	16532566	100	80	100	90	0	0	0	0	40	80	90	70	90
4	16532565	100	100	95	0	0	0	0	0	30	80	90	90	80
5	16532564	100	100	85	100	0	0	0	0	40	80	90	90	100
6	16532563	80	100	95	0	0	0	0	0	30	80	60	80	100
7	16532562	100	100	100	0	0	0	0	0	30	80	90	90	100
8	16532561	100	95	75	100	80	100	0	0	60	80	90	90	100
9	16532560	100	100	90	0	0	0	0	0	30	80	60	70	100
10	16532559	100	100	70	100	75	80	90	100	80	80	90	80	90
11	16532558	100	100	95	0	0	0	0	0	30	80	60	90	100
12	16532557	100	100	100	0	0	0	0	0	30	80	90	80	100
13	16532556	100	0	0	0	0	0	0	0	10	80	90	80	100
14	16532555	100	70	100	100	100	0	0	0	50	80	60	90	90
15	16532554	100	70	0	0	0	0	0	0	20	80	30	90	90
16	16532553	100	100	100	0	0	0	0	0	30	80	60	90	100
17	16532552	100	100	100	0	0	0	0	0	30	80	90	70	100
18	16532551	100	100	100	90	100	0	0	0	50	80	90	70	100
19	16532549	100	95	100	90	80	0	0	0	50	80	90	90	100
20	16532548	100	100	100	100	0	0	0	0	40	80	90	90	100
21	16532547	80	100	90	0	0	0	0	0	30	80	60	90	90
22	16532545	100	100	90	100	0	0	0	0	40	80	90	100	90
23	16532544	100	0	0	0	0	0	0	0	10	80	30	0	0
24	16532542	100	100	100	90	0	0	0	0	40	80	60	90	90
25	16532540	100	70	100	100	0	0	0	0	40	80	90	100	100
26	16532539	100	100	85	0	0	0	0	0	30	80	60	100	90
27	16532538	100	100	100	0	0	0	0	0	30	80	90	100	90

KELAS KHUSUS | KELAS B 2016 | KELAS C 2016 | KELAS D 2016, A 2017 | KELAS A 20

Gambar 1. Dataset yang digunakan

3.1. Konstruksi Data Set

Berikut ini adalah konstruksi dataset yang ingin digunakan untuk menerapkan *bayes classifier* dengan menggunakan perhitungan *naive bayes*. Data yang dikumpulkan adalah data yang diperoleh dari hasil perkuliahan pada semester genap tahun 2017-2018 dengan jumlah rekord sebanyak 99 buah dari 4 kelompok belajar, dapat dilihat pada gambar 1 diatas.

Rincian metadata dari dataset mahasiswa yang dihilangkan sebagaian pada tabel 1 dapat dilihat pada tabel 2.

Berikut ini adalah sebagian potongan dataset yang ingin digunakan di dalam permodelan *naive bayes*.

Tabel 2. Metadata dataset mahasiswa

Id	Kolom	Tipe	Keterangan
1.	NIM	Text	Nomor induk mahasiswa
2.	quis1	Numerik	Nilai quis
3.	quis2	Numerik	Nilai quis
4.	quis3	Numerik	Nilai quis
5.	quis4	Numerik	Nilai quis
6.	quis5	Numerik	Nilai quis
7.	quis6	Numerik	Nilai quis
8.	quis7	Numerik	Nilai quis
9.	quis8	Numerik	Nilai quis
10.	jumlah_quis	Numerik	Jumlah quis yang diikuti
11.	total_quis	Numerik	Total jumlah quis seluruh kelas
12.	Absen	Numerik	Prosentase kehadiran
13.	UTS	Numerik	Nilai ujian tengah semester
14.	UAS	Numerik	Nilai ujian akhir semester

Di dalam model data ini, ditetapan UAS sebagai variabel kelas yaitu variabel *y* dalam model (12). Model (12) digunakan untuk memprediksi nilai UAS mahasiswa berdasarkan nilai-nilai keaktifan di kelas (nilai-nilai quis).

Selanjutnya model *naive bayes* dapat kita tulis sebagai berikut:

argmax

P(*UAS*|*quis*$_1$,*quis*$_2$,*quis*$_3$,...,*quis*$_8$,*Jumlah_quis*,*Total_quis*, *Absen*,*UTS*) = *argmax*

P(quis1|UAS).P(quis2|UAS).P(quis3|UAS)...,

P(quis8|UAS).P(Jumlah_quis|UAS).P(Total_quis|UAS).P(Absen|UAS).P(UTS|UAS).P(UAS)/P(quis1,quis2,quis3,...,quis8,

Jumlah_quis,Total_quis,Absen,UTS) (12)

Model ini dapat dihitung pada lembar *spreadsheet* seperti lembar *excel* akan tetapi itu membutuhkan banyaknya ketelitian untuk menghitung semua kombinasi probabilitias. Karena itu, pada penelitian ini digunakan pustaka *bnlearn* pada bahasa R yang memiliki algoritma *naive bayes* untuk menghitung secara otomatis persamaan model (12).

3.2. Perhitungan Naive Bayes Menggunakan Bahasa R

Untuk memulai menghitung model (12) menggunakan bahasa R, terlebih dulu memanggil pustaka bnlearn pada konsol bahasa R dengan perintah:

```
>library(bnlearn,quietly=TRUE,verbose=FALSE,warn.conflicts = FALSE)
```

Selanjutnya impor dataset pada gambar 1 ke dalam Rstudio dengan menambahkan kolom Nama yaitu sebagai berikut.

```
>dataframe_nb=read.csv("C:\\Users\\
USER\\Documents\\Dataset nilai
TBO.csv")
```

Selanjutnya ambil kolom-kolom data yang penting saja yang hendak dihitung sesuai model pada rumus (13). Yaitu dengan perintah sebagai berikut:

```
>dataframe_nb2<-
dataframe_nb[,c("quis1","quis2","qu
is3","quis4","quis5","quis6","quis7
","quis8","jumlah_quis","total_quis
","Absen","UTS","UAS")]
```

Karena pustaka bnlearn bekerja dengan data tipe factor pada model naive bayes, dilakukan terlebih dulu penerjemahan kolom-kolom dataset ke dalam tipe factor.

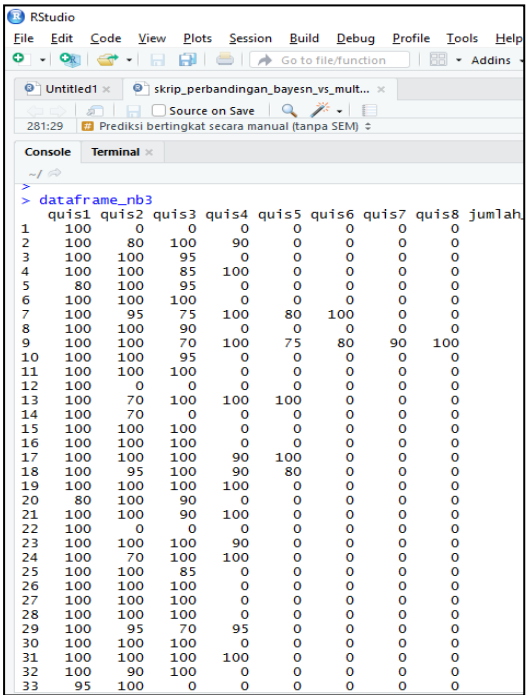
```
> quis1 = factor(dataframe
nb2[,"quis1"])

> quis2 = factor(dataframe
nb2[,"quis2"])

> quis3 = factor(dataframe
nb2[,"quis3"])

> quis4 = factor(dataframe
nb2[,"quis4"])

> quis5 = factor(dataframe
nb2[,"quis5"])

> quis6 = factor(dataframe
nb2[,"quis6"])

> quis7 = factor(dataframe
nb2[,"quis7"])

> quis8 = factor(dataframe
nb2[,"quis8"])

> jumlah_quis = factor(dataframe
nb2[,"jumlah quis"])

> total_quis = factor(dataframe
nb2[,"total quis"])

> Absen = factor(dataframe
nb2[,"Absen"])

> UTS=factor(dataframe_nb2[,"UTS"])

> UAS=factor(dataframe_nb2[,"UAS"])
```

Kemudian semua kolom disatukan kembali membentuk sebuah dataframe dalam bahasa R sebagai berikut:

```
> dataframe_nb3<-
data.frame(quis1,quis2,quis3,quis4,
quis5,quis6,quis7,quis8,jumlah_quis
,total_quis,Absen,UTS,UAS)
```

Hasilnya dapat ditampilkan dengan menggunakan perintah sebagai berikut:

```
> dataframe_nb3
```



Gambar 2. Dataset mahasiswa di dalam R

Kemudian pembuatan model naive bayes sebagaimana persamaan (13) dapat dilakukan dengan dengan menggunakan fungsi naive.bayes yang ada pada pustaka bnlearn. Penggunaan fungsi ini adalah sebagai berikut:

```
> bn = naive.bayes(dataframe_nb3,
"UAS")
```

Model *naive bayes* (13) yaitu bn jika dinyatakan ke dalam model DAG maka dia dapat dinyatakan ke dalam ekspresi:

```
[UAS][quis1|UAS][quis2|UAS][quis3|U
AS][quis4|UAS][quis5|UAS][quis6|UAS
][quis7|UAS][quis8|UAS][jumlah_quis
|UAS][total_quis|UAS][Absen|UAS][UT
S|UAS]
```

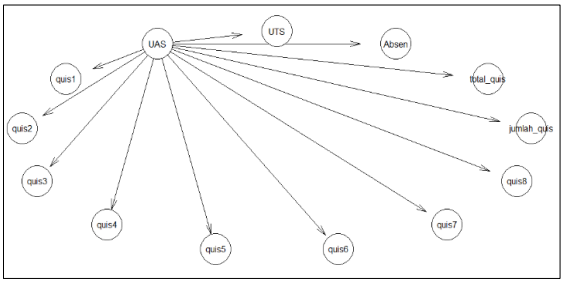
Ini sesuai dengan asumsi *naive* bayes yang dikenakan kepada model (12) yaitu:

$$P(UAS).P(quis_1|UAS).P(quis_2|UAS).P(quis_3|UAS).P(quis_4|UAS).P(quis_5|UAS).P(quis_6|UAS).P(quis_7|UAS).P(quis_8|UAS).P(jumlah\_quis|UAS).P(total\_quis|UAS).$$

P(*Absen*|*UAS*).P(*UTS*|*UAS*)

Kemudian dengan melakukan perintah plot, bentuk graph dari model naive bayes dapat ditunjukkan sebagai berikut:

```
> plot(bn)
```



Gambar 3. Bentuk DAG dari model

Untuk melakukan prediksi menggunakan *naive bayes* terhadap dataset, bagi dua dataset sebagai dataset *learning* dan dataset *test*. Untuk melakukan ini dilakukan perintah pembagian dataset sebelumnya sebagai berikut:

```
> dataframe_nb4 = dataframe
nb2[1:70,]

> dataframe_nb5 = dataframe
nb2[71:98,]
```

Dimana dataframe_nb4 adalah dataset *learning* dan dataset dataframe_nb5 adalah dataset *test*. Akan tetapi fungsi *naive.bayes*() pada pustaka bnlearn hanya menerima tipe data factor sehingga terlebih dulu adalah perlu untuk menerjemahkan dataframe_nb4 dan dataframe_nb5 ke dalam tipe data factor. Penerjemahan dataframe_nb4 menjadi dataframe_nb6 yang bertipe factor:

```
> quis1 = factor(dataframe
nb4[,"quis1"])

> quis2 = factor(dataframe
nb4[,"quis2"])

> quis3 = factor(dataframe
nb4[,"quis3"])

> quis4 = factor(dataframe
nb4[,"quis4"])

> quis5 = factor(dataframe
nb4[,"quis5"])

> quis6 = factor(dataframe
nb4[,"quis6"])

> quis7 = factor(dataframe
nb4[,"quis7"])

> quis8 = factor(dataframe
nb4[,"quis8"])

> jumlah_quis = factor(dataframe
nb4[,"jumlah quis"])

> total_quis = factor(dataframe
nb4[,"total quis"])

> Absen = factor(dataframe
nb4[,"Absen"])

> UTS=factor(dataframe_nb4[,"UTS"])

> UAS=factor(dataframe_nb4[,"UAS"])

> dataframe_nb6<-data.frame(quis1,
quis2,quis3,quis4,quis5,quis6,quis7
, quis8, jumlah_quis,total_quis,
Absen,UTS,UAS)
```

Juga untuk dataframe_nb5 menjadi dataframe_nb7:

```
> quis1 = factor(dataframe
nb5[,"quis1"])

> quis2 = factor(dataframe
nb5[,"quis2"])

> quis3 = factor(dataframe
nb5[,"quis3"])

> quis4 = factor(dataframe
nb5[,"quis4"])

> quis5 = factor(dataframe
nb5[,"quis5"])

> quis6 = factor(dataframe
nb5[,"quis6"])

> quis7 = factor(dataframe
nb5[,"quis7"])

> quis8 = factor(dataframe
nb5[,"quis8"])

> jumlah_quis = factor(dataframe
nb5[,"jumlah_quis"])

> total_quis = factor(dataframe
nb5[,"total_quis"])

> Absen = factor(dataframe
nb5[,"Absen"])

> UTS=factor(dataframe_nb5[,"UTS"])

> UAS=factor(dataframe_nb5[,"UAS"])
```

```
> dataframe_nb7<-data.frame(quis1,
quis2,quis3,quis4,quis5,quis6,quis7
,quis8,jumlah_quis,total_quis,Absen
,UTS,UAS)
```

Selanjutnya proses pembelajaran dilakukan untuk model *naive bayes* dengan menggunakan dataframe_nb6 sebagai berikut:

```
> bn = naive.bayes(dataframe_nb6,
"UAS")
```

Kemudian dilakukan perhitungan probabilitas bagi model (12) sebagai berikut:

```
> fitted = bn.fit(bn,dataframe nb6)
> fitted
```

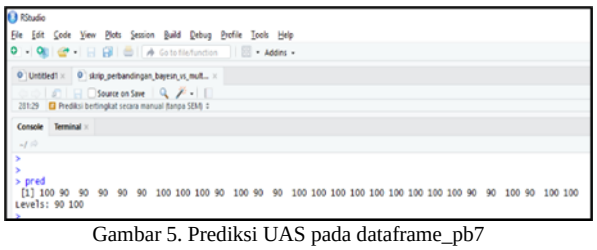
Diperoleh hasil perhitungan probabilitas *naive bayes* sebagai berikut:

```
> fitted = bn.fit(bn, dataframe_nb6)
> fitted

  Bayesian network parameters

  Parameters of node UAS (multinomial distribution)

Conditional probability table:
          0         50         70         80         90        100
0.01428571 0.01428571 0.05714286 0.07142857 0.40000000 0.44285714

  Parameters of node quis1 (multinomial distribution)

Conditional probability table:

     UAS
quis1          0         50         70         80         90        100
  0   0.00000000 0.00000000 0.25000000 0.40000000 0.03571429 0.00000000
  65  0.00000000 0.00000000 0.25000000 0.20000000 0.03571429 0.03225806
  70  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.03225806
  80  0.00000000 0.00000000 0.00000000 0.00000000 0.14285714 0.09677419
  85  0.00000000 0.00000000 0.00000000 0.00000000 0.07142857 0.00000000
  90  0.00000000 1.00000000 0.00000000 0.00000000 0.03571429 0.03225806
  95  0.00000000 0.00000000 0.25000000 0.00000000 0.00000000 0.00000000
  100 1.00000000 0.00000000 0.25000000 0.40000000 0.67857143 0.80645161

  Parameters of node quis2 (multinomial distribution)

Conditional probability table:

     UAS
quis2          0         50         70         80         90        100
  0   1.00000000 0.00000000 0.50000000 0.60000000 0.07142857 0.09677419
  60  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.03225806
  65  0.00000000 0.00000000 0.00000000 0.00000000 0.00000000 0.03225806
```

Gambar 4. Hasil Perhitungan *conditional probability*

Setelah model *naive bayes* telah diperoleh yaitu bn maka dengan menggunakan bn, dapat dilakukan prediksi naive bayes terhadap dataset test dataframe_nb7. Proses prediksi dilakukan terhadap kolom UAS dari dataframe_nb7. Prediksi dilakukan dengan perintah:

```
> pred = predict(bn, dataframe_nb7)
```



Gambar 5. Prediksi UAS pada dataframe_pb7

Hasil prediksi dapat dilihat dengan mengetik perintah sebagaimana pada gambar 5. Untuk menunjukkan dengan lebih jelas perbandingan itu, dapat dilihat dengan perintah sebagai berikut:

```
> dataframe_nb8 =
data.frame(dataframe nb7[,"UAS"],
pred)

> dataframe_nb8
```

Untuk melakukan analisis yang lebih rinci, dapt dibuat perintah sebagai berikut:

```
> table(pred, dataframe_nb7[,
"UAS"])
```

```
> table(pred, dataframe_nb7[, "UAS"])

pred  90 100
  90   7   4
  100  2  15
```

Gambar 6. Perbandingan Frekuensi

Gambar 6 menjelaskan bahwa prediksi nilai 90 menghasilkan sama dengan nilai asli 90 adalah sejumlah 6 prediksi. Sedang prediksi salah dimana nilai prediksi 90 tetapi nilai asli 100 adalah sebanyak 2 prediksi. Demikian juga jumlah prediksi benar dimana nilai prediksi 100 dan nilai asli 100 adalah berjumlah 15 prediksi benar. Jumlah prediksi salah dimana nilai prediksi 100 tetapi nilai asli 90 adalah 4 prediksi.

Dengan demikian, jumlah prediksi benar adalah 22 prediksi dan jumlah prediksi salah adalah 6 prediksi. Sehingga akurasi model adalah:

Akurasi model naive bayes = 22 x 100%/(22+6) = 78,6%

4. KESIMPULAN

Secara keseluruhan, hasil paper yang ditulis ini menyajikan tentang bagaimana memahami konsep penurunan naïve bayes secara matematis, kemudian perhitungannya didemonstrasikan ke dalam bahasa pemrograman R yang khusus untuk mengolah data sesuai dengan kasus yang dipilih sampai pada hasil akhir daripada perhitungan beserta *plot graph* DAG. Dalam hal ini, paper ini ingin memberikan semacam tutorial singkat terkait konsep dan model penggunaan.

Sementara untuk demonstrasi yang dilakukan dalam penelitian ini menggunakan jumlah data 99 rekord (terbagi atas 70 rekord untuk training dan sisanya digunakan untuk melakukan pengujian prediksi) diperoleh akurasi prediksi dari model yang dibuat mencapai 78,6%.

Akurasi model ini cukup layak untuk digunakan sebagai *tool* bagi dosen untuk memprediksi hasil pembelajaran mata kuliah di masa depan. Selain itu, penggunaan *naive bayes* ini juga berguna untuk digunakan oleh dosen wali untuk memprediksi kemampuan mahasiswa pada semester berikut dengan melihat nilai-nilai yang telah diambil sebelumnya. Ini diharapkan dapat memudahkan dosen untuk melakukan antisipasi dan memberikan nasehat kepada mahasiswa.

Sebagai catatan, untuk menambah akurasi prediksi model dapat dilakukan dengan menambah dataset untuk training

DAFTAR PUSTAKA

[1] S. Das dan A. K. Kolya, "Sense GST: Text mining & sentiment analysis of GST tweets by Naive Bayes algorithm," dalam International Conference on Research in Computational Intelligence and Communication Networks, ICRCICN, 2017.

[2] P. D. Atika dan Suhadi, "Implementasi Algoritma Naïve Bayes Classifier untuk Analisis Sentimen Customer pada Toko Online," Factor Exacta, vol. 12, no. 4, pp. 303-314, 2020.

[3] A. Kusuma dan A. Nugroho, "Analisa Sentimen Pada Twitter Terhadap Kenaikan Tarif Dasar Listrik Dengan Metode Naïve Bayes," Jurnal Ilmiah Teknologi Informasi Asia, vol. 5, no. 2, pp. 137-146, 2021.

[4] R. Kosasih dan A. Alberto, "Sentiment analysis of game product on shopee using the TF-IDF method and naive bayes classifier," LKOM Jurnal Ilmiah, vol. 13, no. 2, pp. 101-109, 2021.

[5] C. S. Hsu, I. C. Chen dan C. L. Huang, "Image Classification Using Pairwise Local Observations Based Naive Bayes Classifier," dalam Proceedings of APSIPA Annual Summit and Conference, 2015.

[6] N. Venkateswaran, K. Nirmala dan V. K. C, "HoG Based Naive Bayes Classifier for Glaucoma Detection," dalam Proceedings of the 2017 IEEE Region 10 Conference (TENCON), Malaysia, 2017.

[7] S. Basuki, S. Maghfiroh dan Y. Azhar, "Klasifikasi Tweets Tindak Kejahatan Berbahasa Indonesia Menggunakan Naive Bayes Setio," Repositor, vol. 2, no. 7, 2020.

[8] H. F. Putro, R. T. Vulandari dan W. L. Saptomo, "Penerapan Metode Naive Bayes Untuk Klasifikasi Pelanggan Hakam," Jurnal Teknologi Informasi dan Komunikasi (TIKomSiN), vol. 8, no. 2, pp. 19-24, 2020.

[9] M. Y. Bakhtiar, "Klasifikasi Penelitian Dosen Menggunakan Naïve Bayes Classifier dan Algoritma Genetika," STRING (Satuan Tulisan Riset dan Inovasi Teknologi), vol. 5, no. 2, pp. 134-143, Desember 2020.

[10] K. S. Nugroho, I. dan F. Marisa, "Naive Bayes classifier optimization for text classification on e-government using particle swarm optimization," Jurnal Teknologi dan Sistem Komputer, vol. 8, no. 1, pp. 21-26, 2020.

[11] M. dan M. H. Santoso, "Analysis Naïve Bayes In Classifying Fruit by Utilizing Hog," Journal of Informatics and Telecommunication Engineering, vol. 4, no. 1, pp. 151-160, Juli 2020.

[12] A. R. Dikananda, I. Ali, F. A. R. Rinaldi dan I. , "Genre e-sport gaming tournament classification using machine learning technique based on decision tree, Naïve Bayes, and random forest algorithm," dalam Annual Conference on Computer Science and Engineering Technology (AC2SET), 2021.

[13] M. Munirah dan D. Desriyanti, "Prediction Of Compatibility Between Lecturers and The Subjects Using The Machine Learning with Naive Bayes Algorithm," International Journal of Scientific & Engineering Research, vol. 11, no. 1, pp. 695-698, Januari 2020.

[14] T. N. Wiyatno, I. Romli dan S. , "Naive Bayes Algorithm Implementation Based on Particle Swarm Optimization in Analyzing the Defect Product," dalam InCEESS 2020, Bekasi, 2021.

[15] F. Tempola dan A. Mubarak, "Optimization Naive Bayes using Particle Swarm Optimization in Volcanic Activities," dalam International Conference on Science and Technology, 2020.