

# Perbandingan Metode Probabilistik *Naive Bayesian Classifier* dan Jaringan Syaraf Tiruan Learning Vector Quantization dalam Kasus Klasifikasi Penyakit Kandungan

**Praditio Aditya Nugraha**

Jurusan Informatika  
Jl. Ir. Sutami 36A Kentingan  
Surakarta  
praditio.nugraha@gmail.com

**Ristu Saptono**

Jurusan Informatika  
Jl. Ir. Sutami 36A Kentingan  
Surakarta  
ristu.uns@gmail.com

**Meiyanto Eko Sulisty**

Jurusan Informatika  
Jl. Ir. Sutami 36A Kentingan  
Surakarta  
meiyantoekosulistyo@gmail.com

## ABSTRAK

Penyakit kandungan merupakan momok menakutkan bagi kaum wanita. Hal tersebut dikarenakan kaum wanita sangat beresiko tinggi untuk terkena penyakit kandungan. Penyakit kandungan yang dimaksud dalam penelitian ini adalah radang panggul, mioma uteri, kanker serviks, dan kanker ovarium. Penelitian ini mengajukan dua pendekatan berbeda untuk penyelesaian kasus diagnosa penyakit kandungan, yaitu pendekatan probabilistik dengan *Naive Bayesian Classifier* dan jaringan syaraf tiruan *Learning Vector Quantization*. Pada pendekatan *Naive Bayesian Classifier* digunakan *laplacian smoothing* sehingga dapat menghindari kegagalan klasifikasi, yang diharapkan dapat meningkatkan performa dari *Naive Bayesian Classifier* dalam penyelesaian kasus ini. Begitu juga pada metode *Learning Vector Quantization*, dilakukan pemilihan model paling tepat berdasar kepada tiga parameter yaitu *learning rate*, *learning rate reduction*, dan *epoch*. Pengujian dilakukan dengan menggunakan prosedur *random stratified and repeated holdout* pada empat skenario simulasi dengan porsi data *training* dan *testing*: 20/80, 40/60, 60/40, dan 80/20. Dari hasil uji coba dapat ditunjukkan bahwa metode *Naive Bayesian Classifier* dengan penambahan *Laplacian Smoothing* memiliki tingkat akurasi pengujian paling tinggi, yaitu secara berturut-turut 88%, 92.4%, 92.8% dan 92.4% dibandingkan dengan *Naive Bayesian Classifier* tanpa *Laplacian Smoothing* (32%; 67.8%; 79%; 89.6%) dan *Learning Vector Quantization* (82.4%; 88.8%; 89.4%; 95.2%). Dalam hal kecepatan pembelajaran, *Learning Vector Quantization* dapat menyelesaikan proses pembelajaran dengan waktu yang lebih singkat dari *Naive Bayesian Classifier*, hal tersebut dikarenakan pada kasus ini *Learning Vector Quantization* dapat memberikan performa yang baik dengan jumlah *epoch* yang relatif kecil. Secara umum *Naive Bayesian Classifier* dengan *Laplacian Smoothing* merupakan metode paling

baik dalam menyelesaikan kasus klasifikasi penyakit kandungan.

**Kata Kunci:** Penyakit Kandungan, *Naive Bayesian Classifier*, *Laplacian Smoothing*, *Learning Vector Quantization*

## 1. PENDAHULUAN

Setiap wanita memiliki kemungkinan yang sangat tinggi untuk mengidap penyakit kandungan, maka dari itu penyakit kandungan dapat dikatakan sebagai masalah yang sangat mengerikan yang harus dihadapi oleh wanita. Penyakit kandungan memiliki banyak jenis, empat diantaranya adalah: kanker ovarium, mioma uteri, kanker serviks, dan radang panggul.

Keempat jenis penyakit kandungan diatas merupakan jenis penyakit kandungan yang umumnya dijumpai pada wanita, selain itu keempat jenis penyakit kandungan tersebut memiliki kesamaan gejala sehingga diperlukan kejelian dan ketelitian dalam menganalisa gejala pada pasien sehingga dapat menghasilkan diagnosa yang tepat. Dengan diagnosa yang tepat, maka penanganan terhadap penyakit tersebut dapat segera dilakukan demi mencegah dampak yang semakin buruk.

Dibutuhkan metode yang tepat untuk dapat menyelesaikan masalah ini, sehingga diagnosa yang diberikan sistem menjadi akurat, salah satunya dengan menggunakan jaringan syaraf tiruan. Beberapa penelitian menggunakan pendekatan jaringan syaraf tiruan, seperti yang dilakukan oleh Prabhawaningrum 2013 [1] yaitu diagnosa penyakit kandungan radang panggul, mioma uteri, kanker serviks, dan kanker ovarium. Parameter yang digunakan berupa 18 gejala yang mungkin terjadi pada keempat jenis penyakit kandungan tersebut. Penelitian ini berfokus pada perbandingan algoritma dengan pendekatan jaringan syaraf tiruan. Algoritma yang dibandingkan adalah *Levenberg-Marquardt* dengan *Backpropagation*. Kedua metode tersebut dapat

menyelesaikan kasus diagnosa penyakit kandungan tersebut dengan tingkat akurasi masing-masing 71% dan 81%. Namun pada penelitian yang dilakukan oleh Nurkhozin 2011 [2] metode LVQ dianggap lebih baik kinerjanya daripada BP dalam menyelesaikan kasus diagnosa penyakit diabetes milletus. Selain jaringan syaraf tiruan, pendekatan probabilistik juga sering dipakai pada aplikasi kesehatan. Pada penelitian yang dilakukan Bhuvaneswari 2012 [3] yaitu analisa penggunaan *Naïve Bayesian Classifier* dalam penerapannya pada aplikasi kesehatan (*healthcare application*). Pada penelitian ini disimpulkan bahwa metode *Naïve Bayesian Classifier* sangat baik digunakan sebagai metode untuk sistem pendukung keputusan dalam bidang kesehatan (*healthcare*).

Yang menjadi masalah pada *Naïve Bayesian Classifier* adalah kemungkinan hadirnya nilai nol pada model probabilitas yang menyebabkan metode ini gagal mengklasifikasi. Hadirnya nol pada model probabilitas tersebut diakibatkan oleh terbatasnya data yang dikumpulkan untuk dijadikan data *training*. Maka dari itu, hadirnya *zero probability* harus dihindari yaitu dengan menerapkan *smoothing* [4].

Berdasarkan hasil penelitian diatas, dengan menggunakan *dataset* Prabhawaningrum 2013 [1] akan dilakukan penelitian menggunakan pendekatan yang berbeda. Pada penelitian ini akan dibandingkan performa dari masing-masing pendekatan, yaitu perbandingan antara pendekatan probabilistik dengan metode *Naïve Bayesian Classifier* dan jaringan syaraf tiruan *Learning Vector Quantization*.

**2. LANDASAN TEORI**

**2.1 Machine Learning**

Menurut Jones 2008 [5] dalam bukunya yang berjudul “Artificial Intelligent – A System Approach” menyebutkan bahwa machine learning berfokus kepada algoritma dan metode yang membuat sebuah komputer dapat memiliki kemampuan untuk me-lakukan pembelajaran. Sudut pandang lain mengenai machine learning adalah dari bagaimana cara komputer belajar dari contoh yang sudah tersedia. *Machine learning* dapat mempelajari tentang data-data yang sudah ada dan keterkaitannya, kemudian menghasilkan informasi.

Terdapat tiga jenis pembelajaran yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*.

**2.2 Klasifikasi**

Klasifikasi adalah sebuah proses untuk mencari model atau fungsi yang menjelaskan dan membedakan kelas atau konsep dari data, dengan tujuan untuk menggunakan model dan melakukan prediksi dari kelas suatu objek dimana tidak diketahui label dari kelas tersebut. Model yang ada

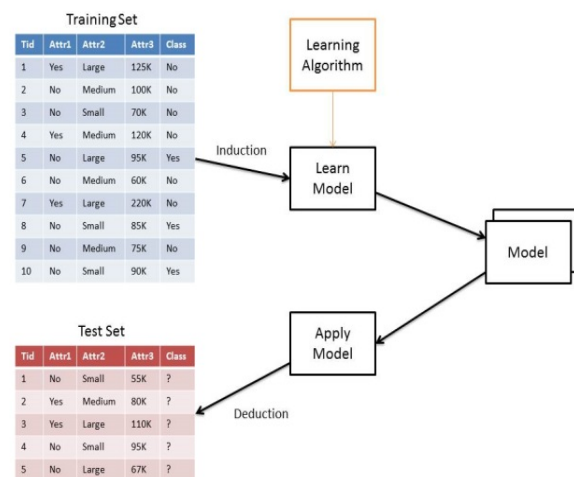
berasal dari analisis dari kumpulan *training data* (objek data dimana kelas dari label diketahui) [6].

Algoritma yang dapat digunakan untuk klasifikasi adalah antara lain *Naïve Bayesian Classifier*, *Adaptive Bayes Network*, *Decision Tree*, dan *Support Vector Machine*.

Tujuan dari klasifikasi adalah untuk:

- a. Menemukan model dari *training set* yang membedakan record kedalam kategori atau kelas yang sesuai, model tersebut kemudian digunakan untuk mengklasifikasikan *record* yang kelasnya belum diketahui sebelumnya pada *test set*.
- b. Mengambil keputusan dengan mem-prediksikan suatu kasus, berdsarkan hasil klasifikasi yang diperoleh.

Berdasarkan pada Gambar 1, untuk mendapatkan model maka diperlukan analisis terhadap data latih (*training set*). Sedangkan untuk mengetahui tingkat keberhasilan metode dalam memecahkan masalah klasifikasi (biasanya dilihat dari tingkat akurasi), maka diperlukan data uji (*test set*) digunakan untuk mengetahui tingkat akurasi dari model yang telah dihasilkan. Klasifikasi dapat digunakan untuk memprediksi nama atau nilai kelas dari suatu obyek data.



**Gambar 1.** Pendekatan umum pemecahan masalah klasifikasi [7]

**2.3 Naïve Bayesian Classifier**

*Naïve Bayesian Classifier* (NBC) adalah salah satu metode dalam machine learning yang bertugas untuk mengklasifikasi. Metode ini berakar pada teorema Bayes. Salah satu ciri dari *Naïve Bayesian Classifier* adalah asumsi independensi yang kuat (naïve). *Naïve Bayesian Classifier* dari segi performa lebih baik dari algoritma *Decision Tree* dan algoritma *Selected Neural Network Classifier*, juga memiliki kecepatan dan keakuratan yang tinggi bila diimplementasikan dalam data yang ukurannya besar.

Keuntungan penggunaan *Naïve Bayesian Classifier* adalah metode ini hanya membutuhkan jumlah data pelatihan (*training data*) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian [8].

Asumsi independensi yang kuat antar parameter (*naïve*) membuat syarat peluang menjadi sangat sederhana dan mudah untuk dihitung. Kemudian penjabaran  $(F_1, \dots, F_n|C)$  dapat disederhanakan menjadi sebagai berikut:

$$P(F_1 \dots F_n|C) = \prod_{i=1}^n P(F_i|C)$$

Dengan persamaan diatas, teorema Bayes dapat dituliskan kembali sebagai berikut:

$$P(C|F_1 \dots F_n) = \frac{1}{P(F_1, F_2, \dots, F_n)} P(C) \prod_{i=1}^n P(F_i|C)$$

$$P(C|F_1 \dots F_n) = \frac{P(C)}{Z} \prod_{i=1}^n P(F_i|C)$$

Persamaan diatas merupakan model dari teorema *Naïve Bayes* yang dapat digunakan sebagai metode untuk klasifikasi. Z diatas merupakan *evidence* yaitu karakteristik-karakteristik sampel secara global. Kemudian diambil nilai tertinggi untuk menjadi hasil klasifikasi sesuai dengan persamaan berikut:

$$C_{NB} = \underset{c}{\operatorname{argmax}} p(C = c) \prod_{i=1}^n P(F_i|C = c)$$

**2.4 Laplacian smoothing**

Keterbatasan data tersebut menyebabkan adanya nilai nol pada model probabilitas. Hal ini menyebabkan *Naïve Bayesian Classifier* tidak dapat mengklasifikasi sebuah data input. Maka dari itu, adanya nilai nol pada model probabilitas harus dihindari. Cara untuk menghindari adalah dengan menerapkan *smoothing* pada saat pembangunan model probabilitas.

Tujuan daripada *smoothing* adalah untuk mengurangi probabilitas dari hasil/keluaran yang terobservasi, dan juga sekaligus meningkatkan/menambah probabilitas hasil/keluaran yang belum terobservasi [4].

*Laplacian smoothing* adalah salah satu metode *smoothing* yang dapat diterapkan pada *Naïve Bayesian Classifier*. Konsepnya sangat sederhana, yaitu dengan menambahkan nilai positif yang kecil pada setiap nilai probabilitas kondisional yang ada sehingga terhindarnya nilai nol pada model probabilitas. Dengan konsepnya yang sederhana tersebut, diharapkan tidak memperlambat waktu eksekusi dalam proses *training* pada metode *Naïve Bayesian Classifier*. Secara umum, rumus *laplacian smoothing* dapat dituliskan sebagai berikut:

$$P(F_i = v_j|c_k) = \frac{n_{ijk} + 1}{n_k + s_i}$$

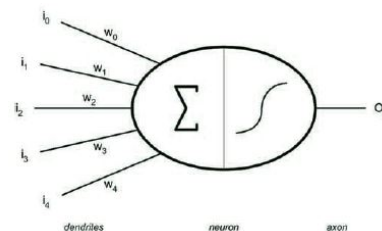
Dengan  $n_{ijk}$  adalah jumlah data yang termasuk dalam kelas  $c_k$  dimana  $F_i = v_j$ . Kemudian  $n_k$  adalah jumlah data yang termasuk dalam kelas  $c_k$  dan  $s_i$  adalah jumlah kemungkinan nilai yang muncul (*possible values*) pada  $F_i$ .

**2.5 Jaringan Syaraf Tiruan**

Jaringan saraf tiruan (JST atau *Artificial Neural Network*), adalah sistem komputasi dimana arsitektur dan operasi diilhami dari pengetahuan tentang sel saraf biologis di dalam otak, yang merupakan salah satu representasi buatan dari otak manusia yang selalu menstimulasi proses pembelajaran pada otak manusia tersebut [9].

Jaringan Syaraf Tiruan (JST) terdiri dari jaringan yang terhubung antara elemen-elemen yang secara kolektif dapat digunakan untuk masalah-masalah yang sulit. Saat dilakukan pembelajaran, Jaringan syaraf tiruan dapat mengeneralisasi untuk menyelesaikan masalah-masalah yang memiliki karakteristik yang sama [5].

Di dalam *neuron* terdapat fungsi yang bekerjasama dengan input dan kemudian menormalisasi *output*-nya melalui fungsi transfer [5]. Seperti pada Gambar 2 berikut ini.



**Gambar 2.** Tampilan neuron secara singkat (Jonas, 2008)

**2.6 Learning Vector Quantization**

*Learning Vector Quantization* (LVQ) adalah suatu metode untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi. Suatu lapisan kompetitif akan secara otomatis belajar untuk mengklasifikasikan vektor-vektor input. Kelas-kelas yang didapatkan sebagai hasil dari lapisan kompetitif ini hanya tergantung pada jarak antara vektor-vektor input. Jika 2 vektor input mendekati sama, maka lapisan kompetitif akan meletakkan kedua vektor input tersebut ke kelas yang sama [10].

Dibawah ini merupakan langkah perhitungan dalam pembelajaran menggunakan *Learning Vector Quantization*.

1. Tetapkan: W, MaxIter, α.
2. Masukkan:
  - Input x(m,n);
  - m = jumlah input,

- $n$  = jumlah data
  - $target(1,n)$ .
3. Tetapkan kondisi awal:
    - $epoch = 0$ ;
  4. Kerjakan jika: ( $epoch < MaxIter$ )
    - a.  $epoch = epoch + 1$ ;
    - b. Kerjakan untuk  $i=1$  sampai  $n$ 
      - i. Tentukan  $J$  sedemikian hingga  $\|x-w_j\|$  minimum
      - ii. Perbaiki  $w_j$  dengan ketentuan:
        - Jika  $T = C_j$  maka:  
 $w_j(\text{baru}) = w_j(\text{lama}) + \alpha (x-w_j(\text{lama}))$
        - Jika  $T \neq C_j$  maka:  
 $w_j(\text{baru}) = w_j(\text{lama}) - \alpha (x-w_j(\text{lama}))$
5. Kurangi nilai  $\alpha$ .

**2.7 Holdout Validation**

Pada kondisi terbatasnya data yang digunakan untuk *training* dan *testing*, diperlukan metode untuk mendapatkan hasil tingkat akurasi dari sebuah metode pada machine learning. Salah satu cara untuk validasi adalah dengan menggunakan metode *holdout*. Metode *holdout* adalah metode yang akan menyediakan sejumlah data untuk digunakan sebagai data *testing*, dan sisanya sebagai data *training* [11].

Saat proses pengacakan data untuk dibagi sebagai data *training* dan *testing*, sangat mungkin terjadi *overrepresented* pada salah satu atau lebih klasifikasi. Dalam artian bahwa klasifikasi tersebut dominan dibandingkan klasifikasi lainnya, sehingga data *training* dan *testing* yang tercipta menjadi tidak representatif. Maka dari itu diperlukan prosedur *stratification holdout*, dimana dengan prosedur ini dapat dijamin bahwa setiap klasifikasi dapat terwakili pada data *training* dan *testing* yang tercipta secara proporsional. Menurut Freitas 2002 [12] kelas yang terbagi dari hasil proses *holdout* proporsinya harus sedekat mungkin dengan proporsi aslinya.

Dilakukan perulangan terhadap seluruh proses *training* dan *testing* beberapa kali dengan data *training* dan *testing* yang teracak. Kemudian diambil nilai rata-ratanya. Prosedur ini dikatakan sebagai *repeated holdout*.

**2.8 Penyakit Kandungan**

Penyakit kandungan merupakan penyakit yang berbahaya dan berpotensi besar terjadi pada kaum wanita. Penyakit kandungan ini biasanya menyerang organ reproduksi wanita sehingga memungkinkan untuk menyebabkan kemandulan. Beberapa penyakit kandungan adalah radang panggul, mioma uteri, kanker serviks dan kanker ovarium. Penyakit-penyakit tersebut menyerang organ reproduksi wanita seperti ovarium, tuba fallopi, uterus, endometrium, serviks, dan vagina.

Mioma uteri adalah tumor jinak otot polos yang terdiri dari sel-sel jaringan otot polos, jaringan

fibroid dan kolagen [13]. Gejala mioma uteri adalah sebagai berikut [14] [13] [15]: frekuensi buang air kecil bertambah, sulit buang air besar, infertilitas, pendarahan mens abnormal, terdapat benjolan di perut bagian bawah, pendarahan diluar siklus haid, nyeri haid, nyeri ketika berhubungan seksual, anemia.

Radang panggul atau *pelvic inflammatory disease* (PID) adalah infeksi traktus genital atas yang merupakan salah satu komplikasi dari infeksi menular seksual (IMS) [16]. PID merupakan kelaian ginekologis yang paling sering terjadi di Amerika Serikat dan Inggris [15]. Gejala penyakit radang panggul berupa [14] [16] [15]: nyeri perut bagian bawah, temperatur oral lebih dari 38°C, keluar cairan dari vagina, pendarahan tidak teratur, sakit kepala, lesu, nyeri berhubungan seksual dan nyeri buang air kecil. Kanker ovarium adalah kanker yang bermula pada indung telur (ovarium) wanita. Kanker ovarium memiliki potensi menyebarkan sel ganas dengan sangat cepat ke seluruh rongga abdomen [17].

Gejala kanker ovarium adalah sebagai berikut [14] [18] [19] [15]: nyeri ketika berhubungan seksual, kembung, sulit buang air besar, sering buang air kecil, nafsu makan menurun, cepat lelah, anemia, nyeri panggul dan nyeri punggung bagian bawah.

Kanker serviks merupakan kelainan yang terjadi pada sel-sel serviks yang berkembang dengan cepat dan tidak terkontrol [20]. Kanker serviks 80% disebabkan oleh HPV (*Human Papilloma Virus*) [15]. Gejala kanker serviks meliputi [20] [19] [15]: pendarahan abnormal, pendarahan haid abnormal, nyeri panggul, nyeri ketika berhubungan seksual, keputihan, nyeri buang air kecil, anemia, nyeri punggung bagian bawah, penurunan nafsu makan dan penurunan berat badan drastis.

**3. METODOLOGI PENELITIAN**

**3.1 Pengumpulan Data**

Dalam memprediksi penyakit kandungan yang diderita, digunakan 18 gejala klinis yang dirasakan oleh pasien.

Tabel 1 merupakan gejala klinis yang dijadikan sebagai parameter dalam mendiagnosa jenis penyakit kandungan tersebut [17] [15] [14] [19] [18].

**Tabel 1.** Daftar Gejala Klinis

ID Gejala	Gejala	Nilai
P001	Anemia	Tidak
		Ya
P002	Nyeri Haid	Tidak
		Nyeri Setelah Haid
		Nyeri Ketika Haid
P003	Susah Hamil	Tidak

**Tabel 1.** Daftar Gejala Klinis

ID Gejala	Gejala	Nilai
		Ya
P004	Benjolan di Perut	Tidak
		Ya
P005	Pendarahan	Tidak
		Pendarahan Menstruasi Abnormal
		Pendarahan Tiba-tiba
P006	Nyeri Berhubungan Seksual	Tidak
		Ya
P007	Cepat Lelah	Tidak
		Ya
P008	Penurunan Berat Badan	Tidak
		Ya
P009	Nyeri Panggul	Tidak
		Ya
P010	Gangguan Pencernaan	Tidak
		Sembelit
		Diare
P011	Nyeri Perut	Tidak
		Nyeri Pada Rongga Perut
		Nyeri Perut Bagian Bawah
		Nyeri Perut Bagian Bawah dan Pinggul
P012	Nyeri Punggung	Tidak
		Ya
P013	Penurunan Nafsu Makan	Tidak
		Ya
P014	Demam	Tidak
		Ya
P015	Sakit Kepala	Tidak
		Ya
P016	Kembung	Tidak
		Ya
P017	Keputihan	Tidak
		Ya
P018	Gangguan BAK	Tidak
		Sering BAK
		Nyeri BAK
		Nyeri dan Sering BAK

Berdasarkan pada gejala-gejala pada Tabel 1, dapat diklasifikasikan menjadi 5 kategori yang berada pada Tabel 2.

**Tabel 2.** Kategori Penyakit

No.	Kategori Penyakit
1	Tidak Termasuk 4 Jenis Penyakit Kandungan
2	Radang Panggul
3	Mioma Uteri
4	Kanker Serviks
5	Kanker Ovarium

Berdasarkan uraian diatas, metode yang akan di-implementasikan penulis dalam penyelesaian kasus ini yaitu *Naïve Bayesian Classifier* dan *Learning Vector Quantization* dapat diterapkan. Hal tersebut dikarenakan memang spesialisasi dari kedua metode yang akan digunakan adalah untuk

masalah klasifikasi. Kedua metode ini dapat menyelesaikan masalah klasifikasi dimana nilai-nilai yang digunakan bertipe kategorial.

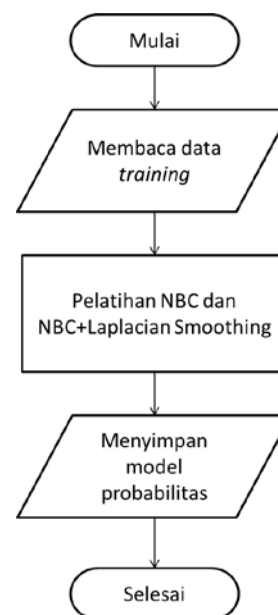
Data yang digunakan pada penelitian ini didapatkan dari rekam medik pasien di RSUD Dr. Moewardi Solo yang sebelumnya telah diambil pada penelitian yang dilakukan oleh Prabhawaningrum 2013 [1]. Data yang digunakan pada penelitian ini adalah sebanyak 125 data. Pembagian data *training* dan *testing* akan dilakukan secara acak sesuai dengan persentase pembagian data yang dikehendaki pada setiap klasifikasi sehingga akan tercipta dataset yang bermacam-macam pada setiap percobaan.

### 3.2 Pembelajaran NBC

Pada setiap data *training* yang telah disediakan berdasarkan skenario yang ditentukan dan pembagian yang telah dilakukan menurut prosedur pengujian *random stratified and repeated holdout*, akan dilakukan pembelajaran dengan menggunakan metode *Naïve Bayesian Classifier*. Tidak hanya itu, penulis juga melakukan modifikasi pada metode tersebut dengan menerapkan *laplacian smoothing* dalam proses pembelajarannya.

Gambar 3 merupakan rancangan alur pembelajaran dengan menggunakan *Naïve Bayesian Classifier* dengan dan tanpa menggunakan *smoothing*.

Hasil pembelajaran tersimpan berupa nilai probabilitas yang berasal dari metode *Naïve Bayesian Classifier* maupun dari hasil *smoothing*. Kemudian hasil pembelajaran tersebut nantinya akan dilewati oleh data-data pada data *training* dan data *testing*.



**Gambar 3.** Alur Pembelajaran NBC

### 3.3 Pembelajaran LVQ

Pada setiap data *training* yang telah disediakan berdasarkan skenario yang ditentukan dan pembagian yang telah dilakukan menurut prosedur pengujian *random stratified and repeated holdout*, akan dilakukan pembelajaran dengan menggunakan metode *Learning Vector Quantization*. Berbeda dengan *Naive Bayesian Classifier*, pada metode *Learning Vector Quantization* terdapat beberapa parameter penentu dalam proses pembelajaran. Parameter-parameter tersebut adalah sebagai berikut:

a. *Epoth Maksimal*

Untuk kepentingan simulasi proses pembelajaran pada metode *Learning Vector Quantization*, yaitu menentukan maksimal iterasi (*epoch*) yang dilakukan saat proses pembelajaran.

b. *Learning rate*

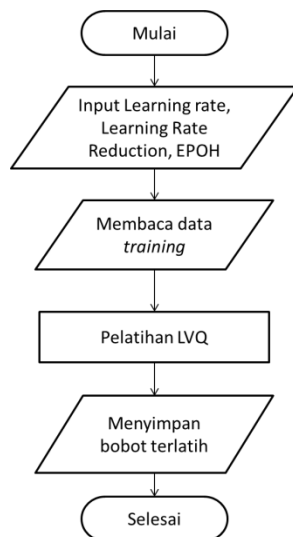
Untuk kepentingan simulasi proses pembelajaran pada metode *Learning Vector Quantization*, yaitu menentukan *learning rate* awal.

c. *Learning rate reduction*

Untuk kepentingan simulasi proses pembelajaran pada metode *Learning Vector Quantization*, yaitu menentukan pengurangan *learning rate* pada setiap satu kali *epoch*.

Gambar 4 merupakan rancangan dalam pembelajaran menggunakan *Learning Vector Quantization*.

Hasil pembelajaran tersimpan berupa bobot jaringan yang telah terlatih. Kemudian hasil pembelajaran tersebut nantinya akan dilewati oleh data-data pada data *training* dan data *testing*.



Gambar 4. Alur Pembelajaran LVQ

#### 3.3.1 Penentuan Model LVQ

Salah satu yang menjadi pembahasan penting dalam pendekatan jaringan syaraf tiruan adalah pengaruh penggunaan *learning rate* dan jumlah *epoch* dalam memaksimalkan performa pada penekatan tersebut. *Learning Vector Quantization*

sebagai salah satu metode jaringan syaraf tiruan pun melibatkan *learning rate* dan *epoch* (iterasi) dalam proses pembelajarannya. Maka dari itu perlu ditelaah penggunaan kedua parameter tersebut.

Pada penelitian ini akan dilakukan pengujian terhadap pengaruh *learning rate*. Namun diperlukan juga pengujian terhadap *learning rate reduction*, karena pada setiap iterasi nilai *learning rate* akan berubah sesuai dengan *learning rate reduction* yang digunakan. Kemudian akan diuji pula bagaimana pengaruh jumlah iterasi terhadap tingkat akurasi. Dimana akan dilakukan percobaan 10 kali dengan data *training* dan *testing* yang terpisah secara acak dengan prosedur *random stratified and repeated holdout*.

Setelah diketahui performa maksimal dari metode ini, menjadi sangat objektif untuk membandingkan metode ini dengan *Naive Bayesian Classifier*.

### 3.4 Rancangan Skenario Percobaan

Pada penelitian ini akan dilakukan pengujian performa masing-masing metode. Untuk itu penulis merencanakan untuk membuat skenario yang berbeda-beda porsi data pembelajarannya (*training*) dan data pengujiannya. Sehingga dengan adanya skenario tersebut akan dilihat seperti apa performa metode yang diuji jika dihadapkan pada skenario-skenario tersebut. Skenario yang akan dibuat adalah seperti pada Tabel 3.

Tabel 3. Tabel Rencana Simulasi

Nama Simulasi Percobaan	Porsi Data		Jumlah Data Terbagi	
	Training	Testing	Training	Testing
20/80	20%	80%	25	100
40/60	40%	60%	50	75
60/40	60%	40%	75	50
80/20	80%	20%	100	25

### 3.5 Prosedur Proses Pengujian

Pada setiap skenario yang telah ditentukan, dilakukan pengujian pada kedua metode tersebut. Pengujian ini berdasar kepada prosedur pengujian *random stratified and repeated holdout*, dimana prosedur ini melakukan pemecahan data sesuai dengan porsi yang ditentukan. Pemecahan tersebut dilakukan secara *random* (acak). Gambar 5 adalah gambaran prosedur *random stratified and repeated holdout*.

Ide dari prosedur pengujian dan pembagian data ini berdasar pada prosedur-prosedur berikut ini:

1. *Holdout*

Merupakan prosedur pemecahan data (*splitting*) menjadi data *training* dan *testing*.

2. *Stratified Random Sampling*

Prosedur untuk mengacak data yang akan masuk ke data *training* dan *testing*. Namun prosedur ini tidak melakukan pengacakan secara murni, namun secara bertingkat (*stratified*) untuk menghindari adanya pemecahan data yang tidak



proporsional jika ditinjau dari jumlah data pada setiap klasifikasi (*overpresented*).

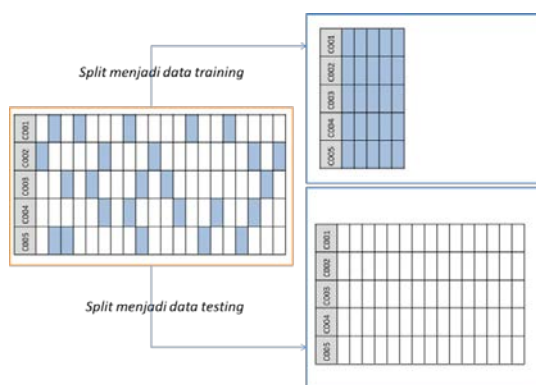
3. *Repeated Holdout*

Prosedur *holdout* yang dilakukan berkali-kali (berulang) kemudian diambil nilai rata-rata akurasi dari setiap percobaan.

Pengujian dilakukan kepada metode *Naïve Bayesian Classifier* dengan dan tanpa *laplacian smoothing* dan pada metode *Learning Vector Quantization*. Pengujian dilakukan dengan mengukur tingkat akurasi masing-masing metode dengan rumus sebagai berikut :

$$Akurasi = \frac{Jumlah\ prediksi\ benar}{Jumlah\ Data} \times 100\%$$

Akurasi tersebut akan didapat pada masing-masing percobaan kemudian dihitung rata-rata dan standar deviasinya.



Gambar 5. Prosedur *Stratified Random Sampling*

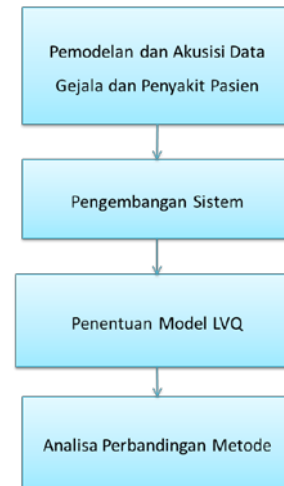
Selain akurasi, dihitung juga waktu pembelajaran dari masing-masing metode tersebut (dalam satuan *milisecond*).

3.6 Implementasi

Pada penelitian ini, implementasi dilakukan dengan menggunakan aplikasi yang dibangun oleh penulis dengan menggunakan bahasa pemrograman PHP. Implementasi dilakukan dengan menggunakan komputer yang memiliki spesifikasi sebagai berikut:

- Processor : Intel Core i5
- Memory : 4.00 GB
- Operating System : Windows 8 64-bit
- Database : MySQL 5.1.41
- Web Server : Apache 2.2.1.6

Implementasi dalam penelitian ini dilakukan dengan langkah kerja berikut.



Gambar 6. Alur Implementasi Penelitian

3.6.1 Pengembangan Sistem

Sistem yang dibangun merupakan sistem yang digunakan sebagai pendukung penelitian ini. Pada sistem ini terdapat fungsi untuk menentukan parameter dasar dalam penelitian ini yaitu pemasukan klasifikasi penyakit dan gejala. Kemudian terdapat fungsi untuk pemasukan data pembelajaran sebagai *knowledge* pada sistem ini.

Kemudian tentu saja sistem yang dibangun memiliki kemampuan untuk melakukan pembelajaran terhadap data yang sudah dimasukan dengan menggunakan dua metode yang akan dibandingkan, yaitu *Naïve Bayesian* (ditambah dengan *laplacian smoothing*), dan *Learning Vector Quantization*.

Sistem yang dikembangkan dapat melakukan analisa terhadap kedua metode yang dibandingkan. Analisa tersebut meliputi perhitungan akurasi rata-rata dengan menggunakan prosedur *holdout* yang dilakukan secara berulang-ulang. Pembagian data (*data splitting*) dapat ditentukan berdasar kepada porsi data pelatihan dan data pengujian.

3.6.2 Penentuan Model LVQ

Untuk dapat melihat pengaruh *epoch*, *learning rate*, dan *learning rate reduction* maka diperlukan evaluasi terlebih dahulu pada metode *Learning Vector Quantization*. Evaluasi ini bertujuan untuk mendapatkan kombinasi yang baik antara *epoch*, *learning rate*, dan *learning rate reduction* ditinjau dari kecepatan dan tingkat akurasinya, sehingga didapat best performance of *Learning Vector Quantization*. Pengujian dilakukan dengan menggunakan prosedur yang sama, yaitu *random stratified and repeated holdout*.

Dalam Mengevaluasi performa *Learning Vector Quantization* digunakan skenario normal, dalam hal ini digunakan skenario 60% data *training* dan 40% data *testing* dimana dilakukan perhitungan akurasi sebanyak 10 kali kemudian diambil rata-rata akurasinya.

Penulis akan melakukan kombinasi terhadap *learning rate* dan *learning rate reduction* dengan memberikan *learning rate reduction* yang lebih kecil hingga lebih besar terhadap *learning rate* yang diuji.

Kemudian setelah didapatkan kombinasi terbaik dari *learning rate* dan *learning rate reduction*, diuji parameter *epoch* maksimal sehingga diharapkan dengan diketahuinya *epoch* yang tepat dapat menekan waktu pembelajaran menggunakan metode LVQ. Diambil beberapa sampel *epoch* kemudian diambil *epoch* paling kecil dengan akurasi maksimal.

### 3.6.3 Evaluasi Perbandingan

Pada tahap ini dilakukan perbandingan terhadap kedua metode dengan melakukan pengujian dengan prosedur *random stratified and repeated holdout*. Parameter pada *Learning Vector Quantization* meliputi *epoch*, *learning rate*, dan *learning rate reduction* digunakan kombinasi yang menghasilkan akurasi tertinggi (*best performance of Learning Vector Quantization*) yang telah dikaji pada tahap sebelumnya. Sedangkan pada metode *Naïve Bayesian Classifier* akan diimplementasikan antara *non-smoothing* dan penggunaan *smoothing*.

Perbandingan dilakukan dengan menggunakan skenario simulasi yang telah dibuat. Kemudian hasil perbandingan akurasi dan waktu pembelajaran akan dituliskan pada tabel yang telah dirancang untuk perbandingan kedua metode ini.

Pada tahap ini akan dihasilkan tingkat akurasi rata-rata dari hasil percobaan di setiap skenario simulasi pada kedua metode. Selain akurasi, dibandingkan pula kecepatan waktu pembelajaran pada *Naïve Bayesian Classifier smoothing*, *non-smoothing* dan *Learning Vector Quantization*.

### 3.7 Analisis Hasil

Analisis hasil akan dilakukan berdasarkan hasil evaluasi metode dan evaluasi perbandingan yang telah dilakukan sebelumnya.

## 4. HASIL DAN PEMBAHASAN

### 4.1 Deskripsi dan Pemodelan Data

Pada penelitian ini, terdapat 4 buah penyakit kandungan yang dapat dideteksi yaitu radang panggul, mioma uteri, kanker serviks, dan kanker ovarium. Kemudian terdapat juga kelas yang merupakan kelas untuk yang tidak termasuk pada 4 jenis penyakit kandungan tersebut.

Penulis memasukan 5 kelas tersebut kepada aplikasi yang telah dibuat, kemudian masing-masing kelas tersebut akan secara otomatis memiliki ID yang unik (berbeda-beda). Tabel 4 merupakan 5 kelas yang telah dimasukan pada sistem.

Penulis juga memasukan 18 gejala klinis tersebut kepada aplikasi yang telah dibuat, kemudian masing-masing gejala tersebut akan

secara otomatis memiliki ID yang unik (berbeda-beda).

Dalam perhitungan menggunakan metode *Learning Vector Quantization*, nilai pada *possible value* dalam setiap gejala akan dilibatkan dalam perhitungan matematika. Maka dari itu diberikan nilai pada rentang 0 – 1 tergantung pada kasusnya setiap gejala. Hal ini memungkinkan bahwa pemberian nilai berdasarkan tingkat parahnya kasus, semakin parah kasusnya maka semakin besar nilainya. Tabel 5 adalah gejala yang dimasukan pada sistem [1] [21].

**Tabel 4.** Kategori Penyakit Dalam Sistem

ID Penyakit	Kategori Penyakit
C001	Tidak Termasuk 4 Jenis Penyakit Kandungan
C002	Radang Panggul
C003	Mioma Uteri
C004	Kanker Serviks
C005	Kanker Ovarium

**Tabel 5.** Gejala dan *Possible Value* pada Sistem

ID Gejala	Gejala	Nilai	
		Gejala	Nilai
P001	Anemia	Tidak	0
		Ya	1
P002	Nyeri Haid	Tidak	0
		Nyeri Setelah Haid	0.5
		Nyeri Ketika Haid	1
P003	Susah Hamil	Tidak	0
		Ya	1
P004	Benjolan di Perut	Tidak	0
		Ya	1
P005	Pendarahan	Tidak	0
		Pendarahan Menstruasi Abnormal	0.5
		Pendarahan Tiba-tiba	1
P006	Nyeri Berhubungan Seksual	Tidak	0
		Ya	1
P007	Cepat Lelah	Tidak	0
		Ya	1
P008	Penurunan Berat Badan	Tidak	0
		Ya	1
P009	Nyeri Panggul	Tidak	0
		Ya	1
P010	Gangguan Pencernaan	Tidak	0
		Sembelit	0.5
		Diare	1
P011	Nyeri Perut	Tidak	0
		Nyeri Pada Rongga Perut	0.25
		Nyeri Perut Bagian Bawah	0.5
		Nyeri Perut Bagian Bawah dan Pinggul	1



ID Gejala	Gejala	Nilai	
P012	Nyeri Punggung	Tidak	0
		Ya	1
P013	Penurunan Nafsu Makan	Tidak	0
		Ya	1
P014	Demam	Tidak	0
		Ya	1
P015	Sakit Kepala	Tidak	0
		Ya	1
P016	Kembung	Tidak	0
		Ya	1
P017	Keputihan	Tidak	0
		Ya	1
P018	Gangguan BAK	Tidak	0
		Sering BAK	0.25
		Nyeri BAK	0.5
		Nyeri dan Sering BAK	1

Setelah ditentukan gejala dan klasifikasinya, penulis melakukan akusisi data. Data yang dimasukkan adalah 125 data yang berasal dari penelitian yang dilakukan oleh Prabhawaningrum 2013 [1].

#### 4.2 Pengembangan Sistem

Sistem yang dibangun hanya digunakan sebagai pendukung, bukan sebagai hal yang difokuskan pada penelitian ini. Maka dari itu penulis tidak menjelaskan secara detail bagaimana alur dalam sistem ini. Pada tahap ini penulis melakukan pengembangan sistem yang dapat melakukan seluruh proses yang akan dilakukan pada penelitian ini, meliputi:

1. Menambahkan klasifikasi penyakit
2. Menambahkan parameter/gejala, dimana jika jenis parameternya kategorial maka terdapat opsi untuk memasukan apa saja value pada sebuah parameter tersebut
3. Menambahkan data berdasarkan klasifikasi dan parameter yang telah ditentukan
4. Melakukan pelatihan dengan menggunakan metode *Naïve Bayesian Classifier* dan *Learning Vector Quantization*
5. Menganalisis kedua metode tersebut, yaitu dengan menyediakan prosedur pengujian *random stratified and repeated holdout*.

#### 4.3 Penentuan Model LVQ

Sebagai nilai awal, digunakan *epoch* sebesar 50, dengan *learning rate* cukup besar yaitu 0.3. Pada percobaan ini akan dilihat bagaimana pengaruh *learning rate reduction* pada tingkat akurasi dan stabilitas dari metode *Learning Vector Quantization*. Namun sebelumnya perlu diketahui bahwa pengurangan *learning rate* yang digunakan

pada penelitian ini adalah dengan prosedur sebagai berikut:

$$Lr = Lr - (r * Lr)$$

Pengujian dilakukan dengan memberikan nilai *learning rate reduction* yang melebihi *learning rate* yang digunakan sampai lebih kecil, yaitu dalam kisaran 0.7 – 0.1 dengan interval sebesar 0.1. Sehingga terdapat 7 kali percobaan dengan *learning rate reduction* yang berbeda-beda. Tabel 6 merupakan hasil pengujian awal yang telah dilakukan.

Tabel 6. Hasil Pengujian Awal LVQ

EPOH	LR	LRR	Akurasi Training	σ	Akurasi Testing	σ
50	0.3	0.7	90.33	6.94	84.4	6.79
		0.6	93.07	3.43	86.4	6.02
		0.5	94.13	2.1	87.6	4.5
		0.4	93.73	2.95	86.4	3.75
		0.3	91.47	7.28	84	11.04
		0.2	89.33	9.7	84	10.37
		0.1	72.4	36.19	67.4	32.82

Dapat dilihat bahwa dengan *learning rate* yang cukup besar (0.3) lalu diberikan *reduction* yang besar pula menyebabkan penurunan yang sangat signifikan membuat nilai *learning rate* pada iterasi setelahnya sangat kecil. Berbeda dengan penggunaan *reduction* yang kecil, membuat *learning rate* pada iterasi selanjutnya masih pada angka yang besar. *Learning rate* yang kecil pada iterasi setelahnya seperti yang terjadi pada *learning rate reduction* = 0.7 membuat tingkat akurasinya masih lebih baik daripada pada penggunaan *learning rate reduction* = 0.1. Selain akurasi, tingkat kestabilan pada *learning rate reduction* = 0.1 sangatlah buruk. Hal tersebut disebabkan nilai *learning rate* yang cenderung besar pada setiap iterasinya dikarenakan *reduction* yang kecil, padahal *learning rate* yang kecil dapat meningkatkan tingkat akurasi. Namun, nilai *learning rate* yang terlalu kecil pada tiap iterasi pun cenderung tidak baik. Karena pada *learning rate reduction* = 0,5 tingkat akurasinya lebih baik, baik itu pada data *training* maupun data *testing*.



Gambar 7. Grafik Hasil Pengujian Awal (LR=0.3)

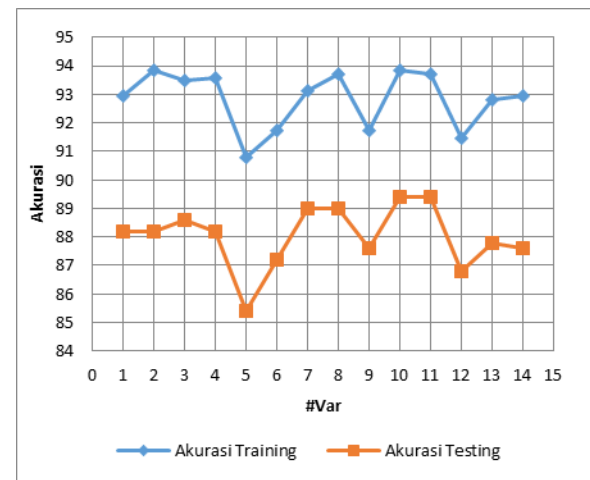
Setelah melakukan pengujian terhadap *learning rate reduction*, dapat dilihat bahwa memang penggunaan *learning rate* yang kecil dapat memaksimalkan performa metode *Learning Vector Quantization*. Namun perlu ditelaah lagi bagaimana dengan penggunaan *learning rate* yang lebih kecil.

Kemudian berdasarkan hasil diatas, dapat disimpulkan bahwa *learning rate reduction* yang terlalu besar juga tidak baik karena akan menyebabkan *learning rate* pada iterasi berikutnya secara signifikan terlalu kecil. Berikut merupakan pengujian yang telah dilakukan dengan menggunakan beberapa kombinasi *learning rate* dan *learning rate reduction*.

Tabel 7. Hasil Pengujian Varasi LR dan LRR

ID	EPOH	LR	LRR	Akurasi Training	$\sigma$	Akurasi Testing	$\sigma$
1	50	0.2	0.4	92.93	5.11	88.2	5.53
2	50	0.2	0.3	93.87	2.61	88.2	3.82
3	50	0.2	0.2	93.47	2.84	88.6	3.66
4	50	0.2	0.1	93.6	2.8	88.2	3.71
5	50	0.1	0.3	90.8	7.26	85.4	4.9
6	50	0.1	0.2	91.73	6.79	87.2	5.51
7	50	0.1	0.1	93.13	2.19	89	4.03
8	50	0.1	0.075	93.73	2.52	89	4.03
9	50	0.075	0.1	91.73	5.65	87.6	5.06
10	50	0.075	0.075	93.87	2.28	89.4	3.89
11	50	0.075	0.05	93.73	2.43	89.4	3.89
12	50	0.05	0.075	91.47	6.72	86.8	5.43
13	50	0.05	0.05	92.8	5.38	87.8	4.47
14	50	0.05	0.025	92.93	4.99	87.6	4.88

Pada baris berwarna putih tersebut dapat dilihat bahwa kombinasi *learning rate* dan *learning rate reduction* menghasilkan tingkat akurasi yang baik. Sehingga dapat disimpulkan bahwa nilai *learning rate* dan *learning rate reduction* terbaik dalam penyelesaian kasus klasifikasi penyakit kandungan ini adalah secara berurut-urut 0.075 dan 0.075.

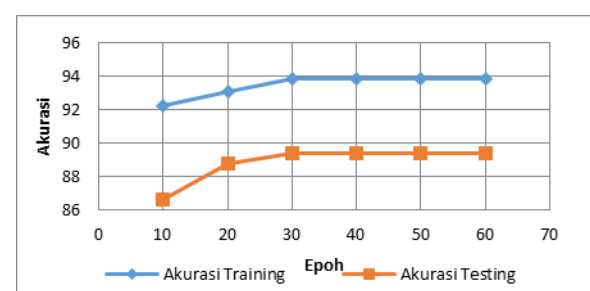


Gambar 8. Grafik Hasil Pengujian Varasi LR dan LRR

Kemudian yang menjadi hal penting pada pembelajaran menggunakan metode *Learning Vector Quantization* adalah jumlah maksimum *epoch* (iterasi). Karena jumlah iterasi ini akan berpengaruh terhadap berapa lama proses pembelajaran menggunakan metode *Learning Vector Quantization*. Maka dari itu dilakukan pengujian terhadap pengaruh jumlah *epoch* maksimum terhadap tingkat akurasi. Tabel 8 merupakan hasil dari pengujian tersebut.

Tabel 8. Pengujian Variasi EPOH

EPOH	Akurasi Training	s	Akurasi Testing	s
10	92.27	5.96	86.6	5.42
20	93.07	4.01	88.8	4.55
30	93.87	2.28	89.4	3.89
40	93.87	2.28	89.4	3.89
50	93.87	2.28	89.4	3.89
60	93.87	2.28	89.4	3.89



**Gambar 9.** Grafik Pengujian Variasi EPOCH

Pada hasil pengujian diatas terlihat bahwa jumlah iterasi berpengaruh pada tingkat akurasi. Hal ini terkait dengan penelitian yang dilakukan oleh Hermanenda 2013 [22] yang menyatakan bahwa jumlah epoch berpengaruh terhadap tingkat akurasi. Semakin tinggi jumlah epoch maksimum maka semakin tinggi pula akurasinya. Namun perlu dilihat bahwa ternyata terdapat batas dimana peningkatan jumlah epoch maksimum tidak berpengaruh terhadap tingkat akurasi. Pada epoch maksimum 10 – 30 memang terjadi peningkatan yang cukup signifikan. Namun dari epoch maksimum 30 – 60 ternyata tidak terjadi peningkatan pada tingkat akurasi. Maka dari itu dipilih batas dimana tidak terjadi peningkatan akurasi lagi, yaitu pada epoch maksimum 30. Sehingga lama waktu pembelajaran dapat ditekan.

Berdasarkan hasil pengujian diatas maka dapat disimpulkan bahwa kombinasi learning rate, learning rate reduction dan epoch maksimum adalah secara berturut-turut 0.075, 0.075, dan 30.

**4.4 Perbandingan Metode**

Setelah mengetahui performa maksimal dari metode Learning Vector Quantization, selanjutnya adalah membandingkan metode tersebut dengan pendekatan probabilistik Naïve Bayesian Classifier. Dimana pada metode Naïve Bayesian Classifier akan dilihat bagaimana pengaruh penerapan laplacian smoothing.

Perbandingan dilakukan dengan menggunakan 4 skenario simulasi. Pembeda masing-masing skenario adalah porsi data training dan testingnya. Sehingga dapat dilihat performa masing-masing metode pada keadaan dimana porsi data training kecil, sedang, hingga besar. Pada Tabel 9 merupakan hasil perbandingannya.

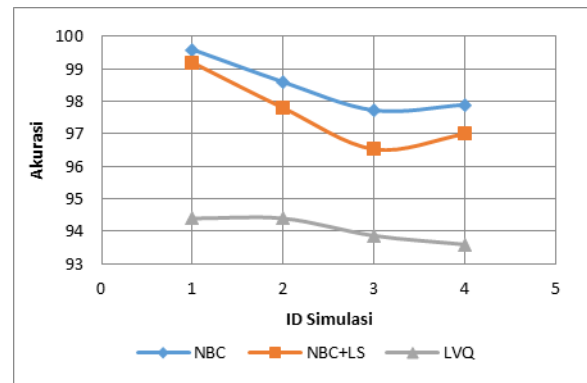
**Tabel 9.** Perbandingan Akurasi Training Metode

ID	Simulasi	Training					
		NBC		NBC+LS		LVQ	
		Akurasi	$\sigma$	Akurasi	$\sigma$	Akurasi	$\sigma$
1	20/80	99.6	1.26	99.2	1.69	94.4	8.26
2	40/60	98.6	1.35	97.8	1.75	94.4	6.24
3	60/40	97.73	1.55	96.53	2.11	93.87	2.28
4	80/20	97.9	0.88	97	1.05	93.6	1.42

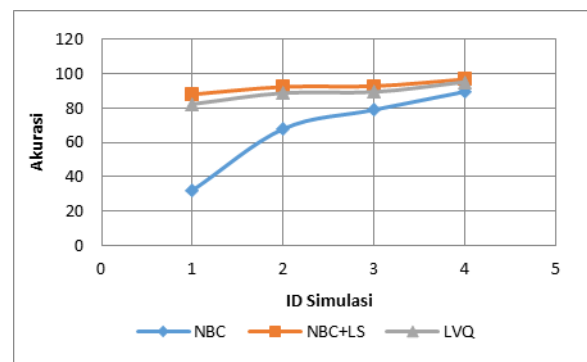
**Tabel 10.** Perbandingan Akurasi Training Metode

ID	Simulasi	Training					
		NBC		NBC+LS		LVQ	
		Akurasi	$\sigma$	Akurasi	$\sigma$	Akurasi	$\sigma$
1	20/80	32	9.26	88	2.54	82.4	6.77

2	40/60	67.87	8.7	92.4	2.52	88.8	7.23
3	60/40	79	6.34	92.8	2.15	89.4	3.89
4	80/20	89.6	6.02	96.8	3.68	95.2	4.54



**Gambar 9.** Grafik Perbandingan Metode Pada Akurasi Training



**Gambar 10.** Grafik Perbandingan Metode Pada Akurasi Testing

Selain akurasi, dilihat juga waktu pembelajaran masing-masing metode. Tabel 11 merupakan tabel perbandingan waktu pembelajaran (dalam satuan millisecond).

Perhitungan waktu pembelajaran dihitung dimulai dari awal proses pembelajaran hingga akhir pembelajaran sesuai dengan prosedur pada masing-masing metode dengan menggunakan fungsi yang disediakan bahasa pemrograman PHP dalam mengestimasi runtime.

**Tabel 11.** Hasil Perbandingan Kecepatan Metode

Simulasi Percobaan	NBC	NBC + LS	LVQ
20/80	448.9070 ms	449.1006 ms	289.9188 ms
40/60	483.6761 ms	483.8774 ms	325.6558 ms
60/40	510.5688 ms	510.7479 ms	403.7964 ms
80/20	546.7126 ms	546.8752 ms	533.9356 ms

**4.5 Analisa Hasil Pengujian**

Berdasar hasil pengujian yang dilakukan menggunakan 4 simulasi, ternyata pada kasus klasifikasi penyakit kandungan ini penggunaan *laplacian smoothing* sangat membantu *Naïve Bayesian Classifier* dalam memprediksi data *testing* terutama pada saat data *training* yang disediakan sangat kecil. Pada data *training* yang sangat kecil, yaitu data *training* 20% dan *testing* 80% *laplacian smoothing* sangat membantu menaikkan tingkat akurasi dalam metode *Naïve Bayesian Classifier*.

Tingkat akurasi dalam memprediksi data *testing* pada metode *Naïve Bayesian Classifier* sangat buruk, yaitu 32%. Hal ini dikarenakan keterbatasan data *training*, sehingga menyebabkan banyaknya nilai nol pada model probabilitas, sehingga *Naïve Bayesian Classifier* seringkali gagal dalam mengklasifikasikan berdasarkan gejala masukan. Sedangkan dengan porsi data yang sama, penggunaan *laplacian smoothing* mampu meningkatkan tingkat akurasi *Naïve Bayesian Classifier* dengan sangat signifikan, yaitu sebesar 88%. Selain itu juga terjadi kestabilan pada *Naïve Bayesian Classifier* dengan *laplacian smoothing* dibandingkan dengan tanpa *laplacian smoothing* yang cenderung tidak stabil.

Dengan diterapkannya *laplacian smoothing*, maka sebenarnya nilai probabilitas kondisional pada model probabilitas bukanlah nilai sebenarnya. Seperti yang dijelaskan sebelumnya bahwa tujuan *smoothing* adalah menaikkan probabilitas keluaran yang belum terobservasi, dan menurunkan probabilitas outcomes yang terobservasi [4]. Maka dari itu sangatlah wajar bahwa performa untuk data *training* menurun tipis. Jika dilihat dari hasil pengujian, terlihat bahwa akurasi *Naïve Bayesian Classifier* dengan *laplacian smoothing* pada data *training* sedikit dibawah *Naïve Bayesian Classifier* tanpa *laplacian smoothing*. Namun perbedaan tersebut sangat kecil sekali jika dibandingkan dengan keuntungan yang didapat dengan menerapkan *laplacian smoothing* pada *Naïve Bayesian Classifier*.

Berdasarkan hasil pengujian yang dilakukan dengan menggunakan 4 skenario diatas, dapat diambil kesimpulan bahwa penerapan *laplacian smoothing* sangat membantu untuk meningkatkan performa metode *Naïve Bayesian Classifier*. Maka, *Naïve Bayesian Classifier* dengan *laplacian smoothing* lebih baik daripada *Naïve Bayesian Classifier* tanpa *laplacian smoothing*.

Pada metode *Learning Vector Quantization*, tingkat akurasi pada data *training* yang sedikit memang cukup baik, bahkan lebih baik dibandingkan *Naïve Bayesian Classifier* tanpa *laplacian smoothing*. Namun jika dibandingkan

dengan *Naïve Bayesian Classifier* dengan *laplacian smoothing*, *Learning Vector Quantization* lebih tidak stabil. Ketidakstabilan tersebut terlihat terutama saat metode ini dihadapkan dengan terbatasnya data, dimana data *training* lebih sedikit daripada data *testing*. Secara keseluruhan dapat dikatakan bahwa *Naïve Bayesian Classifier* dengan *laplacian smoothing* performanya lebih baik daripada *Learning Vector Quantization*, dilihat dari tingkat akurasi dan stabilitasnya. Hal tersebut terlihat pada setiap simulasi metode *Naïve Bayesian Classifier* dengan *laplacian smoothing* akurasinya lebih tinggi daripada *Learning Vector Quantization*. Selain itu juga standar deviasi pada *Naïve Bayesian Classifier* dengan *laplacian smoothing* lebih kecil daripada standar deviasi pada *Learning Vector Quantization*.

Biasanya hal yang paling dipertimbangkan dalam penggunaan pendekatan jaringan syaraf tiruan adalah waktu pembelajaran yang lama. Namun pada penyelesaian kasus diagnosa penyakit kandungan ini tidak membutuhkan *epoch* yang besar untuk mencapai performa maksimal dari metode *Learning Vector Quantization*. Sehingga waktu pembelajaran menggunakan metode *Learning Vector Quantization* pada kasus ini sangat cepat.

Pada waktu kecepatan pembelajaran, dapat dilihat berdasar pada Tabel 11 ternyata kedua metode dapat menyelesaikan proses pembelajaran dengan waktu yang sangat cepat. Salah satu yang mempengaruhi kecepatan pembelajaran pada kedua metode ini adalah jumlah data *training*. Namun yang sangat terlihat sekali adanya pengaruh tersebut adalah pada metode *Learning Vector Quantization*. Hal tersebut dikarenakan proses perhitungan pada setiap satu *epoch* di *Learning Vector Quantization* adalah melibatkan seluruh data *training* yang ada, sehingga kecepatannya tergantung kepada seberapa banyak data yang dilatih. Penggunaan *laplacian smoothing* pada *Naïve Bayesian Classifier* berpengaruh sangat sedikit terhadap waktu eksekusi pembelajaran. Secara rata-rata memang metode *Learning Vector Quantization* memiliki kecepatan yang sedikit lebih cepat daripada *Naïve Bayesian Classifier*. Hal tersebut dikarenakan metode *Learning Vector Quantization* pada penyelesaian kasus ini ternyata tidak membutuhkan *epoch* yang banyak.

Maka berdasarkan penjelasan diatas dapat disimpulkan bahwa *Naïve Bayesian Classifier* dengan *laplacian smoothing* lebih handal daripada *Learning Vector Quantization* saat dihadapkan pada keempat simulasi yang diberikan, terutama pada saat kondisi dimana data *training* sangat terbatas. Namun kemampuan metode *Learning Vector Quantization* cukup bersaing pada saat keadaan

dimana data *training* yang tersedia besar, terlebih kecepatan pembelajaran *Learning Vector Quantization* dalam menyelesaikan kasus klasifikasi penyakit kandungan ini ternyata sedikit lebih cepat daripada *Naïve Bayesian Classifier*.

##### 5. SIMPULAN DAN SARAN

Penyelesaian masalah diagnosa penyakit kandungan dengan dua pendekatan berbeda telah dilakukan. Dimana pendekatan yang dilakukan adalah dengan menggunakan pendekatan probabilistik *Naïve Bayesian Classifier* dan jaringan syaraf tiruan *Learning Vector Quantization*. Pada pemilihan model metode *Learning Vector Quantization* didapatkan learning rate, learning rate reduction, dan epoch paling efektif yaitu secara berturut-turut 0.075, 0.075, dan 30.

*Naïve Bayesian Classifier dengan Laplacian Smoothing* memiliki akurasi paling baik dibandingkan dengan *Naïve Bayesian Classifier* tanpa *Laplacian Smoothing* dan *Learning Vector Quantization* pada seluruh skenario pengujian. Pada skenario data pelatihan 20%, 40%, 60% dan 80%, *Laplacian Smoothing* dapat membantu meningkatkan akurasi pengujian dengan sangat signifikan yaitu secara berturut-turut 88%, 92.4%, 92.8% dan 92.4% dibandingkan dengan *Naïve Bayesian Classifier* tanpa *smoothing* (32%; 67.8%; 79%; 89.6%) dan *Learning Vector Quantization* (82.4%; 88.8%; 89.4%; 95.2%).

Pada masalah kecepatan pembelajaran, kedua metode ini memiliki kecepatan pembelajaran yang cepat. Secara rata-rata berdasar kepada seluruh percobaan, *Learning Vector Quantization* sedikit lebih cepat dibandingkan *Naïve Bayesian Classifier* (dengan dan tanpa *Laplacian Smoothing*). Pada saat data pelatihan 20% waktu eksekusi pada metode *Learning Vector Quantization* jauh lebih cepat dibandingkan *Naïve Bayesian Classifier*. Namun sering bertambahnya data pelatihan, waktu eksekusi ketiga metode hampir sama.

Secara umum *Naïve Bayesian Classifier* dengan *Laplacian Smoothing* merupakan metode paling baik dalam menyelesaikan kasus klasifikasi penyakit kandungan.

Adapun saran yang dipertimbangkan untuk pengembangan penelitian ini antara lain adalah menelaah dependensi (keterkaitan) antar gejala sehingga dapat dilakukan peningkatan independensi dengan menghilangkan beberapa gejala, sehingga diharapkan dapat meningkatkan akurasi pada pendekatan probabilistik *Naïve Bayesian Classifier*, karena asumsi independensinya terpenuhi. Salah satu cara melihat keterkaitan kemunculan gejala pada tipe nilai *binary*/kategorial adalah dengan menggunakan

algoritma *apriori*, yaitu dengan mencari *support* dan *confidence* antar gejala.

Kemudian pada gejala yang memiliki *possible value* lebih dari dua (*non-binary*), ada kemungkinan tidak semua jenis kemunculannya terkait pada gejala lainnya. Sehingga perlu dilakukan pemecahan agar nilainya menjadi *binary*. Maka dari itu diterapkan *dummy variable* sehingga gejala tersebut terpecah menjadi gejala yang lebih spesifik dengan nilai *binary*. Dengan demikian penghilangan gejala berdasarkan identifikasi keterkaitan antar gejala menjadi lebih efektif.

##### 6. DAFTAR PUSTAKA

- [1] Prabhawaningrum, A. 2013. *Perbandingan Algoritma Levenberg-Marquardt dengan Backpropagation untuk Mendiagnosa Jenis Penyakit Kandungan*. Surakarta: Universitas Sebelas Maret.
- [2] Nurkhozin, A., Irawan, M. I., dan Mukhlash, I. 2011. Klasifikasi Penyakit Diabetes Mellitus Menggunakan Jaringan Syaraf Tiruan Backpropagation dan Learning Vector Quantization. *Prosiding Seminar Nasional Penelitian, Pendidikan, dan Penerapan MIPA*, Fakultas MIPA, Universitas Negeri Yogyakarta.
- [3] Bhuvanewari, R., Kalaiselvi, K. 2012. Naïve Bayesian Classification Approach in Healthcare Applications. *International Journal of Computer Science and Telecommunications* Vol. 3, No. 1, Januari 2012, hlm 106-110.
- [4] Arguello, J., 2013. *Naïve Bayes Text Classification* ([https://ils.unc.edu/courses/2013\\_fall/inls613\\_001/lectures/04-NaiveBayesClassification.pdf](https://ils.unc.edu/courses/2013_fall/inls613_001/lectures/04-NaiveBayesClassification.pdf). Diakses tanggal 3 Maret 2014 pukul 12.59). The University of North Carolina.
- [5] Jones, M. T. 2012. *Artificial Intelligence: A System Approach*. New Delhi: Infinity Science Press LLC.
- [6] Han, J. and Kamber, M. 2006. *Data Mining Concepts dan Techniques 2<sup>nd</sup> Edition*. San Francisco: Morgan Kaufmann Publishers.
- [7] Tan, P. N., Michael, S., and Kumar, V. 2006. *Introduction to Data Mining*. USA: Addison Wesley.Mining. USA:Addison Wesley.
- [8] Kusriani, L. T. 2009. *Algoritma Data Mining*. Yogyakarta: Andi.
- [9] Hermawan, A. 2006. *Jaringan Saraf Tiruan: Teori dan Aplikasi*. Yogyakarta: Andi.

- [10] Kusumadewi, S. 2003. *Artificial Intelligence (Teknik & Aplikasinya)*. Yogyakarta: Graha Ilmu.
- [11] Witten, I. H., Eibe, F., dan Mark A. H. 2011. *Data Mining: Practical Machine Learning Tools and Techniques 3<sup>rd</sup> edition*. Burlington: Elsevier.
- [12] Freitas, A. A. 2002. *Data Mining and Knowledge Discovery with Evolutionary Algorithms*. Verlag Berlin Heidelberg: Springer.
- [13] Hardibroto, B. R. 2005. Mioma Uteri. *Majalah Kedokteran Nusantara*, 38, 254-259.
- [14] Nugroho, T. 2010. *Buku Ajar Ginekologi untuk Mahasiswa Kebidanan*. Yogyakarta: Nuha Media.
- [15] Norwitz, E., & Shorge, J. 2008. *At a glance Obstetri & Ginekologi*. Jakarta: Erlangga.
- [16] Tjhay, F. 2009. Ancaman Penyakit Radang Panggul pada Infeksi Menular Seksual. *Majalah Kedokteran Damianus*, 8, 105-114.
- [17] Livoti, C., & Topp, E. 2006. *Menyingkap Tabir yang Selama Ini Tersembunyi Tentang Vagina*. Jakarta: Indeks.
- [18] Widayati, P., Ariyanto, A., & Lestari, W. 2009. Produksi Kit Immunoradiometricassay (IRMA) CA-125 untuk Deteksi Dini Kanker Ovarium. *Jurnal Ilmu Kefarmasian*, 7, 91-97.
- [19] Harahap, R. E. 1984. *Kanker Ginekologi*. Jakarta: Gramedia.
- [20] Mayangsari, D. 2010. Kanker Serviks. Dipetik Januari 10, 2013, dari <http://angsamerah.com/img/KankerServiks.pdf>
- [21] Yuwono, B., 2009. Perancangan dan Implementasi Jaringan Syaraf Tiruan Untuk Mendiagnosa Jenis Penyakit Kandungan. *TEKNOMATIKA*, vol. 1, no. 2, Januari 2009, hlm. 25-34.
- [22] Hermanenda G. A., Cholissodin I., dan Supianto A.A. 2013. Pengklasifikasian Kualitas Minuman Anggur Menggunakan Algoritma *Learning Vector Quantization* Berbasis Asosiasi. Universitas Brawijaya, Malang.