PEMBOBOTAN FITUR DATASET MENGGUNAKAN *GAIN RATIO* GUNA MENINGKATKAN AKURASI METODE *NAÏVE BAYESIAN CLASSIFIER*

TESIS

NOVRIADI ANTONIUS SIAGIAN

187038008

PROGRAM STUDI S2 TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI

UNIVERSITAS SUMATERA UTARA

MEDAN

2020

PEMBOBOTAN FITUR DATASET MENGGUNAKAN *GAIN RATIO* GUNA MENINGKATKAN AKURASI METODE *NAÏVE BAYESIAN CLASSIFIER*

TESIS

Diajukan untuk melengkapi tugas dan memenuhi syarat memperoleh ijazah Magister Teknik Informatika

NOVRIADI ANTONIUS SIAGIAN

187038008

PROGRAM STUDI S2 TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER DAN TEKNOLOGI INFORMASI
UNIVERSITAS SUMATERA UTARA
MEDAN

2020

PERSETUJUAN

Judul	: Pembobotan Fitur Dataset Menggunakan Gain Ratio Guna Meningkatkan Akurasi Metode Naïve Bayesian Classifier
Kategori	: Ilmu Komputer
Nama	: Novriadi Antonius Siagian
Nomor Induk Mahasiswa	: 187038008
Program Studi	: S2 Teknik Informatika
Fakultas	: ILMU KOMPUTER DAN TEKNOLOGI INFORMASI UNIVERSITAS SUMATERA UTARA

Komisi Pembimbing :

Pembimbing 2,

Dr. Sawaluddin, M.IT
NIP. 19581203 199802 1001

Pembimbing 1,

Dr. Sutarman, M.Sc
NIP. 19631026 199103 1001

Diketahui/disetujui oleh
Program Studi S2 Teknik Informatika
Ketua,

Prof. Dr. Muhammad Zarlis
NIP. 19570701 198601 1 003

PERNYATAAN PERSETUJUAN PUBLIKASI KARYA ILMAH UNTUK KEPENTINGAN AKADEMIS

Sebagai sivitas akademika Universitas Sumatera Utara, saya yang bertanda tangan di bawah ini:

Nama	: Novriadi Antonius Siagian
NIM	: 187038008
Program Studi	: Magister (S2) Teknik Informatika
Jenis Karya Ilmiah	: Tesis

Demi pengembangan ilmu pengetahuan, menyetujui untuk memberikan kepada Universitas Sumatera Utara Hak Bebas Royalti Non-Eksklusif (Non-Exclusive Royalty Free Right) atas tesis saya yang berjudul:

PEMBOBOTAN FITUR DATASET MENGGUNAKAN GAIN RATIO GUNA MENINGKATKAN AKURASI METODE NAÏVE BAYESIAN CLASSIFIER CLASSIFIER Beserta perangkat yang ada (jika diperlukan). Dengan Hak Bebas Royalti Non-Eksklusif ini, Universitas Sumatera Utara berhak menyimpan, mengalih media, memformat, mengelola dalam bentuk database, merawat dan mempublikasikan tesis saya tanpa meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis dan sebagai pemegang dan/atau sebagai pemilik hak cipta.

Demikian pernyataan ini dibuat dengan sebenarnya.

Medan, 7 Juli 2020

Novriadi Antonius Siagian

187038008

Telah diuji pada

Tanggal : 04/09/2020

PANITIA PENGUJI TESIS

Ketua : Dr. Sutarman, M.Sc

Anggota : 1. Dr. Sawaluddin, M.IT
2. Prof. Dr. Opim Salim Sitompul
3. Suherman, Ph.D

RIWAYAT HIDUP

DATA PRIBADI

Nama Lengkap berikut gelar	: Novriadi Antonius Siagian, S.Kom
Tempat dan Tanggal Lahir	: Pekanbaru, 29 November 1994
Alamat Rumah	: Jl. Mawar, Gg Sidohony Pekanbaru - Riau
Telp/HP	: 081388851394
Email	: novriadi.antonius95@gmail.com
Website	: https://www.novriadi.com/
Instansi Tempat Kerja	: -

DATA PENDIDIKAN

SD	: Negeri 009 Pekanbaru	TAMAT : 2007
SMP	: Methodist Pekanbaru	TAMAT : 2010
SMA	: SMK Negeri 5 Pekanbaru	TAMAT : 2013
S1	: Teknik Komputer UNILAK Pekanbaru	TAMAT : 2017
S2	: Teknik Informatika USU	TAMAT : 2020

UCAPAN TERIMA KASIH

Puji syukur kehadirat Tuhan Yang Maha Esa karena atas rahmat dan karuniaNya penulis dapat menyelesaikan tesis yang berjudul **"PEMBOBOTAN FITUR DATASET MENGGUNAKAN GAIN RATIO GUNA MENINGKATKAN AKURASI METODE NAÏVE BAYESIAN CLASSIFIER CLASSIFIER"** untuk memenuhi salah satu syarat dalam mencapai gelar Magister pada Jurusan Teknik Informatika Universitas Sumatera Utara Medan. Dalam kesempatan ini penulis menyadari bahwa banyak pihak yang ikut berperan dalam menyelesaikan tesis ini baik moril maupun materil. Oleh karena itu penulis mengucapkan rasa terima kasih kepada:

1. Bapak Prof. Dr. Runtung Sitepu, S.H., M.Hum, selaku Rektor Universitas Sumatera Utara Medan.
2. Bapak Prof. Dr. Opim Salim Sitompul, M.Sc, selaku Dekan Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara Medan.
3. Bapak Prof. Dr. Muhammad Zarlis, M.Sc, selaku Ketua Program Studi S2 Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara Medan.
4. Bapak Dr. Syahril Eefendi, S.Si, M.IT, selaku Sekretaris Program Studi S2 Teknik Informatika, Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara Medan
5. Bapak Dr. Sutarman, M.Sc, selaku Dosen Pembimbing I yang telah memberikan bimbingan dan arahan dalam penyelesaian tesis ini.
6. Bapak Dr. Sawaluddin, M.IT, selaku Dosen Pembimbing II yang juga telah memberikan saran dan masukan untuk perbaikan dan penyelesaian tesis ini.
7. Kedua orangtua penulis: Bapak dan Ibu atas dukungan dan doanya untuk kelancaran dalam menyelesaikan tesis ini.
8. Bapak/Ibu staf, dosen dan karyawan/ti Universitas Sumatera Utara.

9. Teman-teman seangkatan di MTI-Kom-B-2018 yang telah bersama-sama menempuh pendidikan pada Program Studi S2 Teknik Informatika Fakultas Ilmu Komputer dan Teknologi Informasi Universitas Sumatera Utara Medan.

Penulis menyadari bahwa penelitian ini masih jauh dari kata sempurna, ini dikarenakan oleh keterbatasan, kemampuan dan pengetahuan penulis. Harapan penulis, semoga penelitian ini bermanfaat bagi penulis khususnya dan pembaca pada umumnya. Sekali lagi penulis mengucapkan terima kasih, semoga Tuhan membalas kebaikan yang telah diberikan. Amin.

Medan, 7 Juli 2020

Penulis

Novriadi Antonius Siagian

187038008

ABSTRAK

Metode *Naïve bayes* masih memiliki tingkat kelemahan ketika melakukan seleksi atribut, karena *Naïve bayes* sendiri adalah suatu metode pengklasifikasian statistik yang hanya berdasarkan pada teorema *Bayes* sehingga hanya bisa dipakai dengan tujuan untuk memprediksi probabilitas keanggotaan pada suatu grup atau kelas. Oleh karena itu dibutuhkan pembobotan atribut supaya bisa meningkatkan akurasi lebih efektif. Sehingga berdasarkan pembobotan *Gain ratio* disebut dengan *Weight Naïve bayesian classifier (WNB),* mampu memberikan akurasi yang lebih baik dari pada *Naïve bayesian classifier konvensional*. Dimana peningkatan dalam nilai akurasi tertinggi yang diperoleh dari dataset Kualitas Air sama dengan 88,57% dalam model klasifikasi Weight Naïve bayesian classifier, sedangkan nilai akurasi terendah diperoleh dari dataset *Haberman* yang 78,95% dalam model klasifikasi *Naïve bayesian classifier konvensional*. Peningkatan akurasi model klasifikasi Weight Naïve bayesian classifier dalam dataset Kualitas Air adalah 2,9%. Sementara peningkatan nilai akurasi dalam dataset *Haberman* adalah 1,8%. Berdasarkan pengujian yang telah dilakukan pada semua data pengujian, dapat dikatakan bahwa model klasifikasi *Weight Naïve bayesian classifier* dapat memberikan nilai akurasi yang lebih baik daripada yang dihasilkan oleh model klasifikasi *Naïve bayesian classifier konvensional*.

Keywords: Naïve Bayes, Gain ratio, Weight Naïve bayesian classifier, Water Quality, Haberman, Python

DATASET WEIGHTING FEATURES USING GAIN RATIO TO IMPROVE METHOD ACCURACY NAÏVE BAYESIAN CLASSIFIER CLASSIFICATION

ABSTRACT

Naïve bayes method have still has weakness level when do the attribute selection, because its *Naïve bayes* is the method of statistic classification that only based on the Bayes theorem so it only can be used with the purpose to predict the probability of membership in a class or group. It may be needed the weighting of attribute to be able to increase the accuracy that more effective. So, based on the weighting of Gain ratio that is called with *Weight Naïve bayesian classifier* (WNB), able to give the accuracy that batter than *Naïve bayesian classifier konvensional*. Where is the increasing of high score accuracy that is gotten from data set of *water quality* is 88,57% in *Weight Naïve bayesian classifier classification* model, than the lower score of accuracy is gotten from *Haberman* data set, that is 78,95% from *Naïve bayesian classifier konvensional classification* model. The increasing of accuracy *Weight Naïve bayesian classifier classification* model in water quality data set is 2,9%. While, the increasing of accuracy score in *Haberman* data set is 1,8%. Based on the testing that has been done to all the testing data, it can be said that the *Weight Naïve bayesian classifier classification* model can give the better accuracy score than produced by *Naïve bayesian classifier classification* model.

Keywords: Naïve Bayes, Gain ratio, Weight Naïve bayesian classifier, Water Quality, Haberman, Python

DAFTAR ISI

DAFTAR TABEL

DAFTAR GAMBAR

Halaman

BAB 1
PENDAHULUAN

1.1. Latar Belakang

Naïve bayesian classifier (NBC) merupakan pengklasifikasi probabilitas sederhana dalam pembelajaran mesin dengan menghitung probabilitas dataset dengan tujuan untuk memprediksi probabilitas keanggotaan pada suatu grup atau kelas.

Klasifikasi data dilakukan dengan cara menganalisis sampel pelatihan dan memberikan kategorisasi (memberikan kelas) pada data berdasarkan nilai prediksi dari sebuah atribut. Atribut merupakan karakteristik atau ciri khas dari sebuah data. Prediksi yang diharapkan dari sebuah atribut adalah nilai yang menghasilkan deskripsi kategori yang akurat (Amra & Maghari, 2017). Untuk meningkatkan akurasi pengklasifikasian dari sebuah *predictor* maka algoritma yang dapat dikembangkan adalah dengan cara memberikan bobot pada atribut sebelum dilakukan tahapan klasifikasi data (Han *et al.*, 2015).

Penelitian Duan & Lu (2010) mengusulkan pembobotan atribut dari sampel data sebelum dilakukannya tahap klasifikasi. Hal ini bertujuan untuk melakukan peningkatan nilai akurasi dari metode *Naïve bayesian* konvensional dengan cara menerapkan reduksi atribut melalui pemberian bobot pada atribut dari sampel data menggunakan metode *Information gain*. Penerapan dari metode Information gain Weighted Naïve bayesian classifier (IGWNBC) menghasilkan nilai akurasi (*correctness rate*) yang signifikan lebih baik dari metode *Naïve bayesian* konvensional. *Correctness rate* yang dihasilkan dari dataset *car*, *zoo* dan *mushroom* rata-rata memiliki akurasi diatas 97%.

Penelitian Wang & Sun (2016) mengusulkan pemberian bobot pada atribut sampel data menggunakan model *Multivariable Linear Regression* agar menghasilkan nilai koefisien (*weight coefficient*) kemudian diklasifikasikan menggunakan *Naïve bayesian Classifier* (NBC). Penerapan dari model MLRM dan NBC menghasilkan nilai akurasi (*correctness rate*) yang signifikan lebih baik dari metode *Naïve bayesian* konvensional. *Correctness rate* yang dihasilkan dari dataset *UCI Machine Learning* sebesar 80%.

Penelitian Mao *et al.* (2017) mengusulkan pembobotan atribut lokal menggunakan metode *K-Nearest Neighbors Classifier* (KNN). Metode ini dilakukan untuk menghasilkan sejumlah *K* tetangga terdekat dari data sampel dan menghitung nilai probabilitas dari masing-masing atribut untuk kemudian diberikan bobot pada masing-masing atribut kemudian diklasifikasikan menggunakan *Naïve bayesian Classifier* (NBC). Data sampel digunakan berasal dari data historis rute bus dan data kondisi cuaca selama bulan Agustus sampai Desember 2014. Hasil akurasi yang diperoleh dari metode *Naïve bayesian Local Attribute Weighted KNN* sebesar 89%.

Penelitan Duneja & Puyalnithi (2017) Dasar pembobotan atribut pada KNN menggunan *Gain ratio*. KNN dengan menggunakan *Gain ratio* dinilai lebih intuitif dan mudah untuk dipahami. Adapun hasil yang diperoleh dalam penelitian tersebut, pembobotan atribut dengan menggunakan *Gain ratio* mampu meningkatkan akurasi tertinggi sebesar 85%.

Berbagai faktor yang mempengaruhi para penderita jantung seperti: historis uji laboratorium, data demografis, tekanan darah dan sebagainya, berdasarkan hasil penelitian yang dilakukan menggunakan *Naïve bayesian Classifier* konvensional memiliki *correctness rate* sebesar 89% (Repaka *et al.*, 2019).

Berdasarkan latar belakang yang telah dipaparkan, pemberian bobot pada atribut akan diusulkan adalah metode *Gain ratio,* yang merupakan modifikasi dari *Information gain* yang dipakai untuk mengurangi biasnya. *Gain ratio* memperbaiki *Information gain* dengan mengambil informasi intrinsik dari setiap atribut. Oleh sebab itu pada penelitian ini memanfaatkan *Gain ratio* dalam pemberian bobot terhadap atribut. Diharapkan hal ini dapat meningkatkan nilai akurasi klasifikasi *Naïve bayesian Classifier.*

1.2. Rumusan Masalah

Berdasarkan uraian dari latar belakang, maka rumusan masalah adalah Metode *Naïve bayes* masih memiliki tingkat kelemahan ketika melakukan seleksi atribut, karena *Naïve bayes* sendiri adalah metode pengklasifikasian statistik yang hanya berdasarkan pada teorema *Bayes* sehingga hanya dipakai dengan tujuan untuk memprediksi probabilitas keanggotaan pada suatu grup atau kelas. Oleh karena itu dibutuhkan pembobotan atribut supaya bisa meningkatkan akurasi lebih efektif.

1.3. Batasan Masalah

Adapun batasan masalah yang dibahas adalah:

1. Penelitian ini untuk membahas pengaruh metode pemberian bobot menggunakan *Gain ratio* untuk meningkatkan akurasi *Naïve bayesian classifier* (NBC).
2. Melakukan perbandingan antara *Naïve bayesian classifier* (NBC) konvensional dan Weight menggunakan 2 dataset yang memiliki class dan atribut yang berbeda.
3. Pengukuran akurasi *Naïve bayesian* menggunakan *Confusion matrix*.
4. Penelitian ini menggunakan 2 dataset yaitu *Water quality status* dari hasil penelitian Denades *et al.* (2016), dimana data tersebut merupakan hasil pengumpulan data oleh Kementerian Lingkungan Hidup tentang Ketentuan Kualitas Air yang digolongkan kedalam empat kategori. Dataset *Water quality Status* berjumlah sebanyak 120 *instance* dengan 8 atribut dan terdiri dari 4 kelas. Dataset *Haberman* dari *KEEL-Dataset Repository* dengan alamat url: https://sci2s.ugr.es/keel/dataset.php?cod=62, dimana deskripsi data Haberman merupakan sejumlah data pasien kanker payudara (*breast cancer*) yang memiliki 306 *instance* dengan 3 atribut dan terdiri dari 2 kelas.

1.4. Tujuan Penelitian

Adapun tujuan dari penelitian ini untuk meningkatkan nilai akurasi dari metode *Naïve bayesian Classifier* (NBC) dengan memberikan pembobotan pada setiap atribut memanfaatkan *Gain ratio* serta membandingkan penggunaan *Naïve bayesian Classifier* (NBC) konvensional dan weight.

1.5. Manfaat Penelitian

Manfaat penelitian ini diharapkan dapat digunakan sebagai berikut:

1. Dapat menjadi acuan dalam pembobotan dari atribut dengan memanfaatkan *Gain ratio* pada metode *Naïve bayesian Classifier* (NBC)
2. Dapat memberikan informasi terhadap metode pemberian bobot dataset menggunakan *Gain ratio* untuk meningkatkan akurasi metode *Naïve bayesian Classifier* (NBC)

BAB 2
LANDASAN TEORI

Pada bab ini dijelaskan mengenai klasifikasi sebagai proses analisis dengan memiliki 4 komponen utama teknik klasifikasi. Ada beberapa proses klasifikasi dimana atribut keputusan menjadi label kelas dipresentasikan dalam bentuk aturan klasifikasi dan mengoptimasi keakurasian dari aturan klasifikasi yang dihasilkan. Pada bab ini, dijelaskan mengenai metode *Naïve bayesian (Naïve bayesian Classifier)* dan *Gain ratio*. Setelah itu, dijelaskan mengenai *Weight naïve bayes*, pembobotan dan bobot.

2.1. Teknik Klasifikasi

Proses analisis data untuk menentukan model, supaya dapat menguraikan atau membedakan data kelas yang penting agar bisa digunakan sebagai bahan untuk memprediksi adanya kelas dari sebuah objek yang dimana label kelasnya tidak dapat diketahui. Model yang ditemukan berdasarkan dari sebuah analisis *data training* atau dari sebuah objek data yang dimana kelasnya diketahui (Han, et al., 2012). Ada beberapa model yang merupakan algoritma klasifikasi yang sering digunakan, diantaranya: algoritma genetika, *k-nearest neighbor*, C4.5, *rule based, naive bayes*.

Teknik klasifikasi terdapat empat komponen utama yaitu:

(1) *Class label attribute*

Proses klasifikasi untuk mempresentasikan label berbetuk kategori yang terdapat pada objek data. Contohnya: risiko penyakit paru, risiko kredit, jenis pinjaman dan sebagainya.

(2) *Predictor*

Variabel yang mempresentasikan karakteristik data pada atribut data Contohnya: makan atau tidak, memancing atau tidak dan lain sebagainya.

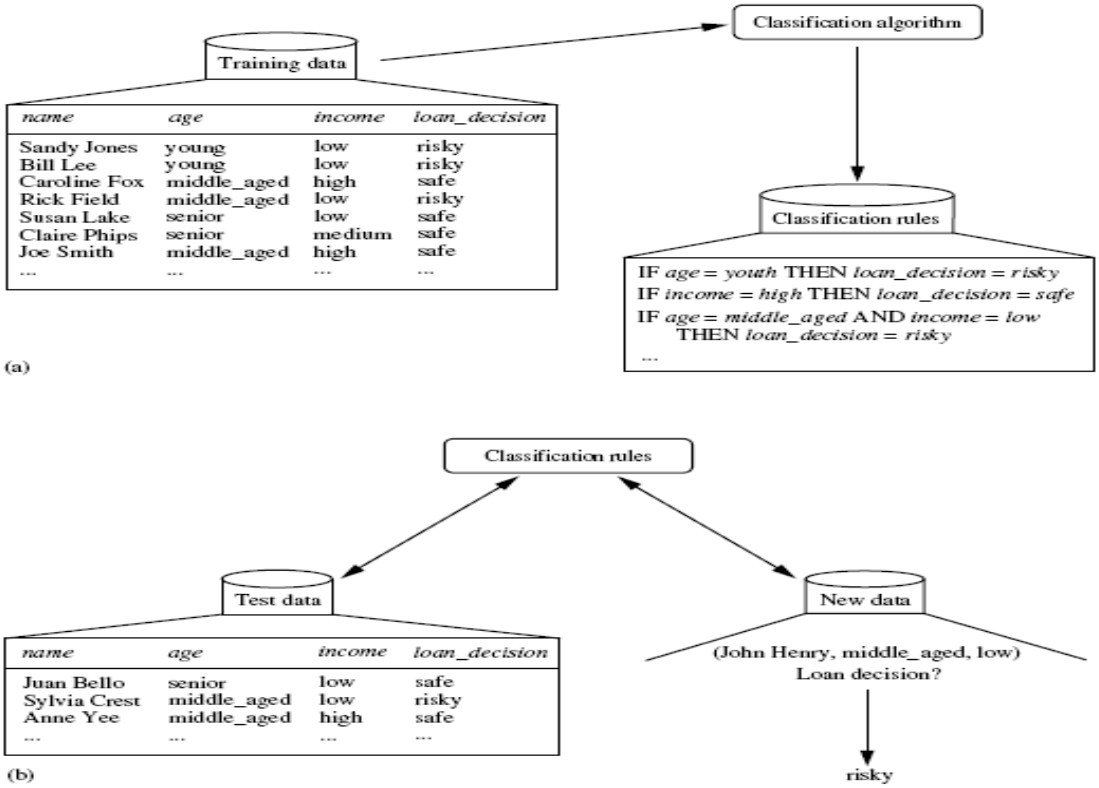
(3) *Training dataset*

Satu set data memiliki nilai dari komponen *class* dan *predictor* digunakan untuk menentukan kelas yang cocok berdasarkan *predictor.*

(4) *Testing dataset*

Suatu proses dalam pengklasifikasian dimana *predictor* menggunakan data baru yang merupakan hasil dari pengukuran akurasi.

Proses klasifikasi dapat dilihat pada contoh Gambar 2.1



Gambar 2.1 Proses Klasifikasi

(Sumber: Han, *et al*. 2012)

Gambar 2.1. (a) proses pembelajaran data pelatihan dianalisis dengan algoritma klasifikasi. Atribut keputusan menjadi label kelas dan model pembelajaran dipresentasikan dalam bentuk aturan klasifikasi. (b) proses klasifikasi digunakan untuk memperkirakan keakuratan aturan klasifikasi. Jika akurasi tersebut diterima maka aturan yang akan diperoleh juga digunakan untuk klasifikasi data baru (Han *et al.*, 2012).

2.2. Metode *Naïve bayesian (Naïve bayesian Classifier)*

Naïve bayesian merupakan pengklasifikasi probabilitas sederhana dalam pembelajaran mesin dengan menghitung probabilitas dari dataset tujuan untuk memprediksi probabilitas keanggotaan suatu grup atau kelas. Berdasar teorema *Bayes* (aturan *Bayes*) dengan perkiraan independensi (ketidaktergantungan) kuat. Pengertian

independensi adalah jika salah satu fitur dalam data sampel tidak saling berkaitan dengan adanya atau tidak fitur lain dari data yang sama.

Naive Bayes hanya memerlukan jumlah data latih yang kecil untuk bisa mengetahui perkiraan parameter yang dibutuhkan dalam sebuah proses klasifikasi.

Berikut persamaan dasar **teorema *Bayes*** oleh (Amra & Maghari, 2017):

$$P\ (C|\ X) = \frac{P(X|C)P(C)}{P(Ci)} \qquad (2.1)$$

Keterangan :

P(C|X) : Probabilitas *Posterior* kelas (*C*, target) diberikan *predictor* (*X*, atribut)

P(X|C) : Mencari nilai parameter yang paling besar (*likelihood*)

P(C) : Probabilitas kelas (*C*, target) sebelumnya

P(X) : Probabilitas *predictor* (*X*, atribut) sebelumnya

Saat Proses pelatihan berlangsung, wajib dilakukan proses pembelajaran probabilitas akhir berdasar informasi yang didapat dari data latih.

Berikut persamaan *Naïve bayesian* untuk klasifikasi (Wang & Sun, 2016):

$$P\ (C_j\ |X) = \frac{P(C_j)\prod_{k=1}^{n} P(X_k \mid C_j)}{P(X)} \qquad (2.2)$$

Keterangan:

$P\ (C_j\ |X)$: Probabilitas data *X* pada kelas *C* ke-*j*

$P\ (C_j)$: Probabilitas awal kelas *C* ke-*j*.

$\prod_{k=1}^{n} P(X_k \mid C_j)$: Probabilitas independent kelas *C* ke-*j* dari fitur vector *X* ke-*k*

Nilai *P(X)* tetap sehingga perhitungan prediksinya cukup menghitung bagian $P(C_j)\prod_{k=1}^{n} P(X_k \mid C_j)$ dengan memilih kelas paling besar sebagai hasil prediksi. Sedangkan untuk probabilitas independen $\prod_{k=1}^{n} P(X_k \mid C_j)$ merupakan pengaruh dari semua fitur data kepada setiap kelas *C* ke-*j*.

Menurut Han & Kamber, (2012), Proses *Naïve bayesian classifier* sebagai berikut:

1) Diasumsikan *variable* D menjadi pelatihan *tuple* dan label kelas yang terkait. Setiap *tuple* diwakili oleh vektor atribut *n*-dimensi, $X = (x_1, x_2,...,x_n)$, menggambarkan pengukuran *n* yang dilakukan pada *tuple* dari atribut *n*, masing-masing, $A_1, A_2, ..., A_n$.

2) Misalkan ada kelas m, C_1, C_2, ..., C_m. Diberikan *tuple*, *X*, *classifier* akan memprediksi *X* termasuk kelas yang memiliki probabilitas posterior tertinggi, *Classifier naive bayesian* akan memprediksi bahwa *X tuple* milik kelas C_i jika dan hanya jika :

$$P(C_i \mid X) > P(C_j \mid X) \quad for\ 1 \leq j \leq m, j \neq i \tag{2.3}$$

Memaksimalkan $P(C_i \mid X)$. *Ci* kelas untuk $P(C_i \mid X)$ dimaksimalkan disebut posteriori maksimal menggunakan teorema *Bayes* pada persamaan 2.1

3) Ketika $P(X)$ sebagai konstan terdahap seluruh kelas. Jika probabilitas kelas sebelumnya tidak di ketahui, akan di letakan pada kelas yang sama, yaitu, $P(C_1) = P(C_2) = \cdots = P(C_m)$, maka dari itu akan memaksimalkan $P(X \mid C_i)$. Jika tidak, akan di maksimalkan $P(X \mid C_i)\, P(C_i)$. Dalam hal ini, probalitas sebelum nya dapat di perkirakan dengan $P(C_i) = |C_{i,D}| / |D|$, dimana $|C_{i,D}|$ adalah jumlah *tuple* pelatihan kelas C_i di *D*.

4) Ketika *Dataset* diberikan atribut, akan terasa lebih sulit mengihitung $P(X|C_i)$. Untuk mengurangi perhitungan dalam mengevaluasi $P(X|C_i)$, dibuat asumsi *naïve* independensi kelas bersyarat. Nilai-nilai atribut adalah kondisi independen satu sama lain, diberikan kelas label dari *tuple* (yaitu bahwa tidak ada hubungan ketergantungan antara atribut) Maka dapat dengan mudah memperkirakan probabilitas $P(x_1 \mid C_i)$, $P(x_2 \mid C_i)$,. . . , $P(x_n \mid C_i)$ dari pelatihan *tuple*. Disini x_k nilai atribut A_k untuk *tuple X*. Akan mengetahui apakah atribut tersebut kategorikal atau benilai kontinu, Misalnya, untuk menghitung $P(X \mid C_i)$ seperti hal berikut :

   1) Jika A_k adalah kategorikal, maka $P(X_k \mid C_i)$ adalah jumlah *tuple* kelas C_i di *D* memiliki nilai X_k untuk atribut A_k, dibagi dengan $|C_i, D|$, jumlah *tuple* kelas C_i di *D*.
   2) Jika A_k bernilai kontinu, diperlukan beberapa pekerjaan, tetapi perhitunganya cukup mudah. Sebuah atribut benilai kontinu memiliki distribusi *Gaussian* dengan rata-rata μ dan standar deviasi σ didefinisikan oleh:

$$g(x,\mu,\sigma) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\pi)^2}{2\sigma^2}}, \tag{2.4}$$

   sehingga :

$$P(X_k \mid C_i) = g(x_k, \mu_{ci}, \sigma_{ci}) \tag{2.5}$$

Hitung μ_{Ci} dan σ_{Ci}, yang merupakan nilai *mean* (rata-rata) dan standar deviasi masing-masing nilai atribut A_k untuk *tuple* pelatihan kelas C_i. Setelah itu gunakan kedua kuantitas dalam Persamaan, bersama-sama dengan x_k, untuk memperkirakan $P(X_k \mid C_i)$.

5) Untuk memprediksi label kelas *X*, $P(X|C_i)P(C_i)$ dievaluasi setiap kelas C_i. Pengklasifikasi memprediksi label kelas *tuple X* adalah kelas C_i, jika:

$$P(X|C_i)\,P(C_i) > P(X|C_j)\,P(C_j) \quad for\ 1 \le j \le m, j \ne i \tag{2.6}$$

Label kelas diprediksi merupakan kelas C_i yang mana $P(X \mid C_i)\,P(C_i)$ adalah maksimal.

2.3. *Gain ratio*

Algoritma C4.5 merupakan metode pohon keputusan dalam pemilihan atributnya didasarkan pada *Gain ratio*. *Gain ratio* merupakan pengembangan dari *Information gain*, yang menghilangkan nilai bias dari setiap atribut. *Gain ratio* mengambil informasi intrinsik dari setiap atribut dari *Information gain* (Priyadarsini *et al.* 2011). Adapun langkah-langkah dalam penentuan *Gain ratio* adalah sebagai berikut :

1) Hitung nilai *Entropy* pada masing – masing atribut, dengan persamaan :

$$Entropy\ (S) = \sum_{i=1}^{n} - pi * log_2 pi \tag{2.7}$$

Dimana:

S : Himpunan Kasus

n : Jumlah Partisi *S*

pi : Proporsi dari *Si* terhadap *S*

2) Hitung nilai *Information gain* pada masing-masing atribut dengan persamaan :

$$Information\ Gain\ (S, A) = Entropy(S) - \sum_{i=1}^{n} \frac{|S_i|}{|S|} \times Entropy(S_i) \tag{2.8}$$

Dimana :

S : Keseluruhan *Dataset*

A : Atribut Subset

N : Jumlah Partisi Atribut *A*

$|S_i|$: Ukuran Subset dari Dataset yang dimiliki atribut pada *A* partisi ke-*i*

$|S|$: Ukuran Jumlah Kasus dalam Dataset

3) Hitung nilai *Split Information* untuk masing-masing atribut dengan persamaan dibawah ini :

$$SplitInfo_A(D) = -\sum_{j=1}^{v} \frac{|D_j|}{|D|} \times log_2(\frac{D_j}{D}) \quad (2.9)$$

Dimana :

D : Keseluruhan Dataset

A : Atribut Subset

v : Jumlah Partisi Atribut *A*

$|D_j|$: Ukuran Subset dari Dataset yang dimiliki atribut pada *A* partisi ke-*j*

$|D|$: Ukuran Jumlah Kasus dalam Dataset

4) Hitung *Gain ratio* dari setiap atribut dengan persamaan :

$$Gain\ Ratio\ (A) = \frac{Gain\ (A)}{SplitInfo(A)} \quad (2.10)$$

Gain ratio ditemukan pada algoritma C4.5, dimana *Gain ratio* digunakan untuk menghitung pengaruh atribut terhadap target dari suatu data (Mitchell, 1997).

2.4. *Weight naïve bayes*

Pengklasifikasi *Bayesian* memiliki keakuratan terbatas karena memerlukan atribut yang tidak tergantung satu sama lain dan memberikan bobot yang sama untuk setiap atribut. Pembobotan adalah nilai sesuatu pada atribut sesuai dengan seberapa penting atau signifikannya. Bobot adalah kelas khusus atribut yang lebih penting dari kelas lain nya (Witten & Frank, 2005). Fungsi pembobotan memiliki sifat-sifat sebagai berikut (Han & Kember, 2012)

1) Memiliki indikator semantik terbaik dari suatu pola adalah dirinya sendiri
2) Menetapkan nilai yang sama untuk dua pola jika keduanya sama kuatnya
3) jika dua pola adalah independen, tidak ada yang bisa menunjukkan arti yang lain. Arti suatu pola dapat disimpulkan ada atau tidak adanya indikator.

Sehingga untuk pemodelan *Weight naïve bayes*, didasarkan pada model pembobotan atribut (Duan & Lu, 2010). Adapun persamaan yang diusulkan berdasarkan pembobotan *Gain ratio* disebut dengan *Weight naïve bayes (WNB)* adalah sebagai berikut:

$$P\ (X_k \mid C_i) = g(x_k, \mu_{ci}, \sigma_{ci}^{wk}) \quad (2.11)$$

dimana:

$P(X_k \mid C_i)$: Probabilitas atribut X_k untuk *tuple* pelatihan kelas C_i

μ_{ci} : *Mean* (nilai rata-rata) untuk *tuple* pelatihan kelas C_i

σ_{ci} : Std.Deviasi dengan bobot (w) pada atribut X_k *tuple* pelatihan kelas C_i

wk : Nilai bobot yang di peroleh dari hasil perthitungan Gain ratio

2.5.Penelitian Terdahulu

Untuk memperkuat bahwa penelitian ini layak untuk diteliti, maka dibawah ini akan dipaparkan beberapa riset yang berkaitan, Dapat dilihat pada Tabel 2.2:

Tabel 2.2 Penelitian Terdahulu

No.	Nama Peneliti dan Tahun	Metode	Hasil Penelitian
1.	Duan & Lu (2010)	Pembobotan atribut dari sampel data sebelum dilakukan klasifikasi. Hal ini bertujuan untuk meningkatkan nilai akurasi metode *Naïve bayesian* konvensional dengan menerapkan reduksi atribut melalui pembobotan setiap atribut dari sampel data menggunakan *Information gain*	Penerapan metode Information gain Weighted Naïve bayesian Classifier (IGWNBC) menghasilkan nilai akurasi (*correctness rate*) yang signifikan lebih baik dari *Naïve bayesian* konvensional. *Correctness rate* yang dihasilkan dari dataset *car*, *zoo* dan *mushroom* memiliki rata-rata akurasi diatas 97%.

Tabel 2.2 Penelitian Terdahulu (*Lanjutan*)

2.	Han *et al.* (2015)	Pendekatan *Principal Component Analysis (PCA)* metode seleksi fitur untuk mereduksi indikator-indikator yang berhubungan dengan pendeteksian terhadap pelanggaran – pelanggaran yang terjadi atas kebijakan keamanan jaringan komputer	Untuk meningkatkan efisiensi dan akurasi dari *network intrusion detection* maka diberikan reduksi indikator-indikator yang ada, kemudian dilakukan pengklasifikasian *Naïve bayesian Classifier*. Nilai akurasi yang diperoleh sebesar 80.31%.
3.	Wang & Sun (2016)	Menerapkan reduksi fitur pada model *Multivariable Linear Regression* melalui pemberian bobot pada atribut dari sampel data menggunakan nilai koefisien (*weight coefficient*).	Penerapan dari model *Multivariable Linear Regression* (MLRM) menghasilkan nilai akurasi (*correctness rate*) yang signifikan lebih baik dari metode *Naïve bayesian* konvensional. *Correctness rate* yang dihasilkan dari dataset *UCI Machine Learning* rata-rata memiliki akurasi diatas 80%.
4.	Amra & Maghari (2017)	Perbandingan terhadap *correctness rate* pada metode klasifikasi *K-Nearest Neighbors Classifier* (KNN) dengan metode klasifikasi *Naïve bayesian Classifier* (NBC). Data sampel diperoleh dari data siswa tingkat menengah pertama yang ada di ibukota Gaza tahun	Tujuan dari penelitian adalah untuk melakukan evaluasi hasil belajar siswa/i yang ada di ibukota Gaza. Hasil akurasi yang signifikan lebih baik diperoleh dari metode NBC sebesar 93.60%.

Tabel 2.2 Penelitian Terdahulu (*Lanjutan*)

		2015.	
5.	Mao *et al*. (2017)	Mengusulkan pemberian bobot pada atribut lokal menggunakan metode *K-Nearest Neighbors Classifier* (KNN). Metode ini dilakukan untuk menghasilkan sejumlah *K* tetangga terdekat dari data sampel dan menghitung nilai probabilitas dari masing-masing atribut untuk kemudian diberikan bobot pada masing-masing atribut sebelum dilakukan klasifikasi menggunakan *Naïve bayesian Classifier* (NBC).	Data sampel yang digunakan berasal dari data historis rute bus dan data kondisi cuaca selama bulan Agustus sampai Desember tahun 2014. Hasil akurasi yang diperoleh dari metode *Naïve bayesian Local Attribute Weighted KNN* sebesar 89%.
6.	Duneja & Puyalnithi (2017)	Menggunakan *Gain ratio* sebagai dasar pembobotan atribut pada KNN. Dari penelitian mereka terlihat bahwa KNN dengan menggunakan *Gain ratio* dinilai lebih intuitif dan mudah untuk dipahami	Hasil akurasi yang diperoleh dari metode *Naïve bayesian Local Attribute Weighted KNN* sebesar 89%.

Tabel 2.2 Penelitian Terdahulu (*Lanjutan*)

7.	Repaka *et al.* (2019)	Memprediksi resiko para penderita penyakit jantung (*Heart Disease Prediction*) menggunakan metode *Naïve bayesian Classifier.* Faktor yang mempengaruhi para penderita penyakit jantung seperti: historis uji laboratorium, data demografis, tekanan darah dan sebagainya, Menganalisis sampel pelatihan dari berbagai faktor yang mempengaruhi para penderita penyakit jantung dan memberikan kategorisasi (memberikan kelas) pada data berdasarkan nilai prediksi dari hasil analisis yang diperoleh.	Berdasarkan hasil penelitian yang telah dilakukan bahwa *Naïve bayesian Classifier* mampu memiliki *correctness rate* sebesar 89%.

2.6. Kontribusi Penelitian

Dari penelitian ini diharapkan meningkatkan nilai akurasi dari metode *Naïve bayesian Classifier* (NBC) dengan memberikan pembobotan pada setiap atribut dengan memanfaatkan *Gain ratio.*

BAB 3

METODE PENELITIAN

Pada bab ini, dijelaskan mengenai proses *Gain ratio* untuk meningkatkan akurasi pada klasifikasi *Naïve bayesian* dengan pemberian bobot pada setiap atribut dataset. Beberapa tahapan untuk pembobotan atribut meliput data *Preprocessing*, pemberian bobot berdasarkan *Gain ratio*, proses klasifikasi *Naïve bayesian* dan analisis pengujian menggunakan *Confusion matrix (two-class prediction)*. Guna mempermudah dalam proses perhitungan pembobotan menggunakan Bahasa Python versi 3.8.3.

3.1. Data Yang Digunakan

Pada penelitian ini, untuk mengetahui kinerja dari metode yang diusulkan maka digunakan satu set data *Water quality Status* dari hasil penelitian Denades *et al.* (2016). Data tersebut merupakan hasil pengumpulan data oleh Kementerian Lingkungan Hidup tentang Ketentuan Kualitas Air yang digolongkan kedalam empat kategori. Dataset *Water quality Status* berjumlah sebanyak 120 *records* dengan 8 *attributes* dan terdiri dari 4 kelas. Rincian dari data *Water quality status* dapat dilihat pada Tabel 3.1:

Tabel 3.1 *Dataset Water quality Status*

No	TSS (mg/L)	DO (mg/L)	COD (mg/L)	*BOD (mg/L)*	*Total phospat (mg/L)*	…	Quality Status
1	2	4	8	2.6	0.1	…	*Good Condition*
2	3	4.5	19.2	3.1	0.14	…	*Good Condition*
3	3	4.4	16	2.9	0.12	…	*Good Condition*
4	4	4.1	4.793	1.32	0.18	…	*Good Condition*
5	4	4.2	8	2.5	0.11	⋯	*Good Condition*
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
120	97	6.3	55.4	10	0.02	…	*Heavily Polluted*

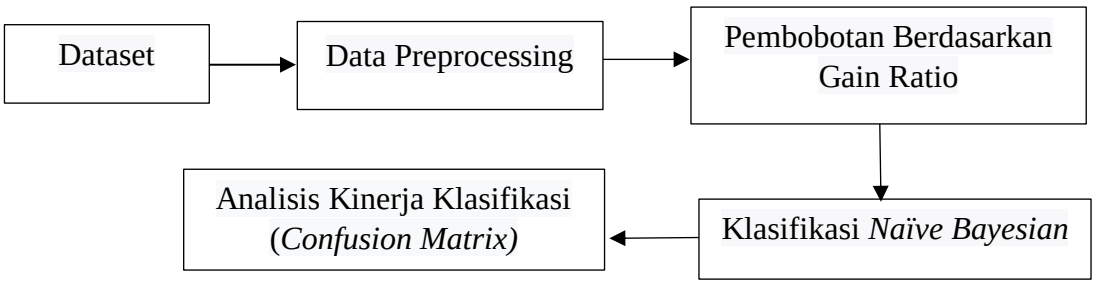
Sumber : *Danades et al. (2016)*

Masing-masing dataset memiliki sejumlah atribut yang proporsional serta mempunyai *missing value*. Data dibagi sebanyak 70% dijadikan data latih dan 30% dijadikan data uji.

3.2. Arsitektur Umum

Penelitian ini menggunakan *Gain ratio* (*GR*) yang difungsikan sebagai alat ukur untuk melihat korelasi dari atribut pada dataset, dimana *Gain ratio* (*GR*) tersebut dijadikan dasar pembobotan terhadap setiap atribut. Diharapkan dengan memberikan bobot pada setiap atribut dapat mengurangi pengaruh dari atribut yang tidak relevan terhadap hasil klasifikasi menggunakan pengklasifikasian *Naïve bayesian*, sehingga mampu meningkat akurasi dari proses klasifikasi tersebut.

Untuk menjelaskan proses tahapan demi tahapan dalam sub bab pada penelitan ini. Secara garis besar dapat dilihat pada Gambar 3.1 berikut:



Gambar 3.1 Arsitektur Umum Penelitian

Pada Gambar 3.1 terlihat untuk meningkatkan akurasi pengklasifikasian *Naïve bayesian* menggunakan metode pembobotan *Gain ratio*, berikut tahapannya:

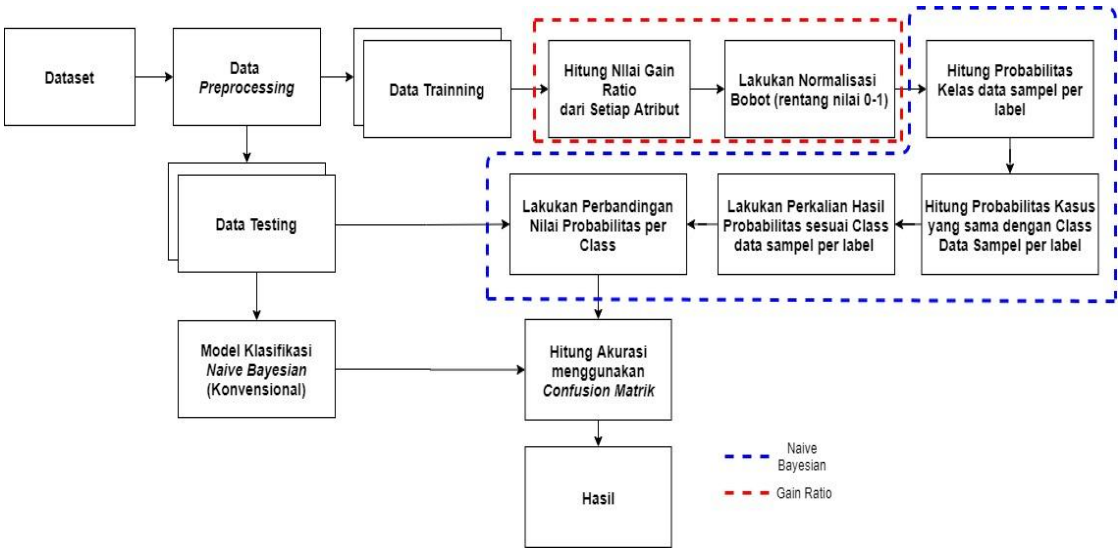
1) Data *Preprocessing*
2) Lakukan proses bobot berdasarkan *Gain ratio*
3) Lakukan proses klasifikasi *Naïve bayesian*
4) Pengujian akurasi menggunakan model klasifikasi *Confusion matrix (two-class prediction)*

3.3. Tahapan Analisis Metode

Untuk mengetahui apakah model dari penelitian ini mampu meningkatkan performa pengklasifikasian *Naïve bayesian*, dilakukan analisis kinerja *Gain ratio+Naïve*

bayesian berdasarkan tabulasi *Confusion matrix (two-class prediction)* yang kemudian hasil akan dibandingkan dengan metode *Naïve bayesian* konvensional.

Prosedur dari penelitian yang diusulkan dapat dilihat pada Gambar 3.2.



Gambar 3.2 Tahapan Analisis Kinerja Metode

Pada Gambar 3.2, terlihat bahwa arsitektur umum penelitian dibagi menjadi beberapa tahapan yaitu: tahapan model pembobotan berdasarkan *Gain ratio* kemudian diklasifikasikan dengan model klasifikasi *Naïve bayesian* dan tahapan model klasifikasi *Naïve bayesian* konvensional. Penelitian ini menggunakan *Gain ratio* dijadikan dasar dalam pemberian bobot terhadap setiap atribut dataset. Semakin tinggi *Gain ratio* dari suatu atribut maka semakin besar korelasi terhadap kelas data, sehingga bobot dari atribut juga semakin tinggi. Selanjutnya proses klasifikasi dengan *Naïve bayesian*, setelah bobot setiap atribut diperoleh akan dilakukan pengklasifikasian dengan *Naïve bayesian*. Penentuan probabilitas antar data, setiap atribut memiliki pengaruh yang berbeda berdasarkan nilai bobot dari atribut tersebut.

Penelitian ini memberikan hasil analisa terhadap kinerja dari metode *Naïve bayesian* dengan pembobotan atribut berdasarkan *Gain ratio* terhadap kinerja dari metode *Naïve bayesian* konvensional.

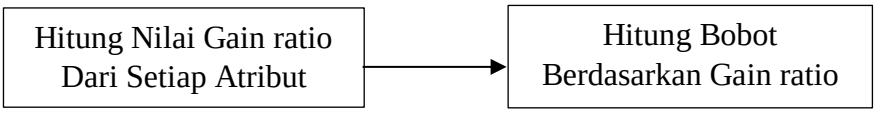
3.3.1 Data Preprocessing

Dalam penelitian ini, dilakukan dua proses *preprocessing*. Proses pertama yaitu dengan penanganan *missing value* dimana atribut yang memiliki nilai numerik (angka)

maka akan digantikan dengan nilai rata-rata (*mean*) dari atribut dalam kolom yang sama. Proses kedua yaitu proses *Cleaning* adalah cara untuk membuang duplikat data.

3.3.2 Proses Pembobotan Atribut berdasarkan Gain ratio

Pada penelitian ini *Gain ratio* (*GR*) tersebut dijadikan dasar pembobotan terhadap setiap atribut. Berikut proses pembobotan bisa dilihat pada Gambar 3.3:



Gambar 3.3 Proses Pembobotan Nilai Atribut

Pada Gambar 3.4, terlihat tahapan proses pembobotan dapat dijelaskan sebagai berikut:

1) Hitung nilai *Gain ratio* dari setiap atribut. Tahapan untuk menentukan *Gain ratio* adalah sebagai berikut :
   1) Hitung *Entropy* menggunakan persamaan (2.7)
   2) Hitung *Gain* menggunkan persamaan (2.8)
   3) Hitung *Split Information* menggunakan persamaan (2.9)
   4) Hitung *Gain ratio* dari setiap atribut dengan persamaan (2.10)

2) Hitung bobot berdasarkan *Gain ratio*. Bobot dihitung menggunakan persamaan normalisasi min-max (Saranya & Manikandan, 2013), dimana bobot terendah setelah dinormalisasi adalah 0.1 dan bobot tertinggi setelah dinormalisasi adalah.

$$W_i = \frac{(G_i - Min(G))}{Max(G) - Min(G)} \times (0.9) + 0.1 \quad (3.1)$$

   Dimana :

   W_i adalah bobot atribut ke-i

   G_i adalah Gain ratio ke-i

   $Min(G)$ adalah nilai terendah dari *Gain ratio*

   $Max(G)$ adalah nilai tertinggi dari *Gain ratio*

3.3.2 Proses Pembentukan Model Klasifikasi Naïve bayesian

Tahapan selanjutnya membentuk model klasifikasi menggunakan metode *Naïve bayesian*. Aturan *Bayes* adalah bahwa hasil (*C*, target) dapat diperkirakan

berdasarkan pada beberapa sampel uji (*X*, atribut) yang sedang diamati. Ada beberapa hal penting dari aturan *Bayes* yaitu:

1) Sebuah probabilitas awal/prior (*C*, target) atau *P(C)* adalah probabilitas dari suatu hipotesis sebelum bukti diamati.
2) Sebuah probabilitas akhir (*C*, target) atau *P(C|X)* adalah probabilitas dari suatu hipotesis setelah bukti diamati.

Perlu diketahui bahwa proses klasifikasi memerlukan sejumlah petunjuk untuk menentukan kelas apa yang sesuai bagi data sampel yang sedang dianalisis.

Untuk mendeskripsikan tahapan-tahapan lebih rinci dari metode *Naïve bayesian*, maka diasumsikan sejumlah sampel 9 fitur *subset* dengan sejumlah 10 *record*. Sebelum dilakukan proses perhitungan probabilitas, maka terlebih dahulu dibentuk ke dalam kategori untuk data yang bertipe numerik, adapun hasil pembentukan kategori seperti berikut:

Tabel 3.2 Sampel Data dalam kategori

No.	X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	Class
1	<20	≤7	≤17	≤4	0	≤13	≤13	0	≤13	0
2	<20	≤7	≤17	≤4	0	≤13	≤13	0	≤13	0
3	<40	≤7	≤17	≤4	0	≤13	≤13	0	≤13	0
4	≥50	≤7	≤17	≤4	1	>26	>26	1	≤13	0
5	<50	≤7	≤24	≤4	0	≤13	≤13	1	≤26	0
6	<50	≤7	≤24	≤4	0	≤13	≤13	0	≤13	0
7	≥50	≤7	≤17	≤8	1	>26	≤13	0	≤13	1
8	<30	≤7	≤24	≤4	0	≤13	≤13	1	≤13	1
9	<50	≤7	≤24	≤8	0	≤13	≤13	0	≤13	0
10	<50	≤7	≤17	≤4	1	≤13	≤13	0	≤13	0

Jika diketahui terdapat data uji = *(X_1 ➔ ≥ 50, X_2 ➔ ≤ 7, X_3 ➔ ≤24, X_4 ➔≤ 8, X_5 ➔1, X_6 ➔>26, X_7 ➔≤13, X_8 ➔1, X_9 ➔≤13*, diberikan *class* pada sampel dari Tabel 3.2, maka selanjutnya dilakukan perhitungan probabilitas posterior tertinggi untuk *class* dari sampel dataset pada Tabel 3.2. Adapun penyelesaian untuk memprediksi *class* dari data uji pada sampel adalah seperti berikut:

Penyelesaian :

1) Hitung probabilitas *Class* data sampel:

   P(C_i) : *P(Class=0) = 8/10 = 0.8*

   P(Class=1) = 2/10 = 0.2

2) Hitung probabilitas jumlah kasus yang sama dengan *Class* data sampel:

P(X | C_i) : *$P(X_1 \rightarrow \geq 50 \mid Class=0) = 1/8 = 0.125$*

$P(X_1 \rightarrow \geq 50 \mid Class=1) = 1/2 = 0.5$

$P(X_2 \rightarrow \leq 7 \mid Class=0) = 8/8 = 1$

$P(X_2 \rightarrow \leq 7 \mid Class=1) = 2/2 = 1$

$P(X_3 \rightarrow \leq 24 \mid Class=0) = 3/8 = 0.375$

$P(X_3 \rightarrow \leq 24 \mid Class=1) = 1/2 = 0.5$

$P(X_4 \rightarrow \leq 8 \mid Class=0) = 1/8 = 0.125$

$P(X_4 \rightarrow \leq 8 \mid Class=1) = 1/2 = 0.5$

$P(X_5 \rightarrow 1 \mid Class=0) = 2/8 = 0.25$

$P(X_5 \rightarrow 1 \mid Class=1) = 1/2 = 0.5$

$P(X_6 \rightarrow >26 \mid Class=0) = 1/8 = 0.125$

$P(X_6 \rightarrow >26 \mid Class=1) = 1/2 = 0.5$

$P(X_7 \rightarrow \leq 13 \mid Class=0) = 7/8 = 0.875$

$P(X_7 \rightarrow \leq 13 \mid Class=1) = 2/2 = 1$

$P(X_8 \rightarrow 1 \mid Class=0) = 2/8 = 0.25$

$P(X_8 \rightarrow 1 \mid Class=1) = 1/2 = 0.5$

$P(X_9 \rightarrow \leq 13 \mid Class=0) = 7/8 = 0.875$

$P(X_9 \rightarrow \leq 13 \mid Class=1) = 2/2 = 1$

3) Lakukan perkalian semua nilai hasil perhitungan probabilitas sesuai dengan *Class* data sampel:

P(X | C_i) :

P(X|Class=0)= 0.125 x 1 x 0.375 x 0.125 x 0.25 x 0.125 x 0.875 x 0.25 x 0.875
= 3.504

P(X|Class=1)= 0.5 x 1 x 0.5 x 0.5 x 0.5 x 0.5 x 1 x 0.5 x 1
= 0.015

P(X | C_i) * P(C_i) :

*P(X|Class=0) * P(Class=0) = 3.504 * 0.8 = 2.80*

*P(X|Class=1) * P(Class=1) = 0.015 * 0.2 = 0.003*

4) Bandingkan hasil akhir prediksi dari perhitungan probabilitas data sampel untuk data uji = ($X_1 \rightarrow \geq 50$, $X_2 \rightarrow \leq 7$, $X_3 \rightarrow \leq 24$, $X_4 \rightarrow \leq 8$, $X_5 \rightarrow 1$, $X_6 \rightarrow >26$, $X_7 \rightarrow \leq 13$, $X_8 \rightarrow 1$, $X_9 \rightarrow \leq 13$ adalah :

Nilai pada *"Class = 0"* lebih besar dari nilai pada *"Class = 1"* maka hasil akhir prediksi Class untuk data uji adalah **"Class = 0"**

3.4. Pengujian Akurasi menggunakan *Confusion matrix*

Pengujian akurasi yang digunakan penelitian ini adalah *confusion matrix* (Witten, 2015). Menurut Han & Kember (2012) akurasi dikenal sebagai tingkat pengakuan atau merujuk pada kemampuan prediksi classifier. Menurut Vijau Kotu (2015) Akurasi adalah kemampuan klasifikasi memilih semua kelas yang perlu dipilih dan menolak semua kelas yang perlu ditolak. Dengan demikian parameter baik dan buruk dalam pengklasifikasian dengan data uji pada kelas yang berbeda yaitu kelas negatif dan positif merupakan proses dari *Confusion matrix (two-class prediction).*

Tabel 3.3 *Confusion matrix (two-class prediction)*

Two-Class Prediction		**Predicted Class**	
		Yes	No
Actual Class	Yes	True Positive	False Negative
	No	False Positive	True Negative

(Sumber: Witten, et al. 2005)

Tabel 3.3 Terlihat model parameter dari klasifikasi 2 kelas yes dan no. jumlah klasifikasi bernilai benar disebut dengan *True Positive* (TP) dan *True Negatives* (TN). Sedangkan prediksi nilai yang tak tepat atau mempunyai nilai positif pada saat melakukan proses prediksi diinginkan bernilai negatif disebut dengan *False Positive* (FP). Apabila prediksi menghasilkan nilai yang tak tepat atau mempunyai nilai negatif pada saat melakukan proses prediksi diinginkan bernilai positif disebut dengan *False Negative* (FN). Kemudian hasil parameter *Confusion matrix* disebut dengan akurasi, persamaan perhitungan sebagai berikut: (Witten, 2005)

$$Akurasi = \frac{TP+TN}{TP+TN+FP+FN} \quad (3.2)$$

TP (*True Positive*) : Jumlah data didalam kelas positif dengan hasil prediksinya diklasifikasi benar secara actual bernilai positif.

TN (*True Negative*) : Jumlah data didalam kelas positif dengan hasil prediksinya diklasifikasibenar secara actual bernilai negative.

FP (*False Positive*) : Salah satu data dari kelas actual bernilai negative tapi hasil prediksi klasifikasinya pada kelas bernilai positif.

FN (*False Negative*) : Salah satu data dari kelas actual bernilai positif tapi hasil prediksi klasifikasinya pada kelas bernilai negatif

3.5. Software dan Tools yang digunakan

Penelitian ini dibangun dengan Bahasa Python versi 3.8.3 dan menggunakan spesifikasi processor Intel Core i7 3.6GHz dengan kapasitas memory 16 GB. Hal ini dilakukan untuk memudahkan perhitungan nilai-nilai dalam proses pembobotan *Gain ratio*, pengujian dan evaluasi model klasifikasi *Naïve bayesian*.

BAB 4
HASIL DAN PEMBAHASAN

Pada bab ini skema pembobotan setiap atribut dataset *Water quality* dan dataset *Haberman* menggunakan *Gain ratio* sebagai parameter. Untuk mempermudah perhitungan digunakan aplikasi Rapid Miner® versi 5.3. Terdapat nilai tertinggi dan terendah disetiap atribut. Setelah itu, dilakukan model klasifikasi *Naïve bayesian (NB)* Konvensional dan model klasifikasi *Weight naïve bayes (WNB)* menggunakan pendekatan *Gain ratio* pada dataset yang digunakan. Pengujian hasil nilai akurasi berdasarkan *Confusion matrix*.

4.1. Hasil

Untuk mengimplementasikan kinerja metode yang diusulkan digunakan dua set data yakni dataset *Water quality* dan dataset *Haberman*. Dataset *Water quality* berasal dari hasil penelitian Denades *et al.* (2016), data tersebut merupakan hasil pengumpulan data oleh Kementerian Lingkungan Hidup tentang Ketentuan Kualitas Air yang digolongkan kedalam empat kategori. Dataset Haberman berasal dari *KEEL-Dataset Repository* dengan alamat url: https://sci2s.ugr.es/keel/dataset.php?cod=62, deskripsi data Haberman merupakan sejumlah data pasien kanker payudara (*breast cancer*) diprediksikan apakah pasien tersebut mampu bertahan hidup selama 5 tahun atau lebih (*Positive*) ataupun sebaliknya pasien tersebut meninggal dalam kurun waktu 5 tahun (*Negative*) pasca dilakukannya operasi.

4.1.1 Hasil Persiapan Data Awal (Data Preprocessing)

Pada penelitian ini, dilakukan dua proses *preprocessing* yaitu dimana atribut yang memiliki nilai numerik (angka) maka akan digantikan dengan nilai rata-rata (*mean*) dari atribut dalam kolom yang sama. *Cleaning* adalah cara untuk membuang duplikat data sehingga jumlah dataset *Water quality Status* yang semula sebanyak 120 *instance* menjadi 117 *instance* dan hal sama juga berlaku untuk jumlah dataset Haberman yang semula sebanyak 306 *instance* menjadi 289 *instance*

Berikut adalah hasil implementasi *data preprocessing* pada dataset *Water quality Status* menggunakan bahasa pemrograman *Python*:

```
==================== Uraian Dataset Water Quality ==========
      TSS (mg/L)  DO (mg/L)  ...    Pij     Quality Status
0            2.0       4.00  ...   0.76     Good Condition
1            3.0       4.50  ...   0.88     Good Condition
2            3.0       4.40  ...   0.91     Good Condition
3            4.0       4.10  ...   0.87     Good Condition
4            4.0       4.20  ...   0.81     Good Condition
..           ...        ...  ...    ...                ...
115         75.0       4.34  ...  10.49  Heavily Polluted
116         85.0       2.59  ...  13.21  Heavily Polluted
117         89.0       6.87  ...  12.58  Heavily Polluted
118         95.0       3.02  ...  14.78  Heavily Polluted
119         97.0       6.30  ...  10.74  Heavily Polluted

[120 rows x 9 columns]

==================== Hasil Preprocessing =================
      TSS (mg/L)  DO (mg/L)  ...    Pij     Quality Status
0            2.0       4.00  ...   0.76     Good Condition
1            3.0       4.50  ...   0.88     Good Condition
2            3.0       4.40  ...   0.91     Good Condition
3            4.0       4.10  ...   0.87     Good Condition
4            4.0       4.20  ...   0.81     Good Condition
..           ...        ...  ...    ...                ...
115         75.0       4.34  ...  10.49  Heavily Polluted
116         85.0       2.59  ...  13.21  Heavily Polluted
117         89.0       6.87  ...  12.58  Heavily Polluted
118         95.0       3.02  ...  14.78  Heavily Polluted
119         97.0       6.30  ...  10.74  Heavily Polluted

[117 rows x 9 columns]
#Record(Before): 120
#Record(After): 117
#Record Remove: 3
```

Gambar 4.1 *Output Data Preprocessing* (Dataset *Water quality Status*)

Untuk menghasilkan *output* seperti Gambar 4.1, maka digunakan listing program *Python* sebagai berikut:

```
import numpy as np
import pandas as pd
dataset = pd.read_csv(r'..\Dataset\Origin\Water.csv')
print('==== Uraian Dataset Water quality =====\n',dataset)
missing=dataset.dropna(axis=0,how='any')
data=missing.drop_duplicates()
data.to_csv(r'..\Dataset\Water.csv',index=False,header=True)
print('\n====== Hasil Preprocessing =======\n',data)
print('#Record(Before):',len(dataset),'\n#Record(After):',len(data),'\n#Record
Remove:',(len(dataset)-len(duplicate)),'\n')
```

Berikut adalah hasil implementasi *data preprocessing* pada dataset *Haberman* menggunakan bahasa pemrograman *Python*:

```
==================== Uraian Dataset Haberman ==================
       Age   Year   Positive     Status
0       38     59          2   negative
1       39     63          4   negative
2       49     62          1   negative
3       53     60          2   negative
4       47     68          4   negative
..     ...    ...        ...        ...
301     57     64          1   positive
302     63     62          0   negative
303     42     61          4   negative
304     43     64          2   negative
305     52     66          4   positive

[306 rows x 4 columns]

===================== Hasil Preprocessing ====================
       Age   Year   Positive     Status
0       38     59          2   negative
1       39     63          4   negative
2       49     62          1   negative
3       53     60          2   negative
4       47     68          4   negative
..     ...    ...        ...        ...
301     57     64          1   positive
302     63     62          0   negative
303     42     61          4   negative
304     43     64          2   negative
305     52     66          4   positive

[289 rows x 4 columns]
#Record(Before): 306
#Record(After): 289
#Record Remove: 17
```

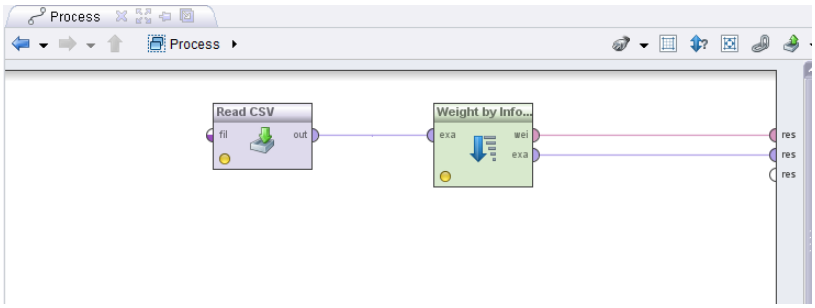
Gambar 4.2 *Output Data Preprocessing* (Dataset *Haberman)*

Untuk menghasilkan *output* seperti Gambar 4.2, maka digunakan listing program *Python* sebagai berikut:

```
dataset = pd.read_csv(r'..\Dataset\Origin\Haberman.csv')
print('==== Uraian Dataset Haberman =====\n',dataset)
missing=dataset.dropna(axis=0,how='any')
data=missing.drop_duplicates()
data.to_csv(r'..\Dataset\Water.csv',index=False,header=True)
print('\n====== Hasil Preprocessing =======\n',data)
print('#Record(Before):',len(dataset),'\n#Record(After):',len(data),'\n#Record
Remove:',(len(dataset)-len(duplicate)),'\n')
```

4.1.2 Hasil Perolehan Nilai Bobot Gain ratio

Gain ratio (*GR*) tersebut dijadikan dasar pembobotan terhadap setiap atribut. Untuk mempermudah proses perhitungan *Gain ratio* terhadap seluruh data, digunakan aplikasi pendukung Rapid Miner® versi 5.3. Adapun *output* hasil perolehan nilai bobot dari proses *Gain ratio* dapat dilihat sebagai berikut:



Gambar 4.3 Skema Proses Perolehan Nilai Bobot Gain ratio

attribute	weight
TSS (mg/L)	0.441
DO (mg/L)	0.257
COD (mg/L)	0.325
BOD (mg/L)	0.488
Total phospat (mg/L)	0.274
Fecal Coliform (mg/L)	0.624
Total Coliform (mg/L)	1
Pij	1

Gambar 4.4 Nilai Bobot Gain ratio (Dataset *Water quality Status)*

Merujuk pada Gambar 4.3 merupakan skema proses penyelesaian bobot menggunakan metode Gain ratio terhadap dataset masing – masing dataset. Operator *Read CSV* digunakan untuk membaca *instance* dataset dan operator *Weight by Information Gain ratio* untuk menghasilkan nilai bobot. Pada Gambar 4.4 terlihat bahwa nilai *Gain ratio* untuk atribut *Total Coliform* dan *Pij* yang tertinggi masing-masing sebesar 1. Nilai *Gain ratio* terendah ada pada atribut *DO* yakni sebesar 0.257.

Adapun *output* hasil perolehan nilai bobot dari proses *Gain ratio* terhadap dataset *Haberman* dapat dilihat pada Gambar 4.5:

attribute	weight
Age	0.219
Year	0.005
Positive	0.195

Gambar 4.5 Nilai Bobot Gain ratio (Dataset *Haberman)*

Gambar 4.5 merupakan hasil perolehan nilai bobot masing – masing karakteristik (atribut) dari dataset *Haberman.* Dari Gambar 4.5 terlihat bahwa nilai *Gain ratio* untuk atribut *Age* yang tertinggi sebesar 0.219, disusul atribut *Year* dan *Positive* masing – masing sebesar 0.195 dan 0.005.

Setelah bobot diperoleh dari kedua dataset maka selanjutnya dilakukan proses awal klasifikasi menggunakan model klasifikasi *Naïve bayesian* konvensional kemudian dilakukan pengujian terhadap metode yang diusulkan.

4.1.3 Hasil Akurasi Model Klasifikasi Naïve bayesian Konvensional

Implementasi hasil akurasi dari klasifikasi terbagi menjadi beberapa bagian yaitu: akurasi model klasifikasi *Naïve bayesian (NB)* Konvensional dan model klasifikasi *Weight naïve bayes (WNB)* menggunakan pendekatan *Gain ratio*

4.1.3.1 Hasil Akurasi Model Naïve bayesian Konvensional (Dataset Water)

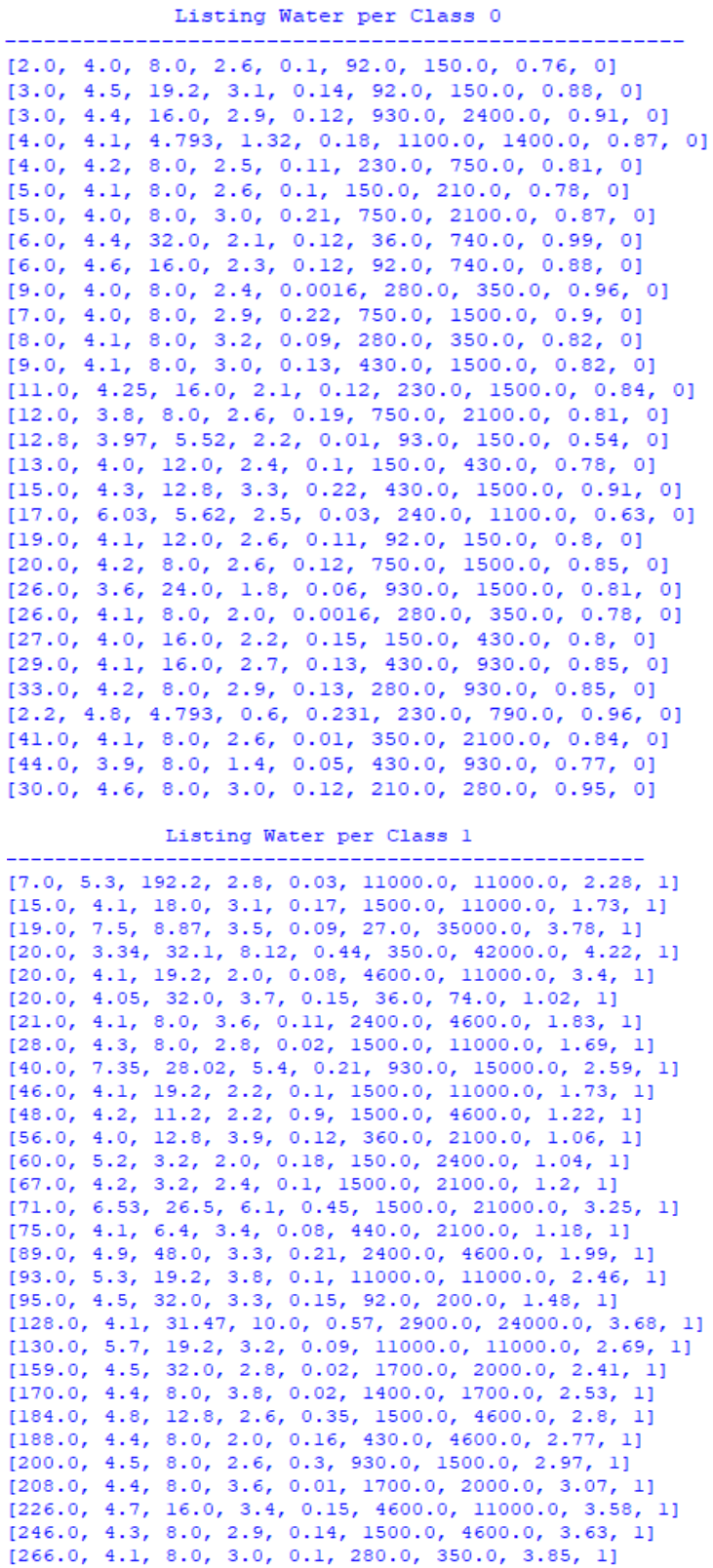
Dataset *Water quality Status* memiliki 8 atribut, 4 kelas dan 120 *instance*. Data dibagi sebanyak 70% dijadikan data latih dan 30% dijadikan data uji. Berikut tahapan dari model klasifikasi *Naïve bayesian Konvensional* untuk dataset *Water* :

1) Hasil pemberian label / kategori kelas (*Encoding Class*):

```
Hasil Encoding Kategori
-----------------------
[Good Condition] => 0
[Lightly Polluted] => 1
[Medium Polluted] => 2
[Heavily Polluted] => 3
```

Gambar 4.6 Hasil Encoding Kelas (Dataset *Water quality*

2) Hasil Pembagian Data per Kelas (*Separate by Class*):



Gambar 4.7a Hasil Pembagian Data per Kelas (Dataset *Water quality)*

```
            Listing Water per Class 2
--------------------------------------------------
[11.0, 8.06, 16.0, 3.44, 0.111, 20000.0, 24000.0, 6.94, 2]
[13.0, 5.1, 16.0, 3.08, 0.25, 9000.0, 70000.0, 5.03, 2]
[14.0, 0.02, 23.72, 8.0, 0.02, 1500.0, 210000.0, 6.65, 2]
[16.0, 7.68, 10.07, 2.68, 0.11199999999999999, 9000.0, 140000.0, 6.05, 2]
[17.0, 5.53, 25.71, 6.8, 0.09, 46000.0, 1100000.0, 9.39, 2]
[19.0, 7.17, 5.0, 2.0, 0.22, 2100.0, 350000.0, 7.39, 2]
[22.0, 6.45, 16.39, 4.8, 0.04, 150000.0, 280000.0, 8.77, 2]
[22.0, 6.57, 2.79, 21.38, 0.18, 4600.0, 110000.0, 5.77, 2]
[25.0, 5.8, 21.26, 7.3, 0.16, 4600.0, 1100000.0, 9.26, 2]
[29.0, 3.85, 7.23, 3.2, 0.12, 2100.0, 280000.0, 7.06, 2]
[29.5, 5.8, 1.7, 10.0, 0.147, 200.0, 90000.0, 5.24, 2]
[30.0, 2.05, 416.0, 150.0, 0.09, 27.0, 35000.0, 7.11, 2]
[32.0, 4.81, 37.66, 9.1, 0.8, 4600.0, 1100000.0, 9.4, 2]
[38.0, 6.33, 17.22, 5.0, 0.03, 28000.0, 350000.0, 7.57, 2]
[52.0, 3.59, 52.26, 20.0, 0.4, 46000.0, 1100000.0, 9.61, 2]
[57.0, 2.65, 43.02, 15.0, 0.12, 9300.0, 1500000.0, 9.88, 2]
[58.0, 6.06, 2.79, 8.0, 0.18, 3600.0, 440000.0, 7.85, 2]
[61.0, 7.75, 26.79, 7.8, 0.77, 4300.0, 930000.0, 9.12, 2]
[63.0, 3.08, 40.92, 8.0, 0.27, 9300.0, 150000.0, 6.37, 2]
[65.0, 4.5, 70.3, 12.0, 0.9, 15000.0, 46000.0, 5.53, 2]
[67.0, 6.18, 4.8, 3.8, 0.01, 150.0, 210000.0, 6.57, 2]
[72.0, 6.02, 6.4, 2.0, 0.07, 3300.0, 420000.0, 7.21, 2]
[76.0, 6.04, 33.66, 6.6, 0.77, 9300.0, 1500000.0, 9.92, 2]
[93.0, 5.33, 84.31, 30.0, 0.54, 9300.0, 150000.0, 6.63, 2]
[128.0, 3.7, 85.46, 15.0, 1.23, 46000.0, 110000.0, 7.4, 2]
[161.0, 4.4, 125.68, 36.0, 0.07, 2900.0, 24000.0, 5.07, 2]
[208.0, 3.5, 176.31, 56.0, 0.11, 21000.0, 29000.0, 6.17, 2]
            Listing Water per Class 3
--------------------------------------------------
[12.0, 3.8, 4.58, 2.4, 0.65, 2800000.0, 44000000.0, 15.31, 3]
[12.0, 5.997000000000001, 9.57, 3.0, 0.11, 2800.0, 2900000.0, 10.68, 3]
[15.0, 4.36, 4.36, 3.0, 0.11, 1500.0, 2100000.0, 10.15, 3]
[18.0, 5.56, 24.93, 5.5, 0.08, 2100.0, 53000000.0, 15.2, 3]
[22.0, 2.9, 56.5, 15.0, 0.01, 46000.0, 11000000.0, 13.02, 3]
[24.0, 3.75, 34.75, 12.0, 0.3, 1500.0, 2100000.0, 10.28, 3]
[25.0, 3.44, 35.63, 10.2, 0.03, 20000.0, 4400000.0, 11.51, 3]
[26.0, 4.62, 38.29, 11.6, 0.06, 1500.0, 29000000.0, 14.31, 3]
[26.0, 3.52, 34.41, 10.0, 0.11, 270000.0, 3500000.0, 11.33, 3]
[26.0, 3.89, 36.16, 10.3, 0.06, 15000.0, 2100000.0, 10.36, 3]
[31.0, 5.15, 14.9, 5.0, 0.67, 3400.0, 4200000.0, 11.33, 3]
[32.0, 0.06, 48.89, 16.0, 0.06, 23000.0, 7500000.0, 12.4, 3]
[32.0, 3.51, 95.26, 30.0, 0.12, 28000.0, 3600000.0, 11.37, 3]
[32.0, 3.85, 32.56, 7.8, 0.07, 21000.0, 2900000.0, 10.86, 3]
[37.0, 6.09, 15.35, 5.5, 0.16, 7500.0, 2100000.0, 10.27, 3]
[38.0, 7.5, 26.64, 8.0, 0.62, 35000.0, 42000000.0, 15.07, 3]
[39.0, 3.8, 9.25, 3.0, 0.06, 28000.0, 3600000.0, 11.13, 3]
[39.0, 3.8, 16.21, 5.0, 0.05, 2800000.0, 35000000.0, 14.93, 3]
[42.0, 8.43, 26.69, 8.4, 0.23, 15000.0, 21000000.0, 13.91, 3]
[44.0, 4.1, 20.98, 7.0, 0.08, 21000.0, 28000000.0, 14.35, 3]
[48.0, 4.65, 12.6, 4.0, 0.69, 1200.0, 29000000.0, 14.3, 3]
[50.0, 5.82, 69.18, 15.0, 0.69, 2700.0, 42000000.0, 15.03, 3]
[58.0, 4.16, 38.15, 6.6, 0.01, 20000.0, 2700000.0, 10.76, 3]
[64.0, 4.1, 6.7, 3.5, 0.08, 28000.0, 3500000.0, 11.11, 3]
[72.0, 4.1, 4.17, 18.0, 0.11, 2700.0, 42000000.0, 14.92, 3]
[75.0, 4.34, 64.76, 20.0, 0.2, 12000.0, 2100000.0, 10.49, 3]
[85.0, 2.59, 51.04, 12.0, 0.1, 64000.0, 12000000.0, 13.21, 3]
[89.0, 6.87, 26.35, 4.5, 0.19, 4300.0, 9300000.0, 12.58, 3]
[95.0, 3.02, 49.34, 12.0, 0.1, 15000.0, 35000000.0, 14.78, 3]
[97.0, 6.3, 55.4, 10.0, 0.02, 1500.0, 2800000.0, 10.74, 3]
```

Gambar 4.7b Hasil Pembagian Data per Kelas (Dataset *Water quality)*

Gambar 4.7a & 4.7b, terlihat hasil partisi sejumlah 117 *instances* dataset *Water quality* menjadi 4 kelas sesuai dengan jumlah recordnya berdasarkan label atau kelas yang di-encoding ditahap sebelumnya mulai dari label kelas 0 sampai kelas 3. Diperoleh bahwa yang termasuk kelas 0 sebanyak 30 record, kelas 1 sebanyak 30 record, kelas 2 sebanyak 27 record dan kelas 3 sebanyak 30 record.

3) Hasil Mean & Std. Deviation per atribut (*Summarize Dataset*):

Rekapitulasi Mean & STD Deviation Water per Atribut

	Mean	Std-Deviation	#Record
TSS	5.322650e+01	5.662840e+01	117
DO	4.612966e+00	1.351605e+00	117
COD	2.866099e+01	4.684026e+01	117
BOD	7.882222e+00	1.529964e+01	117
Total_Phospat	1.937111e-01	2.234782e-01	117
Fecal_Coliform	5.843452e+04	3.643384e+05	117
Total_Coliform	4.243984e+06	1.092569e+07	117
Pij	5.750427e+00	4.819122e+00	117

Gambar 4.8 Hasil Mean & Std. Deviation per atribut (Dataset *Water quality)*

Berdasarkan Gambar 4.8, terlihat bahwa atribut *TSS* memiliki nilai rata-rata (Mean) sebesar 5.322 dan nilai simpangan baku (Std.Deviation) sebesar 5.663 dengan jumlah record sebanyak 117 *instances*. Atribut *DO* memiliki nilai Mean sebesar 4.613 dan nilai simpangan baku (Std.Deviation) sebesar 1.352 dengan jumlah record juga sebanyak 117 *instances.* Hal yang sama juga berlaku untuk atribut berikutnya.

4) Hasil Mean & Std. Deviation per kelas (*Summarize by Class*):

Rekapitulasi Mean & STD Deviation Water per Class 0

```
(14.966666666666667, 11.943324014098717, 30)
(4.218333333333332, 0.42280739344768214, 30)
(11.090866666666667, 6.0976558397657925, 30)
(2.4473333333333334, 0.5944915725938067, 30)
(0.11413999999999998, 0.06449483593334404, 30)
(374.56666666666666, 295.26115656264824, 30)
(967.0, 680.609007205793, 30)
(0.8340000000000002, 0.09171169218130597, 30)
```

```
 Rekapitulasi Mean & STD Deviation Water per Class 1
-----------------------------------------------------
(99.83333333333333, 78.749311254058, 30)
(4.702333333333334, 0.9599449816609763, 30)
(22.918666666666663, 33.84896460835208, 30)
(3.584, 1.7571795582694445, 30)
(0.18666666666666662, 0.19101062135720193, 30)
(2357.5, 3144.3852614483794, 30)
(8970.8, 10075.577157764657, 30)
(2.437666666666666, 0.971677125789019, 30)

 Rekapitulasi Mean & STD Deviation Water per Class 2
-----------------------------------------------------
(54.75925925925926, 47.231705935568804, 27)
(5.111851851851852, 1.878442556679985, 27)
(50.72037037037036, 83.75112428551822, 27)
(16.925185185185185, 29.244720545289535, 27)
(0.28925925925925927, 0.32414142124798084, 27)
(17080.62962962963, 30034.67848126453, 27)
(438814.81481481483, 479300.61524275114, 27)
(7.368888888888888, 1.5491743044735236, 27)

 Rekapitulasi Mean & STD Deviation Water per Class 3
-----------------------------------------------------
(43.5, 24.92990172463582, 30)
(4.4692333333333325, 1.6189823471026128, 30)
(32.12, 22.035614276704973, 30)
(9.476666666666667, 6.136925172650604, 30)
(0.1943333333333333, 0.22321604399964304, 30)
(209790.0, 705776.345520698, 30)
(16146666.666666666, 16734967.177642792, 30)
(12.522999999999998, 1.8830260862184864, 30)
```

Gambar 4.9 Hasil Mean & Std. Deviation per kelas (Dataset *Water quality)*

Merujuk pada Gambar 4.9, jika diperhatikan sebanyak 30 record atribut *TSS* termasuk kedalam kelas 0 bernilai Mean sebesar 14.97 dan Std. Deviasi sebesar 11.94. Sedangkan sebanyak 30 record atribut *TSS* lainnya termasuk kedalam kelas 1 bernilai Mean sebesar 99.83 dan Std. Deviasi sebesar 78.75. Sementara itu sebanyak 27 record atribut *TSS* berikutnya termasuk kedalam kelas 2 bernilai Mean sebesar 54.76 dan Std. Deviasi sebesar 47.23. Untuk 30 record atribut *TSS* sisanya termasuk kedalam kelas 3 bernilai Mean sebesar 43.5 dan Std. Deviasi sebesar 24.93. Hal yang sama juga berlaku untuk atribut lainnya yang teramsuk pada dataset *Water*.

5) Hasil Perolehan Nilai Probabilitas per kelas (*Probabilities by Class*):

```
 Nilai Probabilitas Water per Class
 ------------------------------------
{0: 6.776853537403209e-10, 1: 1.824465841581955e-16, 2: 1.7432648970305922e-24,
3: 7.8815202265158894e-31}
```

Gambar 4.10 Hasil Perolehan Probabilitas per kelas (Dataset *Water quality)*

Merujuk pada Gambar 4.10, jika diperhatikan nilai probabilitas yang ada pada kelas 3 yakni *Heavily Polluted* memiliki probabilitas terbesar yakni 7.88 terhadap dataset *Water quality,* dilanjutkan dengan probabilitas kelas 0 sebesar 6.77, kelas 1 sebesar 1.82 dan kelas 2 sebesar 1.74. Nilai probabilitas kelas *Heavily Polluted* disebut *Maximum a Posteriori Hypothesis* (MAP) karena memiliki nilai probabilitas kelas tertinggi.

6) Hasil Pembagian data (*Train Test Splitting*)

Rekapitulasi Dataset Water (Data Latih)

	TSS	DO	COD	...	Total_Coliform	Pij	Aktual-Class
0	89.0	4.90	48.00	...	4600.0	1.99	1
1	30.0	2.05	416.00	...	35000.0	7.11	2
2	89.0	6.87	26.35	...	9300000.0	12.58	3
3	22.0	2.90	56.50	...	11000000.0	13.02	3
4	38.0	7.50	26.64	...	42000000.0	15.07	3
..	...	...	...	...	...	...	...
77	26.0	3.60	24.00	...	1500.0	0.81	0
78	39.0	3.80	16.21	...	35000000.0	14.93	3
79	75.0	4.34	64.76	...	2100000.0	10.49	3
80	29.0	4.10	16.00	...	930.0	0.85	0
81	11.0	4.25	16.00	...	1500.0	0.84	0

[82 rows x 9 columns]

Gambar 4.11 Data Latih *Water quality (Naïve bayesian Konvensional)*

Pada Gambar 4.11, terlihat uraian tentang rekapitulasi data latih (*training data*) sebanyak 82 *instance* berserta kategori class dari 117 *instance* yang diperoleh melalui 70% dari data dijadikan sebagai data latih dataset *Water quality*. Data latih ini yang akan digunakan untuk pembentukan model klasifikasi *Naïve bayesian* Konvensional.

Rekapitulasi Dataset Water (Data Uji)

	TSS	DO	COD	...	Total_Coliform	Pij	Aktual-Class
0	4.0	4.20	8.00	...	750.0	0.81	0
1	6.0	4.40	32.00	...	740.0	0.99	0
2	6.0	4.60	16.00	...	740.0	0.88	0
3	9.0	4.00	8.00	...	350.0	0.96	0
4	7.0	4.00	8.00	...	1500.0	0.90	0
5	12.8	3.97	5.52	...	150.0	0.54	0
6	13.0	4.00	12.00	...	430.0	0.78	0
7	15.0	4.30	12.80	...	1500.0	0.91	0
8	17.0	6.03	5.62	...	1100.0	0.63	0
9	20.0	4.20	8.00	...	1500.0	0.85	0
10	33.0	4.20	8.00	...	930.0	0.85	0
11	30.0	4.60	8.00	...	280.0	0.95	0
12	7.0	5.30	192.20	...	11000.0	2.28	1
13	15.0	4.10	18.00	...	11000.0	1.73	1
14	19.0	7.50	8.87	...	35000.0	3.78	1
15	20.0	4.10	19.20	...	11000.0	3.40	1
..	...	...	...	...	...	...	...
33	42.0	8.43	26.69	...	21000000.0	13.91	3
34	44.0	4.10	20.98	...	28000000.0	14.35	3

[35 rows x 9 columns]

Gambar 4.12 Data Uji *Water quality (Naïve bayesian Konvensional)*

Merujuk pada Gambar 4.12, terlihat uraian tentang rekapitulasi data uji (*testing data*) sebanyak 35 *instance* dari 117 *instance* yang diperoleh melalui 30% dari data dijadikan sebagai data uji dataset *Water quality*. Data uji ini yang akan digunakan untuk pengujian model klasifikasi *Naïve bayesian* Konvensional yang kemudian akan menghasilkan prediksi kelas untuk model klasifikasi *Naïve bayesian* Konvensional dan diperoleh hasil akurasi pengklasifikasian model klasifikasi *Naïve bayesian* Konvensional.

7) Hasil Akurasi *Naïve bayesian Konvensional* (Dataset *Water quality*)

```
Nilai Akurasi Naive-Bayesian (Dataset Water Quality) adalah  85.71428571428571 %

Confusion Matrix
------------------
[[10  2  0  0]
 [ 0  8  1  0]
 [ 0  0  9  2]
 [ 0  0  0  3]]
```

Gambar 4.13 Confusion matrix *Naïve bayesian Konvensional (Water quality)*

Gambar 4.13, Terlihat *output* nilai akurasi yang dihasilkan oleh model klasifikasi *Naïve bayesian* Konvensional menggunakan dataset *Water quality*. Hasil prediksi

dapat dilihat pada Gambar 4.14, sementara hasil kinerja dan performa pengklasifikasian *Naïve bayesian* Konvensional menggunakan metode *Confusion matrix* didasarkan sub-bab 3 yakni pada Tabel 3.3 merujuk pada persamaan 3.2 Berikut adalah hasil perhitungannya:

1) *Accuracy* $= \frac{3+8+9+10}{3+8+9+10+2+2+1} = \frac{30}{35}$ = 0.8571*100% = **85.71%**

2) *Classification_error* = $\frac{2+2+1}{3+8+9+10+2+2+1} = \frac{5}{35}$ = 0.1429*100% = **14.29%**

Rekap Hasil Klasifikasi Naive-Bayes Classifier (Data Uji Water)

	TSS	DO	COD	...	Pij	Aktual-Class	Prediksi-Class
0	4.0	4.20	8.00	...	0.81	0	0
1	6.0	4.40	32.00	...	0.99	0	0
2	6.0	4.60	16.00	...	0.88	0	0
3	9.0	4.00	8.00	...	0.96	0	0
4	7.0	4.00	8.00	...	0.90	0	0
5	12.8	3.97	5.52	...	0.54	0	1
6	13.0	4.00	12.00	...	0.78	0	0
7	15.0	4.30	12.80	...	0.91	0	0
8	17.0	6.03	5.62	...	0.63	0	1
9	20.0	4.20	8.00	...	0.85	0	0
10	33.0	4.20	8.00	...	0.85	0	0
11	30.0	4.60	8.00	...	0.95	0	0
12	7.0	5.30	192.20	...	2.28	1	2
13	15.0	4.10	18.00	...	1.73	1	1
14	19.0	7.50	8.87	...	3.78	1	1
15	20.0	4.10	19.20	...	3.40	1	1
16	60.0	5.20	3.20	...	1.04	1	1
17	75.0	4.10	6.40	...	1.18	1	1
18	159.0	4.50	32.00	...	2.41	1	1
19	184.0	4.80	12.80	...	2.80	1	1
20	188.0	4.40	8.00	...	2.77	1	1
21	16.0	7.68	10.07	...	6.05	2	2
22	19.0	7.17	5.00	...	7.39	2	2
23	22.0	6.45	16.39	...	8.77	2	3
24	25.0	5.80	21.26	...	9.26	2	2
25	57.0	2.65	43.02	...	9.88	2	3
26	58.0	6.06	2.79	...	7.85	2	2
27	67.0	6.18	4.80	...	6.57	2	2
28	72.0	6.02	6.40	...	7.21	2	2
29	76.0	6.04	33.66	...	9.92	2	2
30	128.0	3.70	85.46	...	7.40	2	2
31	208.0	3.50	176.31	...	6.17	2	2
32	26.0	3.89	36.16	...	10.36	3	3
33	42.0	8.43	26.69	...	13.91	3	3
34	44.0	4.10	20.98	...	14.35	3	3

[35 rows x 10 columns]

Gambar 4.14 Hasil Klasifikasi *Naïve bayesian Konvensional (Water quality)*

4.1.3.2 Hasil Akurasi Model *Naïve bayesian Konvensional (Dataset Haberman)*

Dataset *Haberman* memiliki 3 atribut, 2 kelas dan 289 *instance*. Data dibagi sebanyak 80% dijadikan data latih dan 20% dijadikan data uji. Berikut adalah tahapan dari model klasifikasi *Naïve bayesian Konvensional* untuk dataset *Haberman*:

1) Hasil Pemberian Label / Kategori Kelas (*Encoding Class*):

```
Hasil Encoding Status
------------------------
[ negative] => 0
[ positive] => 1
```

Gambar 4.15 Hasil Encoding Kelas (Dataset *Haberman)*

2) Hasil Pembagian Data per Kelas (*Separate by Class*):

```
        Listing Haberman per Class 0
-----------------------------------------------
[45.0, 66.0, 0.0, 0]
[63.0, 60.0, 1.0, 0]
[69.0, 67.0, 8.0, 0]
[61.0, 62.0, 5.0, 0]
[53.0, 58.0, 4.0, 0]
[60.0, 65.0, 0.0, 0]
[74.0, 65.0, 3.0, 0]
[70.0, 58.0, 0.0, 0]
[50.0, 63.0, 13.0, 0]
[62.0, 65.0, 19.0, 0]
[56.0, 65.0, 9.0, 0]
[50.0, 64.0, 0.0, 0]
[51.0, 59.0, 3.0, 0]
[45.0, 67.0, 1.0, 0]
[39.0, 66.0, 0.0, 0]
[42.0, 59.0, 0.0, 0]
[62.0, 58.0, 0.0, 0]
[65.0, 61.0, 2.0, 0]
[53.0, 59.0, 3.0, 0]
[62.0, 59.0, 13.0, 0]
[41.0, 67.0, 0.0, 0]
[54.0, 65.0, 5.0, 0]
[41.0, 60.0, 23.0, 0]
[41.0, 64.0, 0.0, 0]
[44.0, 58.0, 9.0, 0]
[48.0, 67.0, 7.0, 0]
[34.0, 59.0, 0.0, 0]
[72.0, 63.0, 0.0, 0]
[52.0, 59.0, 2.0, 0]
[46.0, 65.0, 20.0, 0]
[46.0, 58.0, 2.0, 0]
[54.0, 65.0, 23.0, 0]
```

Gambar 4.16a Hasil Pembagian Data per Kelas (Dataset *Haberman)*

```
        Listing Haberman per Class 1
-------------------------------------------------
[38.0, 59.0, 2.0, 1]
[39.0, 63.0, 4.0, 1]
[49.0, 62.0, 1.0, 1]
[53.0, 60.0, 2.0, 1]
[47.0, 68.0, 4.0, 1]
[56.0, 67.0, 0.0, 1]
[64.0, 58.0, 0.0, 1]
[55.0, 69.0, 22.0, 1]
[52.0, 61.0, 0.0, 1]
[61.0, 65.0, 8.0, 1]
[64.0, 68.0, 0.0, 1]
[54.0, 62.0, 0.0, 1]
[44.0, 61.0, 0.0, 1]
[49.0, 61.0, 0.0, 1]
[55.0, 58.0, 1.0, 1]
[55.0, 66.0, 18.0, 1]
[43.0, 63.0, 14.0, 1]
[37.0, 59.0, 6.0, 1]
[42.0, 60.0, 1.0, 1]
[34.0, 60.0, 0.0, 1]
[43.0, 63.0, 2.0, 1]
[59.0, 64.0, 4.0, 1]
[41.0, 58.0, 0.0, 1]
[63.0, 61.0, 0.0, 1]
[68.0, 67.0, 0.0, 1]
[42.0, 62.0, 20.0, 1]
[54.0, 59.0, 7.0, 1]
[77.0, 65.0, 3.0, 1]
[60.0, 61.0, 1.0, 1]
[58.0, 59.0, 0.0, 1]
[58.0, 61.0, 1.0, 1]
[72.0, 67.0, 3.0, 1]
[35.0, 64.0, 13.0, 1]
[39.0, 58.0, 0.0, 1]
[73.0, 68.0, 0.0, 1]
[62.0, 62.0, 6.0, 1]
[61.0, 68.0, 0.0, 1]
[76.0, 67.0, 0.0, 1]
[30.0, 65.0, 0.0, 1]
```

Gambar 4.16b Hasil Pembagian Data per Kelas (Dataset *Haberman)*

Pada Gambar 4.16a & 4.16b, terlihat hasil partisi sejumlah 289 *instance* dataset *Haberman* menjadi 2 kelas sesuai dengan jumlah recordnya berdasarkan label atau kelas yang di-encoding ditahap sebelumnya mulai dari label kelas 0 sampai kelas 1. Diperoleh bahwa yang termasuk kelas 0 sebanyak 79 record, kelas 1 sebanyak 210 record.

3) Hasil Mean & Std. Deviation per atribut (*Summarize Dataset*):

```
 Rekapitulasi Mean & STD Deviation Haberman per Atribut
-------------------------------------------------------
               Mean  Std-Deviation   #Record
Age       52.422145      10.876915       289
Year      62.906574       3.275794       289
Positive   4.221453       7.325239       289
```

Gambar 4.17 Hasil Mean & Std. Deviation per atribut (Dataset *Haberman)*

Berdasarkan Gambar 4.17, terlihat bahwa atribut *Year* memiliki nilai rata-rata (Mean) sebesar 62.90 dan nilai simpangan baku (Std.Deviation) sebesar 3.27 dengan jumlah record sebanyak 289 *instance*. Atribut *Age* memiliki nilai Mean sebesar 52.42 dan nilai simpangan baku (Std.Deviation) sebesar 10.87 dengan jumlah record juga sebanyak 289 *instance*. Hal yang sama juga berlaku untuk atribut *Positive* berikutnya.

4) Hasil Mean & Std. Deviation per kelas (*Summarize by Class*):

```
 Rekapitulasi Mean & STD Deviation Haberman per Class 1
-------------------------------------------------------
(51.871428571428574, 11.09323872085335, 210)
(62.91904761904762, 3.260334184241991, 210)
(2.9857142857142858, 6.02981314938874, 210)

 Rekapitulasi Mean & STD Deviation Haberman per Class 0
-------------------------------------------------------
(53.88607594936709, 10.203050501326517, 79)
(62.87341772151899, 3.3373069467169185, 79)
(7.506329113924051, 9.255453579766819, 79)
```

Gambar 4.18 Hasil Mean & Std. Deviation per kelas (Dataset *Haberman)*

Merujuk pada Gambar 4.18, jika diperhatikan sebanyak 79 record atribut *Age* termasuk kedalam kelas 0 bernilai Mean sebesar 53.88 dan Std.Deviasi sebesar 10.20. Sedangkan sebanyak 210 record atribut *Age* lainnya termasuk kedalam kelas 1 bernilai Mean sebesar 51.87 dan Std.Deviasi sebesar 11.09. Hal yang sama juga berlaku untuk atribut *Year* dan *Positive* berikutnya.

5) Hasil Perolehan Nilai Probabilitas per kelas (*Probabilities by Class*):

```
 Nilai Probabilitas Haberman per Class
---------------------------------------
{0: 4.6380729840685175e-05, 1: 7.000801689881911e-06}
```

Gambar 4.19 Hasil Perolehan Probabilitas per kelas (Dataset *Haberman)*

Merujuk pada Gambar 4.19, jika diperhatikan nilai probabilitas yang ada pada kelas 1 yakni *Positive* memiliki probabilitas tertinggi yakni 7.00 terhadap dataset *Haberman,* Sementara probabilitas kelas 0 yakni *Negative* sebesar 4.63. Sehingga nilai

probabilitas kelas *Positive* disebut *Maximum a Posteriori Hypothesis* (MAP) karena memiliki nilai probabilitas kelas tertinggi.

6) Hasil Pembagian data (*Train Test Splitting*)

Rekapitulasi Dataset Haberman (Data Latih)

	Age	Year	Positive	Aktual-Status
0	41.0	58.0	0.0	0
1	42.0	69.0	1.0	1
2	61.0	62.0	5.0	1
3	61.0	68.0	0.0	0
4	50.0	61.0	0.0	0
..	...	...	...	...
227	34.0	60.0	1.0	0
228	60.0	61.0	25.0	0
229	59.0	62.0	35.0	1
230	49.0	64.0	10.0	1
231	69.0	65.0	0.0	0

[232 rows x 4 columns]

Gambar 4.20 Data Latih *Haberman (Naïve bayesian Konvensional)*

Pada Gambar 4.20, terlihat uraian tentang rekapitulasi data latih (*training data*) sebanyak 232 *instance* berserta kategori class dari 289 *instance* yang diperoleh melalui 80% sebagai data latih dataset *Haberman.* Data latih ini digunakan untuk pembentukan model klasifikasi *Naïve bayesian* Konvensional.

Rekapitulasi Dataset Haberman (Data Uji)

	Age	Year	Positive	Aktual-Status
0	38.0	59.0	2.0	0
1	61.0	65.0	8.0	0
2	55.0	58.0	1.0	0
3	43.0	63.0	14.0	0
4	59.0	64.0	4.0	0
5	63.0	61.0	0.0	0
6	54.0	59.0	7.0	0
7	67.0	66.0	0.0	0
8	56.0	65.0	9.0	1
9	70.0	59.0	8.0	0
10	71.0	68.0	2.0	0
11	41.0	65.0	0.0	0
12	30.0	62.0	3.0	0
13	59.0	64.0	7.0	0
14	67.0	65.0	0.0	0
15	39.0	59.0	2.0	0
16	65.0	61.0	2.0	1
17	54.0	65.0	5.0	1
18	50.0	59.0	0.0	0
19	41.0	64.0	0.0	1
20	44.0	67.0	16.0	0
21	38.0	62.0	3.0	0
22	52.0	59.0	2.0	1
23	48.0	64.0	0.0	0
24	34.0	67.0	7.0	0
25	53.0	63.0	0.0	0
26	58.0	58.0	3.0	0
27	49.0	62.0	0.0	0
28	37.0	60.0	15.0	0
29	37.0	60.0	0.0	0
30	53.0	60.0	1.0	0
31	39.0	67.0	0.0	0
32	40.0	58.0	0.0	0
33	59.0	64.0	0.0	0
34	64.0	61.0	0.0	0
35	45.0	59.0	14.0	0
36	48.0	66.0	0.0	0
37	72.0	58.0	0.0	0
38	47.0	61.0	0.0	0
39	47.0	63.0	23.0	1
40	70.0	67.0	0.0	0
41	41.0	64.0	0.0	0
42	47.0	62.0	0.0	1
43	52.0	62.0	1.0	0
44	44.0	64.0	6.0	1
45	65.0	67.0	1.0	0
46	68.0	68.0	0.0	0
47	59.0	63.0	0.0	0
48	45.0	65.0	6.0	1
49	60.0	59.0	17.0	1
50	53.0	65.0	12.0	1
51	70.0	68.0	0.0	0
52	38.0	66.0	11.0	0
53	50.0	63.0	1.0	0
54	61.0	64.0	0.0	0
55	61.0	65.0	0.0	1
56	43.0	59.0	2.0	1

Gambar 4.21 Data Uji *Haberman (Naïve bayesian Konvensional)*

Pada Gambar 4.21, terlihat uraian tentang rekapitulasi data uji (*testing data*) sebanyak 57 *instance* dari 289 *instance* yang diperoleh melalui 30% dari data dijadikan sebagai data uji dataset *Haberman.* Data uji ini yang digunakan untuk pengujian model klasifikasi *Naïve bayesian* Konvensional yang kemudian menghasilkan prediksi kelas untuk model klasifikasi *Naïve bayesian* Konvensional dan diperoleh hasil akurasi pengklasifikasian model klasifikasi *Naïve bayesian* Konvensional.

7) Hasil Akurasi *Naïve bayesian Konvensional* (Dataset *Haberman*)

```
Nilai Akurasi Naive-Bayes Classifier adalah  78.94736842105263 %

 Confusion Matrix
------------------
[[43  1]
 [11  2]]
```

Gambar 4.22 Confusion matrix *Naïve bayesian Konvensional (Dataset Haberman)*

Gambar 4.22, Terlihat *output* nilai akurasi yang dihasilkan oleh model klasifikasi *Naïve bayesian* Konvensional menggunakan dataset *Haberman*. Hasil prediksi dapat dilihat pada Gambar 4.23, Sementara hasil kinerja dan performa pengklasifikasian *Naïve bayesian* Konvensional menggunakan metode *Confusion matrix* didasarkan sub-bab 3 yakni pada Tabel 3.3 merujuk pada persamaan 3.2. Berikut adalah hasil perhitungannya:

1) Accuracy $= \frac{2+43}{2+43+1+11} = \frac{45}{57} = 0.7895*100\% =$ **78.95%**

2) Classification_error $= \frac{1+11}{32+43+1+11} = \frac{12}{57} = 0.2105*100\% =$ **21.05%**

Berikut pada Gambar 4.23 merupakan hasil rekapitulasi prediksi model klasifikasi *Naïve bayesian* Konvensional terhadap dataset *Haberman*:

Rekap Klasifikasi Naive-Bayes Classifier (Data Uji Haberman)

	Age	Year	Positive	Aktual-Status	Prediksi-Class
0	38.0	59.0	2.0	0	0
1	61.0	65.0	8.0	0	0
2	55.0	58.0	1.0	0	0
3	43.0	63.0	14.0	0	0
4	59.0	64.0	4.0	0	0
5	63.0	61.0	0.0	0	0
6	54.0	59.0	7.0	0	0
7	67.0	66.0	0.0	0	0
8	56.0	65.0	9.0	1	0
9	70.0	59.0	8.0	0	0
10	71.0	68.0	2.0	0	0
11	41.0	65.0	0.0	0	0
12	30.0	62.0	3.0	0	0
13	59.0	64.0	7.0	0	0
14	67.0	65.0	0.0	0	0
15	39.0	59.0	2.0	0	0
16	65.0	61.0	2.0	1	0
17	54.0	65.0	5.0	1	0
18	50.0	59.0	0.0	0	0
19	41.0	64.0	0.0	1	0
20	44.0	67.0	16.0	0	1
21	38.0	62.0	3.0	0	0
22	52.0	59.0	2.0	1	0
23	48.0	64.0	0.0	0	0
24	34.0	67.0	7.0	0	0
25	53.0	63.0	0.0	0	0
26	58.0	58.0	3.0	0	0
27	49.0	62.0	0.0	0	0
28	37.0	60.0	15.0	0	0
29	37.0	60.0	0.0	0	0
30	53.0	60.0	1.0	0	0
31	39.0	67.0	0.0	0	0
32	40.0	58.0	0.0	0	0
33	59.0	64.0	0.0	0	0
34	64.0	61.0	0.0	0	0
35	45.0	59.0	14.0	0	0
36	48.0	66.0	0.0	0	0
37	72.0	58.0	0.0	0	0
38	47.0	61.0	0.0	0	0
39	47.0	63.0	23.0	1	1
40	70.0	67.0	0.0	0	0
41	41.0	64.0	0.0	0	0
42	47.0	62.0	0.0	1	0
43	52.0	62.0	1.0	0	0
44	44.0	64.0	6.0	1	0
45	65.0	67.0	1.0	0	0
46	68.0	68.0	0.0	0	0
47	59.0	63.0	0.0	0	0
48	45.0	65.0	6.0	1	0
49	60.0	59.0	17.0	1	1
50	53.0	65.0	12.0	1	0
51	70.0	68.0	0.0	0	0
52	38.0	66.0	11.0	0	0
53	50.0	63.0	1.0	0	0
54	61.0	64.0	0.0	0	0
55	61.0	65.0	0.0	1	0
56	43.0	59.0	2.0	1	0

Gambar 4.23 Hasil Klasifikasi *Naïve bayesian Konvensional (Haberman)*

4.2. Pengujian

Pengujian dilakukan menggunakan dataset yang diperoleh dari *KEEL-Dataset Repository* yakni *Haberman* dan satu set *data Water quality Status* dari penelitian Denades *et al*. (2016).

Pengujian ini bertujuan untuk melihat kinerja model klasifikasi *Weight naïve bayes* menggunakan pendekatan *Gain ratio* dengan *Naïve bayesian* konvensional.

4.2.1 Pengujian Terhadap Dataset Water quality Status

Dataset *Water quality Status* memiliki 8 atribut, 4 kelas dan 120 *instance*, Distribusi kelas berupa *good condition* (30 *instance*), *lightly polluted* (30 *instance*), *medium polluted* (30 *instance*) dan *heavily polluted* (30 *instance*). Tabel 4.1 menunjukkan informasi atribut dataset *Water quality Status*.

Tabel 4.1 Informasi Atribut Data Set *Water quality Status*

No.	Atribut	Nilai
1	*TSS (mg/L)*	[2-266]
2	*DO (mg/L)*	[0.02-8.43]
3	*COD (mg/L)*	[1.7-416]
4	*BOD (mg/L)*	[0.6-150]
5	*Total phospat (mg/L)*	[0.0016-1.23]
6	*Fecal Coliform (mg/L)*	[27-2800000]
7	*Total Coliform (mg/L)*	[74-53000000]
8	*Pij*	[0.54-15.31]
9	*Quality Status*	{*good condition, lightly polluted, medium polluted, heavily polluted*}

Adapun rincian dataset *Water quality Status* dapat dilihat pada Tabel 4.2.

Tabel 4.2 Rincian Data *Water quality Status*

ID	X1	X2	X3	X4	X5	X6	X7	X8	Class
1	2	4	8	2.6	0.1	92	150	0.76	1
2	3	4.5	19.2	3.1	0.14	92	150	0.88	1
3	3	4.4	16	2.9	0.12	930	2400	0.91	1
4	4	4.1	4.793	1.32	0.18	1100	1400	0.87	1
5	4	4.2	8	2.5	0.11	230	750	0.81	1
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
120	97	6.3	55.4	10	0.02	1500	2800000	10.74	4

Adapun rincian hasil *Data Preprocessing* dataset *Water quality Status* dapat dilihat pada Tabel 4.3.

Tabel 4.3 Hasil Data *Preprocessing Water quality Status*

No	TSS (mg/L)	DO (mg/L)	COD (mg/L)	*BOD (mg/L)*	*Total phospat (mg/L)*	…	Quality Status
1	2	4	8	2.6	0.1	…	*Good Condition*
2	3	4.5	19.2	3.1	0.14	…	*Good Condition*
3	3	4.4	16	2.9	0.12	…	*Good Condition*
4	4	4.1	4.793	1.32	0.18	…	*Good Condition*
5	4	4.2	8	2.5	0.11	…	*Good Condition*
:	:	:	:	:	:	:	:
117	97	6.3	55.4	10	0.02	…	*Heavily Polluted*

Pada tahapan selanjutnya adalah pemberian bobot terhadap atribut dari dataset berdasarkan *Gain ratio* dengan persamaan 3.1. Adapun hasil dari proses pembobotan atribut dilihat pada Tabel 4.4:

Tabel 4.4 Pembobotan Atribut Pada Data *Water quality Status*

Atribut	Gain Ratio	Bobot
TSS (mg/L)	0.441	0.247
DO (mg/L)	0.257	0
COD (mg/L)	0.325	0.091
BOD (mg/L)	0.488	0.310
Total phospat (mg/L)	0.274	0.023
Fecal Coliform (mg/L)	0.624	0.494
Total Coliform (mg/L)	1	1
Pij	1	1

Setelah bobot setiap atribut diperoleh kemudiaan dilakukan proses klasifikasi dengan menggunakan *Naïve bayesian*. Dataset *Water quality Status* memiliki 8 atribut 4 kelas

dan 117 *instances*. Data dibagi sebanyak 70% dijadikan data latih dan 30% dijadikan data uji.

Dalam proses penentuan nilai probabilitas dan hasil prediksi pada *Naïve bayesian* konvensional menggunakan persamaan 2.7, sedangkan penentuan nilai probabilitas dan hasil prediksi pada *Naïve bayesian* berdasarkan pendekatan *Weight naïve bayes* menggunakan persamaan 2.10. Tahapan pengujian dari model klasifikasi *Weight naïve bayes* (WNB) sama dengan tahapan model klasifikasi *Naïve bayesian* konvensional, perbedaan yang terjadi pada tahap *Summarize Dataset* yakni hasil perhitungan nilai Std.Deviasi karena telah diberikan nilai bobot berdasarkan Gain ratio per atribut. Berikut hasil dari mean dan Std. deviasi per atribut (Dataset *Water quality*) pada model klasifikasi *Weight naïve bayes*:

1) Hasil Mean & Std. Deviation per atribut WNB (*Summarize Dataset*):

Rekapitulasi Mean & STD Deviation Water per Atribut

	Mean	Std-Deviation	#Record
TSS	5.322650e+01	8.044664e+01	117
DO	4.612966e+00	3.063303e+00	117
COD	2.866099e+01	6.651889e+01	117
BOD	7.882222e+00	2.197678e+01	117
Total_Phospat	1.937111e-01	1.861268e+00	117
Fecal_Coliform	5.843452e+04	5.152524e+05	117
Total_Coliform	4.243984e+06	1.545126e+07	117
Pij	5.750427e+00	7.639516e+00	117

Gambar 4.24 Hasil Mean & Std. Deviation per atribut WNB (Dataset *Water quality)*

Berdasarkan Gambar 4.24, terlihat bahwa atribut *TSS* memiliki nilai rata-rata (Mean) yang sama dengan model klasifikasi Naïve bayesian konvensional yakni sebesar 5.322, Namun nilai Std.Deviasi adalah sebesar 8.044 dengan jumlah record sebanyak 117 *instances*. Atribut *DO* memiliki nilai Mean yang sama dengan model klasifikasi Naïve bayesian konvensional yakni sebesar 4.613, Namun nilai Std.Deviation adalah sebesar 3.063 dengan jumlah record juga sebanyak 117 *instances.* Hal yang sama juga berlaku untuk atribut berikutnya.

2) Hasil Mean & Std. Deviation per kelas (*Summarize by Class*):

```
 Rekapitulasi Mean & STD Deviation Water per Class 0
-----------------------------------------------------
(14.966666666666667, 17.488538163764776, 30)
(4.218333333333332, 2.061670745302118, 30)
(11.090866666666667, 9.338243170328488, 30)
(2.4473333333333334, 2.2818960880361643, 30)
(0.11413999999999998, 1.690730971959848, 30)
(374.56666666666666, 417.8940518900268, 30)
(967.0, 962.850852869297, 30)
(0.8340000000000002, 1.7376116978918366, 30)

 Rekapitulasi Mean & STD Deviation Water per Class 1
-----------------------------------------------------
(99.83333333333333, 111.77822361895863, 30)
(4.702333333333334, 2.650542231924411, 30)
(22.918666666666663, 48.17178913071249, 30)
(3.584, 3.482849024085815, 30)
(0.1866666666666662, 1.8547379013715577, 30)
(2357.5, 4447.066963509594, 30)
(8970.8, 14249.262902184644, 30)
(2.437666666666666, 2.731089520593781, 30)

 Rekapitulasi Mean & STD Deviation Water per Class 2
-----------------------------------------------------
(54.75925925925926, 67.17608099839441, 27)
(5.111851851851852, 3.7620543715017067, 27)
(50.72037037037036, 118.73141684122879, 27)
(16.925185185185185, 41.69681750114971, 27)
(0.28925925925925927, 2.0385166593458246, 27)
(17080.62962962963, 42475.645013204914, 27)
(438814.81481481483, 677833.6862333213, 27)
(7.368888888888888, 3.3748189201486705, 27)

 Rekapitulasi Mean & STD Deviation Water per Class 3
-----------------------------------------------------
(43.5, 35.73573767624847, 30)
(4.4692333333333325, 3.407880367278833, 30)
(32.12, 31.648822397803112, 30)
(9.476666666666667, 9.379209948476307, 30)
(0.1943333333333333, 1.9117848488644507, 30)
(209790.0, 998118.6342654163, 30)
(16146666.666666666, 23666817.812159948, 30)
(12.522999999999998, 3.8384532364206256, 30)
```

Gambar 4.25 Hasil Mean & Std. Deviation per kelas WNB (Dataset *Water quality)*

Dari Gambar 4.25, jika diperhatikan sebanyak 30 record atribut *TSS* termasuk kedalam kelas 0 bernilai Mean sebesar 14.97 dan Std. Deviasi sebesar 17.49. Sedangkan sebanyak 30 record atribut *TSS* lainnya termasuk kedalam kelas 1 bernilai Mean sebesar 99.83 dan Std. Deviasi sebesar 111.79. Sementara itu sebanyak 27 record atribut *TSS* berikutnya termasuk kedalam kelas 2 bernilai Mean sebesar 54.76 dan Std. Deviasi sebesar 67.18. Sedangkan 30 record atribut *TSS* sisanya termasuk

kedalam kelas 3 memiliki nilai Mean sebesar 43.5 dan Std. Deviasi sebesar 35.74. Hal yang sama juga berlaku untuk atribut lainnya yang termasuk pada dataset *Water*.

3) Hasil Perolehan Nilai Probabilitas per kelas (*Probabilities by Class*):

```
 Nilai Probabilitas Water per Class
-----------------------------------
{0: 7.168483116074066e-14, 1: 3.728875326660344e-18, 2: 4.0679474027017673e-23,
3: 5.356712721538003e-26}
```

Gambar 4.26 Hasil Perolehan Probabilitas per kelas WNB (Dataset *Water quality)*

Merujuk pada Gambar 4.26, jika diperhatikan nilai probabilitas yang tertinggi ada pada kelas 0 yakni *Good Condition* memiliki probabilitas terbesar yakni 7.17 terhadap dataset *Water quality*, dilanjutkan dengan probabilitas kelas 3 sebesar 5.36, kelas 2 sebesar 4.07 dan kelas 1 sebesar 3.73. Sehingga nilai probabilitas kelas *Good Condition* disebut *Maximum a Posteriori Hypothesis* (MAP) karena memiliki nilai probabilitas kelas tertinggi.

4) Hasil Pembagian data (*Train Test Splitting*)

Rekapitulasi Dataset Water (Data Latih)

	TSS	DO	COD	...	Total_Coliform	Pij	Aktual-Class
0	89.0	4.90	48.00	...	4600.0	1.99	1
1	30.0	2.05	416.00	...	35000.0	7.11	2
2	89.0	6.87	26.35	...	9300000.0	12.58	3
3	22.0	2.90	56.50	...	11000000.0	13.02	3
4	38.0	7.50	26.64	...	42000000.0	15.07	3
..	...	...	...	...	...	...	...
77	26.0	3.60	24.00	...	1500.0	0.81	0
78	39.0	3.80	16.21	...	35000000.0	14.93	3
79	75.0	4.34	64.76	...	2100000.0	10.49	3
80	29.0	4.10	16.00	...	930.0	0.85	0
81	11.0	4.25	16.00	...	1500.0	0.84	0

[82 rows x 9 columns]

Gambar 4.27 Data Latih *Water quality (Weight naïve bayes)*

Pada Gambar 4.27, terlihat uraian tentang rekapitulasi data latih (*training data*) sebanyak 82 *instances* berserta kategori class dari 117 *instances* yang diperoleh melalui 70% dari data dijadikan sebagai data latih dataset *Water quality*.

Rekapitulasi Dataset Water (Data Uji)

	TSS	DO	COD	...	Total_Coliform	Pij	Aktual-Class
0	4.0	4.20	8.00	...	750.0	0.81	0
1	6.0	4.40	32.00	...	740.0	0.99	0
2	6.0	4.60	16.00	...	740.0	0.88	0
3	9.0	4.00	8.00	...	350.0	0.96	0
4	7.0	4.00	8.00	...	1500.0	0.90	0
5	12.8	3.97	5.52	...	150.0	0.54	0
6	13.0	4.00	12.00	...	430.0	0.78	0
7	15.0	4.30	12.80	...	1500.0	0.91	0
8	17.0	6.03	5.62	...	1100.0	0.63	0
9	20.0	4.20	8.00	...	1500.0	0.85	0
10	33.0	4.20	8.00	...	930.0	0.85	0
11	30.0	4.60	8.00	...	280.0	0.95	0
12	7.0	5.30	192.20	...	11000.0	2.28	1
13	15.0	4.10	18.00	...	11000.0	1.73	1
14	19.0	7.50	8.87	...	35000.0	3.78	1
15	20.0	4.10	19.20	...	11000.0	3.40	1
16	60.0	5.20	3.20	...	2400.0	1.04	1
17	75.0	4.10	6.40	...	2100.0	1.18	1
18	159.0	4.50	32.00	...	2000.0	2.41	1
19	184.0	4.80	12.80	...	4600.0	2.80	1
20	188.0	4.40	8.00	...	4600.0	2.77	1
21	16.0	7.68	10.07	...	140000.0	6.05	2
22	19.0	7.17	5.00	...	350000.0	7.39	2
23	22.0	6.45	16.39	...	280000.0	8.77	2
24	25.0	5.80	21.26	...	1100000.0	9.26	2
25	57.0	2.65	43.02	...	1500000.0	9.88	2
26	58.0	6.06	2.79	...	440000.0	7.85	2
27	67.0	6.18	4.80	...	210000.0	6.57	2
28	72.0	6.02	6.40	...	420000.0	7.21	2
29	76.0	6.04	33.66	...	1500000.0	9.92	2
30	128.0	3.70	85.46	...	110000.0	7.40	2
31	208.0	3.50	176.31	...	29000.0	6.17	2
32	26.0	3.89	36.16	...	2100000.0	10.36	3
33	42.0	8.43	26.69	...	21000000.0	13.91	3
34	44.0	4.10	20.98	...	28000000.0	14.35	3

Gambar 4.28 Data Uji *Water quality (Weight naïve bayes)*

Pada Gambar 4.28, terlihat uraian tentang rekapitulasi data uji (*testing data*) sebanyak 35 *instances* berserta kategori class dari 117 *instances* yang diperoleh melalui 30% dari data dijadikan sebagai data uji dataset *Water quality*. Data uji ini digunakan untuk pengujian model klasifikasi *Weight naïve bayes* yang kemudian menghasilkan hasil prediksi kelas serta diperoleh hasil akurasi pengklasifikasian model klasifikasi *Weight naïve bayes* yang diusulkan.

5) Hasil Akurasi Dataset *Water quality* (*Weight naïve bayes*)

```
Nilai Akurasi Weighting Naive-Bayes Classifier adalah  88.57142857142857 %

 Confusion Matrix
------------------
[[12  0  0  0]
 [ 2  6  1  0]
 [ 0  0 10  1]
 [ 0  0  0  3]]
```

Gambar 4.29 Confusion matrix *Water quality* (*Weight naïve bayes*)

Pada Gambar 4.29, terlihat output dari nilai akurasi yang dihasilkan oleh model *Weight naïve bayes* menggunakan dataset *Water quality*. Sementara hasil kinerja dan performa pengklasifikasian *Weight naïve bayes* menggunakan metode *Confusion matrix* didasarkan sub-bab 3 yakni pada Tabel 3.3 merujuk pada persamaan 3.2 Berikut adalah tabel perhitungannya:

Tabel 4.5 *Confusion matrix* model *Weight naïve bayes (Water quality)*

Kinerja Klasifikasi	**Predicted Class**			
Actual Class	Predicted Lightly Polluted (0)	Predicted Heavily Polluted (1)	Predicted Medium Polluted (2)	Predicted Good Condition (3)
Actual. Lightly Polluted (0)	12	0	0	0
Actual. Heavily Polluted (1)	2	6	1	0
Actual. Medium Polluted (2)	0	0	10	1
Actual. Good Condition (3)	0	0	0	3

Berdasarkan Tabel 4.5, menghitung nilai *Accuracy* dan tingkat error pengklasifikasian (*Classification_error*) dari model *Weight naïve bayes* menggunakan dataset *Water quality*, Berikut adalah hasil perhitungannya:

1) *Accuracy* $= \frac{12+6+10+3}{12+6+10+3+2+1+1} = \frac{31}{35}$ = 0.8571*100% = **88.57%**

2) *Classification_error* = $\frac{2+1+1}{12+6+10+3+2+1+1} = \frac{4}{35}$ = 0.1143*100% = **11.43%**

Pada Gambar 4.30 merupakan hasil rekapitulasi prediksi model klasifikasi *Weight naïve bayes* terhadap dataset *Water quality:*

Rekap Hasil Klasifikasi Weighting Naive-Bayes Classifier (Data Uji Water)

	TSS	DO	COD	...	Pij	Aktual-Class	Prediksi-Class
0	4.0	4.20	8.00	...	0.81	0	0
1	6.0	4.40	32.00	...	0.99	0	0
2	6.0	4.60	16.00	...	0.88	0	0
3	9.0	4.00	8.00	...	0.96	0	0
4	7.0	4.00	8.00	...	0.90	0	0
5	12.8	3.97	5.52	...	0.54	0	0
6	13.0	4.00	12.00	...	0.78	0	0
7	15.0	4.30	12.80	...	0.91	0	0
8	17.0	6.03	5.62	...	0.63	0	0
9	20.0	4.20	8.00	...	0.85	0	0
10	33.0	4.20	8.00	...	0.85	0	0
11	30.0	4.60	8.00	...	0.95	0	0
12	7.0	5.30	192.20	...	2.28	1	2
13	15.0	4.10	18.00	...	1.73	1	1
14	19.0	7.50	8.87	...	3.78	1	1
15	20.0	4.10	19.20	...	3.40	1	1
16	60.0	5.20	3.20	...	1.04	1	0
17	75.0	4.10	6.40	...	1.18	1	0
18	159.0	4.50	32.00	...	2.41	1	1
19	184.0	4.80	12.80	...	2.80	1	1
20	188.0	4.40	8.00	...	2.77	1	1
21	16.0	7.68	10.07	...	6.05	2	2
22	19.0	7.17	5.00	...	7.39	2	2
23	22.0	6.45	16.39	...	8.77	2	3
24	25.0	5.80	21.26	...	9.26	2	2
25	57.0	2.65	43.02	...	9.88	2	2
26	58.0	6.06	2.79	...	7.85	2	2
27	67.0	6.18	4.80	...	6.57	2	2
28	72.0	6.02	6.40	...	7.21	2	2
29	76.0	6.04	33.66	...	9.92	2	2
30	128.0	3.70	85.46	...	7.40	2	2
31	208.0	3.50	176.31	...	6.17	2	2
32	26.0	3.89	36.16	...	10.36	3	3
33	42.0	8.43	26.69	...	13.91	3	3
34	44.0	4.10	20.98	...	14.35	3	3

[35 rows x 10 columns]

Gambar 4.30 Hasil Klasifikasi *Weight naïve bayes (Water quality)*

4.2.2 Pengujian Terhadap Dataset Haberman

Dataset *Haberman* memiliki 3 atribut, 2 kelas dan 306 *instances*, Distribusi kelas berupa *positive* (81 *instances*) dan *negative* (225 *instances*). Tabel 4.6 menunjukkan informasi atribut dataset *Haberman.*

Tabel 4.6 Informasi Atribut Data Set *Haberman*

No.	Atribut	Nilai
1	*Age*	[30, 83]
2	*Year*	[58, 69]
3	*Positive*	[0, 52]
4	*Survival*	{positive, negative}

Adapun rincian dataset *Haberman* dapat dilihat pada Tabel 4.7 berikut:

Tabel 4.7 Rincian Data *Haberman*

ID	Age	Year	Positive	Survival
1	38	59	2	Negative
2	39	63	4	Negative
3	49	62	1	Negative
4	53	60	2	Negative
5	47	68	4	Negative
6	56	67	0	Negative
7	64	58	0	Negative
⋮	⋮	⋮	⋮	⋮
306	52	66	4	Positive

Sumber : *https://sci2s.ugr.es/keel/dataset.php?cod=62*

Adapun rincian hasil *Data Preprocessing* dataset *Haberman* dapat dilihat pada Tabel 4.8 berikut:

Tabel 4.8 Hasil Data *Preprocessing Haberman*

ID	Age	Year	Positive	Survival
1	38	59	2	Negative
2	39	63	4	Negative
3	49	62	1	Negative
4	53	60	2	Negative
5	47	68	4	Negative
6	56	67	0	Negative
⋮	⋮	⋮	⋮	⋮
289	52	66	4	Positive

Pada tahapan selanjutnya adalah pemberian bobot terhadap atribut dari dataset berdasarkan *Gain ratio* dengan persamaan 3.1. Adapun hasil dari proses pembobotan atribut dapat dilihat pada Tabel 4.9:

Tabel 4.9 Pembobotan Atribut Pada Data *Haberman*

Atribut	Gain ratio	Bobot
Age	0.219	1
Year	0.005	0
Positive	0.195	0.9

Setelah bobot setiap atribut diperoleh kemudiaan dilakukan proses klasifikasi dengan menggunakan *Naïve bayesian*. Dataset *Haberman* memiliki 3 atribut, 2 kelas dan 289 *instances*. Data dibagi sebanyak 80% dijadikan data latih dan 20% dijadikan data uji..

Dalam proses penentuan nilai probabilitas dan hasil prediksi pada *Naïve bayesian* konvensional menggunakan persamaan 2.4, sedangkan penentuan nilai probabilitas

dan hasil prediksi pada *Naïve bayesian* berdasarkan pendekatan *Weight naïve bayes* menggunakan persamaan 2.11. Tahapan pengujian dari model klasifikasi *Weight naïve bayes* (WNB) sama dengan tahapan model klasifikasi *Naïve bayesian* konvensional, perbedaan yang terjadi pada tahap *Summarize Dataset* yakni hasil perhitungan nilai Std.Deviasi karena telah diberikan nilai bobot berdasarkan Gain ratio per atribut. Berikut hasil dari mean dan Std. deviasi per atribut (Dataset *Haberman*) pada model klasifikasi *Weight naïve bayes* :

1) Hasil Mean & Std. Deviation per atribut WNB (*Summarize Dataset*):

```
 Rekapitulasi Mean & STD Deviation Haberman per Atribut
-------------------------------------------------------
                Mean  Std-Deviation   #Record
Age        52.422145      13.600421       289
Year       62.906574       4.429721       289
Positive    4.221453       9.173260       289
```

Gambar 4.31 Hasil Mean & Std. Deviation per atribut WNB (Dataset *Haberman)*

Berdasarkan Gambar 4.31, terlihat bahwa atribut *Age* memiliki nilai rata-rata *(Mean*) yang sama dengan model klasifikasi Naïve bayesian konvensional yakni sebesar 52.42, Namun nilai Std.Deviasi adalah sebesar 13.60 dengan jumlah record sebanyak 289 *instances*. Atribut *Year* memiliki nilai Mean yang sama dengan model klasifikasi Naïve bayesian konvensional yakni sebesar 62.91, Namun nilai Std.Deviation adalah sebesar 4.43 dengan jumlah *record* juga sebanyak 289 *instances.* Hal yang sama juga berlaku untuk atribut *Positive* memiliki nilai Mean yang sama, Namun nilai Std.Deviation adalah sebesar 9.17 dengan jumlah record juga sebanyak 289 *instances.*

2) Hasil Mean & Std. Deviation per kelas (*Summarize by Class*):

```
 Rekapitulasi Mean & STD Deviation Haberman per Class 0
-------------------------------------------------------
(51.871428571428574, 13.868931304048358, 210)
(62.91904761904762, 4.411984409454628, 210)
(2.9857142857142858, 7.582166012027, 210)

 Rekapitulasi Mean & STD Deviation Haberman per Class 1
-------------------------------------------------------
(53.88607594936709, 12.782427246182516, 79)
(62.87341772151899, 4.5048695724667285, 79)
(7.506329113924051, 11.551077689259753, 79)
```

Gambar 4.32 Hasil Mean & Std. Deviation per kelas WNB (Dataset *Haberman)*

Merujuk pada Gambar 4.32, jika diperhatikan sebanyak 210 record atribut *Age* termasuk kedalam kelas 0 bernilai Mean sebesar 51.87 dan Std. Deviasi sebesar 13.87. Sedangkan sebanyak 79 record atribut *Age* lainnya termasuk kedalam kelas 1 bernilai Mean sebesar 53.89 dan Std. Deviasi sebesar 12.78. Hal yang sama juga berlaku untuk atribut lainnya yang termasuk pada dataset *Haberman.*

3) Hasil Perolehan Nilai Probabilitas per kelas (*Probabilities by Class*):

```
 Nilai Probabilitas Haberman per Class
----------------------------------------
{0: 4.0304206393319847e-05, 1: 7.434655707266219e-06}
```

Gambar 4.33 Hasil Perolehan Probabilitas per kelas WNB (Dataset *Haberman*)

Merujuk pada Gambar 4.33, jika diperhatikan nilai probabilitas tertinggi ada pada kelas 1 yakni *Positive* memiliki probabilitas terbesar yakni 7.43 terhadap dataset *Haberman,* Sementara nilai probabilitas kelas 0 hanya sebesar 4.03. Dengan demikian nilai probabilitas kelas *Positive* disebut *Maximum a Posteriori Hypothesis* (MAP) karena memiliki nilai probabilitas kelas tertinggi.

4) Hasil Pembagian data (*Train Test Splitting*)

Rekapitulasi Dataset Haberman (Data Latih)

	Age	Year	Positive	Aktual-Status
0	41.0	58.0	0.0	0
1	42.0	69.0	1.0	1
2	61.0	62.0	5.0	1
3	61.0	68.0	0.0	0
4	50.0	61.0	0.0	0
..	...	...	...	...
227	34.0	60.0	1.0	0
228	60.0	61.0	25.0	0
229	59.0	62.0	35.0	1
230	49.0	64.0	10.0	1
231	69.0	65.0	0.0	0

[232 rows x 4 columns]

Gambar 4.34 Data Latih *Haberman (Weight naïve bayes)*

Merujuk Gambar 4.34, terlihat uraian tentang rekapitulasi data latih (*training data*) sebanyak 232 *instances* berserta kategori kelas dari 289 *instances* yang diperoleh melalui 80% sebagai data latih dataset *Haberman.* Data latih ini digunakan untuk pembentukan model klasifikasi *Weight naïve bayes*.

Rekapitulasi Dataset Haberman (Data Uji)

	Age	Year	Positive	Aktual-Status
0	38.0	59.0	2.0	0
1	61.0	65.0	8.0	0
2	55.0	58.0	1.0	0
3	43.0	63.0	14.0	0
4	59.0	64.0	4.0	0
5	63.0	61.0	0.0	0
6	54.0	59.0	7.0	0
7	67.0	66.0	0.0	0
8	56.0	65.0	9.0	1
9	70.0	59.0	8.0	0
10	71.0	68.0	2.0	0
11	41.0	65.0	0.0	0
12	30.0	62.0	3.0	0
13	59.0	64.0	7.0	0
14	67.0	65.0	0.0	0
15	39.0	59.0	2.0	0
16	65.0	61.0	2.0	1
17	54.0	65.0	5.0	1
18	50.0	59.0	0.0	0
19	41.0	64.0	0.0	1
20	44.0	67.0	16.0	0
21	38.0	62.0	3.0	0
22	52.0	59.0	2.0	1
23	48.0	64.0	0.0	0
24	34.0	67.0	7.0	0
25	53.0	63.0	0.0	0
26	58.0	58.0	3.0	0
27	49.0	62.0	0.0	0
28	37.0	60.0	15.0	0
29	37.0	60.0	0.0	0
30	53.0	60.0	1.0	0
31	39.0	67.0	0.0	0
32	40.0	58.0	0.0	0
33	59.0	64.0	0.0	0
34	64.0	61.0	0.0	0
35	45.0	59.0	14.0	0
36	48.0	66.0	0.0	0
37	72.0	58.0	0.0	0
38	47.0	61.0	0.0	0
39	47.0	63.0	23.0	1
40	70.0	67.0	0.0	0
41	41.0	64.0	0.0	0
42	47.0	62.0	0.0	1
43	52.0	62.0	1.0	0
44	44.0	64.0	6.0	1
45	65.0	67.0	1.0	0
46	68.0	68.0	0.0	0
47	59.0	63.0	0.0	0
48	45.0	65.0	6.0	1
49	60.0	59.0	17.0	1
50	53.0	65.0	12.0	1
51	70.0	68.0	0.0	0
52	38.0	66.0	11.0	0
53	50.0	63.0	1.0	0
54	61.0	64.0	0.0	0
55	61.0	65.0	0.0	1
56	43.0	59.0	2.0	1

Gambar 4.35b Data Uji *Haberman (Weight naïve bayes)*

Pada Gambar 4.35a & 4.35b, terlihat uraian tentang rekapitulasi data uji (*testing data*) sebanyak 57 *instances* dari 289 *instances* yang diperoleh melalui 30% dari data dijadikan sebagai data uji dataset *Haberman.* Data uji ini digunakan untuk pengujian model klasifikasi *Weight naïve bayes*.

5) Hasil Akurasi Dataset *Haberman* (*Weight naïve bayes*)

```
Nilai Akurasi Weighting Naive-Bayes Classifier adalah  80.7017543859649 %

 Confusion Matrix
------------------
[[44  0]
 [11  2]]
```

Gambar 4.36 Confusion matrix *Haberman* (*Weight naïve bayes*)

Pada Gambar 4.36, terlihat output dari nilai akurasi yang dihasilkan oleh model *Weight naïve bayes* menggunakan dataset *Haberman*. Sementara hasil kinerja dan performa pengklasifikasian *Weight naïve bayes* menggunakan metode *Confusion matrix* didasarkan sub-bab 3 yakni pada Tabel 3.3 merujuk pada persamaan 3.2 Berikut adalah tabel perhitungannya:

Tabel 4.10 *Confusion matrix* model *Weight naïve bayes (Haberman)*

Kinerja Klasifikasi	**Predicted Class**	
Actual Class	Predicted Negative (0)	Predicted Positive (1)
Actual. Negative (0)	44	0
Actual. Positive (1)	11	2

Terlihat pada Tabel 4.10, perhitungan nilai *Accuracy* dan tingkat error pengklasifikasian (*Classification_error*) dari model *Weight naïve bayes* menggunakan dataset *Haberman*, Berikut hasil perhitungannya:

1) *Accuracy* $= \frac{44+2}{44+2+11} = \frac{46}{57}$ = 0.8070*100% = **80.70%**

2) *Classification_error* = $\frac{11}{44+2+11} = \frac{11}{57}$ = 0.1143*100% = **19.30**

Berikut pada Gambar 4.37 merupakan hasil rekapitulasi prediksi model klasifikasi *Weight naïve bayes* terhadap dataset *Haberman:*

Rekap Klasifikasi Weighting Naive-Bayes Classifier (Data Uji Haberman)

	Age	Year	Positive	Aktual-Status	Prediksi-Class
0	38.0	59.0	2.0	0	0
1	61.0	65.0	8.0	0	0
2	55.0	58.0	1.0	0	0
3	43.0	63.0	14.0	0	0
4	59.0	64.0	4.0	0	0
5	63.0	61.0	0.0	0	0
6	54.0	59.0	7.0	0	0
7	67.0	66.0	0.0	0	0
8	56.0	65.0	9.0	1	0
9	70.0	59.0	8.0	0	0
10	71.0	68.0	2.0	0	0
11	41.0	65.0	0.0	0	0
12	30.0	62.0	3.0	0	0
13	59.0	64.0	7.0	0	0
14	67.0	65.0	0.0	0	0
15	39.0	59.0	2.0	0	0
16	65.0	61.0	2.0	1	0
17	54.0	65.0	5.0	1	0
18	50.0	59.0	0.0	0	0
19	41.0	64.0	0.0	1	0
20	44.0	67.0	16.0	0	0
21	38.0	62.0	3.0	0	0
22	52.0	59.0	2.0	1	0
23	48.0	64.0	0.0	0	0
24	34.0	67.0	7.0	0	0
25	53.0	63.0	0.0	0	0
26	58.0	58.0	3.0	0	0
27	49.0	62.0	0.0	0	0
28	37.0	60.0	15.0	0	0
29	37.0	60.0	0.0	0	0
30	53.0	60.0	1.0	0	0
31	39.0	67.0	0.0	0	0
32	40.0	58.0	0.0	0	0
33	59.0	64.0	0.0	0	0
34	64.0	61.0	0.0	0	0
35	45.0	59.0	14.0	0	0
36	48.0	66.0	0.0	0	0
37	72.0	58.0	0.0	0	0
38	47.0	61.0	0.0	0	0
39	47.0	63.0	23.0	1	1
40	70.0	67.0	0.0	0	0
41	41.0	64.0	0.0	0	0
42	47.0	62.0	0.0	1	0
43	52.0	62.0	1.0	0	0
44	44.0	64.0	6.0	1	0
45	65.0	67.0	1.0	0	0
46	68.0	68.0	0.0	0	0
47	59.0	63.0	0.0	0	0
48	45.0	65.0	6.0	1	0
49	60.0	59.0	17.0	1	1
50	53.0	65.0	12.0	1	0
51	70.0	68.0	0.0	0	0
52	38.0	66.0	11.0	0	0
53	50.0	63.0	1.0	0	0
54	61.0	64.0	0.0	0	0
55	61.0	65.0	0.0	1	0
56	43.0	59.0	2.0	1	0

Gambar 4.37 Hasil Klasifikasi *Weight naïve bayes* (*Haberman*)

4.3. Pembahasan

Tahapan proses yang dilakukan pada penelitian ini, setelah proses pembobotan setiap atribut diperoleh menggunakan *Gain ratio*, maka tahapan selanjutnya adalah melakukan klasifikasi menggunakan metode Naïve bayes, dimana setiap atributnya diberikan bobot umumnya disebut dengan Weight naïve bayes (WNB), pada penelitian ini metode tersebut telah mampu mengurangi pengaruh dari atribut yang tidak relavan terhadap kelas data sehingga dapat mempengaruhi nilai akurasi.

Guna menjelaskan nilai akurasi yang diperoleh dari model klasifikasi *Naïve bayes* Konvensional dan model klasifikasi *Weight naïve bayes (Gain ratio)* terhadap dataset *Water quality*, maka dilihat dari hasil pengukuran ketepatan memprediksi masing-masing model klasifikasi yang dapat diuraikan seperti pada tabel berikut:

Tabel 4.11 Hasil Prediksi Dataset *Water quality*

No.	Quality Status	Naïve bayes Konvensional		Weight naïve bayes (Gain ratio)	
		True Predicted	False Predicted	True Predicted	False Predicted
1	Good Condition	10	2	12	0
2	Lightly Polluted	8	1	6	3
3	Medium Polluted	9	2	10	1
4	Heavily Polluted	3	0	3	0
Jumlah Record		**30**	**5**	**31**	**4**

Dari Tabel 4.11, dapat dilihat bahwa hasil ketepatan memprediksi dari model klasifikasi *Weight naïve bayes* memberikan jumlah ketepatan prediksi dengan benar (*True Predicted*) yang lebih baik daripada model klasifikasi *Naïve bayes* Konvensional dimana peningkatan jumlah *True Predicted* didapati pada *Quality Status* dengan label *Good Condition* sejumlah 2 record dan *Medium Polluted* sejumlah 1 record, Sehingga peningkatan jumlah ketepatan prediksi dengan benar didapati pada model klasifikasi *Weight naïve bayes.*

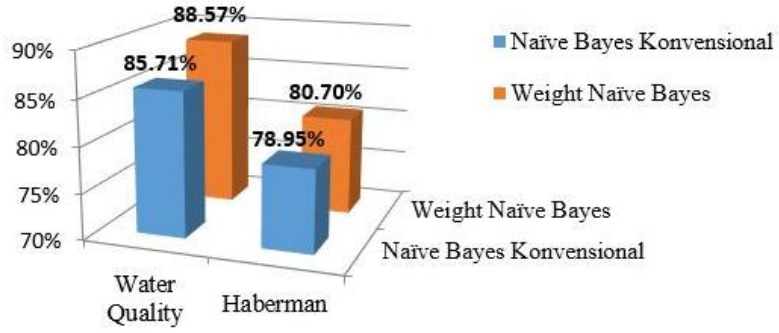
Berdasarkan kemampuan memprediksi dari model klasifikasi *Weight naïve bayes* pada Tabel 4.11, maka dapat diperoleh nilai akurasi seperti pada Gambar 4.29.
Kemudian untuk menjelaskan nilai akurasi yang diperoleh dari model klasifikasi *Naïve bayes* Konvensional dan model klasifikasi *Weight naïve bayes (Gain ratio)* terhadap dataset *Haberman*, maka juga dilihat dari hasil pengukuran ketepatan memprediksi masing-masing model klasifikasi yang dapat diuraikan seperti pada tabel berikut:

Tabel 4.12 Hasil Prediksi Dataset *Haberman*

No.	Status	Naïve bayes Konvensional		Weight naïve bayes (Gain ratio)	
		True Predicted	False Predicted	True Predicted	False Predicted
1	Negative	43	1	44	0
2	Positive	2	11	2	11
Jumlah Record		**45**	**12**	**46**	**11**

Dari Tabel 4.12, dapat dilihat bahwa hasil ketepatan memprediksi dari model klasifikasi *Weight naïve bayes* juga memberikan jumlah ketepatan prediksi dengan benar (*True Predicted*) yang lebih baik daripada model klasifikasi *Naïve bayes* Konvensional dimana peningkatan jumlah *True Predicted* didapati pada *Status* dengan label *Negative* sejumlah 1 record, Sehingga peningkatan jumlah ketepatan prediksi dengan benar didapati pada model klasifikasi *Weight Naïve bayes.* Berdasarkan kemampuan memprediksi dari model klasifikasi *Weight naïve bayes* pada Tabel 4.12, maka dapat diperoleh nilai akurasi seperti pada gambar 4.36.

Untuk melihat dengan jelas nilai akurasi yang didapati dari kedua model klasifikasi terhadap seluruh data yang digunakan dalam pengujian, maka dapat dilihat pada grafik berikut:



Gambar 4.38 Grafik Perbandingan Akurasi Model Klasifikasi

Dari grafik pada Gambar 4.38, terlihat bahwa model klasifikasi *Weight naïve bayes* dapat memberikan nilai akurasi yang lebih baik dari pada model klasifikasi *Naïve bayes* Konvensional jika diterapkan pada kedua dataset yakni *Water quality* dan *Haberman*. Dimana peningkatan nilai akurasi tertinggi didapati pada dataset *Water quality* yaitu sebesar 2.86%, Sedangkan peningkatan nilai akurasi terendah didapati pada dataset *Haberman* yaitu sebesar 1.75%. Adapun peningkatan rata-rata nilai akurasi dari kedua dataset yang digunakan adalah sebesar 2.3%.

Berdasarkan pengujian yang telah dilakukan pada seluruh data, terlihat bahwa model klasifikasi *Weight naïve bayes* dapat memberikan nilai akurasi lebih baik di karenakan ada perubahan pembobotan pada nilai atribut pada dataset yang digunakan. Nilai dari pembobotan *Gain ratio* yang dugunakan untuk menghitung probabilitas

dalam *Naïve bayes*, dimana sebagai parameter untuk melihat hubungan antara setiap atribut pada data, dan dijadikan dasar dalam pembobotan setiap atribut dari dataset. Semakin tinggi *Gain ratio* suatu atribut maka semakin besar hubungan terhadap kelas data. Sehingga nilai akurasi meningkat dari pada nilai akurasi yang dihasilkan oleh model klasifikasi *Naïve bayes* Konvensional. Meningkatnya akurasi pada model klasifikasi *Naïve bayes* dikarenakan adanya jumlah ketepatan bobot dari seleksi atribut pada *Gain ratio*.

BAB 5

KESIMPULAN DAN SARAN

4.1. Kesimpulan

Pada penelitian yang telah dilakukan, maka penulis menghasilkan beberapa kesimpulan sebagai berikut:

1. Pada penelitian yang dilakukan mengenai dataset *Water quality* dan dataset *Haberman* menghasilkan prediksi dari klasifikasi *Naïve bayesian* konvensional dan klasifikasi *Weight naïve bayes*.
2. Keberhasilan dalam memprediksi menggunakan *Weight naïve bayes* mampu memberikan nilai akurasi yang lebih baik dari *Naïve bayesian* konvensional. Dimana peningkatan nilai akurasi dari dataset *Water quality* adalah 88.57%, sedangkan nilai akurasi dari dataset *Haberman* adalah 80.70% pada pada klasifikasi *Weight naïve bayes* .
3. Peningkatan nilai akurasi dari klasifikasi *Weight naïve bayes* pada dataset *Water quality* adalah sebesar 2.9%. Sedangkan peningkatan nilai akurasi pada dataset *Haberman* adalah sebesar 1.8%. Jika dilakukan rata-rata nilai akurasi dari masing-masing dataset menggunakan klasifikasi *Weight naïve bayes* adalah 2.3%.
4. Model klasifikasi *Weight naïve bayes* mampu memberikan nilai akurasi lebih baik dari yang dihasilkan oleh klasifikasi *Naïve bayesian* konvensional.

5.1. Saran

Pada penelitian selanjutnya diharapkan penulis adalah mengembangkan sistem program dalam memprediksi data yang lebih besar, karena masih ada kekurangan dalam penelitian ini sehingga harus disempurnakan dalam penelitian kedepannya dapat memperoleh hasil lebih baik dari sebelumnya. Penulis mengharapkan penelitian ini dilanjutkan dengan menggunakan algoritma lainnya. Semoga mendapatkan keakuratan yang lebih besar serta menghasilkan konsep prediksi yang lebih baik.

DAFTAR PUSTAKA

Amra, Ihsan & Maghari, Ashraf. 2017. Students Performance Prediction using KNN and Naïve bayesian. *8th International Conference on Information Technology (ICIT)*. **pp.** 909-913.

Danades, A., Pratama, D., Anggraini, D., Anggriani, D. 2016. Comparison of Accuracy Level K-Nearest Neighbor Algorithm and Support Vector Machine Algorithm in Classification Water quality Status. *International Conference on System Engineering and Technology*, pp. 137-141.

Duan, Weil & Lu, Xiang. 2010. Weighted Naie Bayesian Classifier Model Based on Information gain. *International Conference on Intelligent System Design and Engineering Application*. **pp.** 819-822.

Han, Xiaoyan. Xu, Liancheng., Ren, Min & Gu, Weiping. 2015. A Naïve bayesian Network Intrusion Detection Algorithm Based on Principal Component Analysis. *7th International Conference on Information Technology in Medicine and Education*. **pp.** 325-328.

Han, J., Kamber, M. & Pei, J. 2012. *Data Mining: Concepts and Techniques*. 3rd Edition. Morgan Kaufmann Publishers: San Francisco.

Jolliffe, I.T. 2002. *Principal Component Analysis*. 2nd Edition. Springer-Verlag: New York.

Johnson, W.A. & Wichern, D.W. 2007. *Applied Multivariate Statistical Analysis*. 6th Edition. Pearson Prentice Hall: New Jersey.

Kotu, V. & Deshpande, B. 2015. *Predictive Analytics and Data Mining*. Morgan Kaufmann Publisher: San Francisco.

Mao, Xin, Zhao, Gang & Sun, Ruoying. 2017. Naïve bayesian Algorithm Classification Model with Local Attribute Weighted based on KNN. **pp.**

Repaka, A. N., Ravikanti, S. D. & Franklin, R. 2019. Design and Implementing Heart Disease Prediction using Naïve bayesian. *International Conference on Trends in Electronics and Informatics (ICOEI)*. **pp.** 292-297.

Wang, Xingang & Sun, Xiu. 2016. An Improved Weighted Naïve bayesian Classification Algorithm based on Multivariable Linear Regression Model. *9th International Symposium on Computational Intelligence and Design*. **pp.** 219-222.

Witten, I.H. & Frank, E. 2005. *Data Mining Practical Machine Learning Tools and Techniques*. 2nd Edition. Morgan Kaufmann Publishers: San Francisco.

Priyadarsini, R.P., Valarmathi, M.L., Sivakumari, S. 2011. Gain ratio Based Feature Selection Method For Privacy Preservation. *ICTACT Journal on Soft Computing* 1 (4): 201-205.

Duneja, A., Puyalnithi, T. 2017. Enhancing Classification Accuracy of K-Nearest Neighbours Algorithm Using Gain ratio. *International Research Journal of Engineering and Technology (IRJET)* **4**(9): 1385-1388.