

# Perbandingan Metode LVQ Dan SVM Dalam Klasifikasi Produk Makanan Untuk Pengidap Penyakit *Stroke Non-Hemoragik*

YUSUP MIFTAHUDDIN<sup>1\*</sup>, IBNU FARHAN SHIHAB<sup>1</sup>

<sup>1</sup>Program Studi Informatika, Institut Teknologi Nasional Bandung  
Email: yusufm@itenas.ac.id

Received 28 01 2023 | Revised 04 02 2023 | Accepted 04 02 2023

## ABSTRAK

Salah satu resiko terjadi stroke non-hemoragik disebabkan oleh berat badan berlebih atau obesitas, sehingga upaya mengatasi masalah tersebut diperlukan sebuah sistem yang dapat menentukan klasifikasi produk makanan yang diperbolehkan untuk pengidap penyakit stroke non-hemoragik. Penelitian ini dilakukan dengan menggunakan metode LVQ dan SVM dengan mengukur tingkat akurasi, precision, recall dan error goal metode dalam klasifikasi produk makanan. Implementasi model klasifikasi LVQ dengan parameter learning rate 0.1 dan epoch 100 menghasilkan nilai akurasi 0.7797, precision 0.7568, recall 0.8750, dan error goal 0.2203. Sedangkan model klasifikasi SVM dengan model hyperparameter (default) menghasilkan nilai akurasi 0.9153, precision 0.9375, recall 0.9091, dan error goal 0.0847. Pada tahap eksperimen dilakukan pengujian terhadap model LVQ dan SVM, dimana LVQ menghasilkan pasangan parameter yang optimal pada learning rate 0,001 dan epoch 10 dengan nilai akurasi 0.9068. Sedangkan SVM dengan model kernel polynomial menghasilkan parameter yang optimal pada cost 100 dengan nilai akurasi 0.9661. Berdasarkan pengujian yang dilakukan, model SVM merupakan metode klasifikasi yang lebih baik dari LVQ.

**Kata kunci:** learning vector quantization, support vector machine, stroke.

## ABSTRACT

One of the risks of non-hemorrhagic stroke is caused by excess body weight or obesity, so an effort to overcome this problem requires a system that can determine the classification of food products that are allowed for people with non-hemorrhagic stroke. This research was conducted using the LVQ and SVM methods by measuring the level of accuracy, precision, recall and error goal method in the classification of food products. Implementation of the LVQ classification model with a learning rate of 0.1 and epoch 100 parameters resulted in an accuracy value of 0.7797, a precision of 0.7568, a recall of 0.8750, and an error goal of 0.2203. While the SVM classification model with the hyperparameter model (default) produces an accuracy value of 0.9153, a precision of 0.9375, a recall of 0.9091, and an error goal of 0.0847. In the experimental stage, tests were carried out on the LVQ and SVM models, where LVQ produced optimal parameter pairs at a learning rate of 0.001 and epoch 10 with an accuracy value of 0.9068. Whereas SVM with a polynomial kernel model produces optimal parameters at a cost of 100 with an accuracy value of 0.9661. Based on the tests conducted, the SVM model is a better classification method than LVQ.

**Keywords:** learning vector quantization, support vector machine, stroke.

## 1. PENDAHULUAN

Salah satu penyakit yang tergolong mematikan di Indonesia adalah Stroke dimana pada tahun 2014 hasil survei *Sample Registration Sistem* (SRS) dari Kemenkes menyatakan bahwa Stroke merupakan penyakit paling mematikan nomor satu di Indonesia sebanyak 21,1% kasus stroke berakhir pada kematian (**Adelia, 2021**). Stroke dibagi menjadi dua jenis yaitu stroke non-hemoragik dan stroke hemoragik. Dari kedua jenis tersebut yang paling banyak dijumpai pada masyarakat Indonesia adalah stroke non-hemoragik dengan mencapai 80% dari semua kasus stroke yang terjadi (**Persatuan Ahli Gizi Indonesia dan AsDI, 2019**).

Berdasarkan keterangan yang ditanyakan langsung kepada dokter ahli saraf yaitu dr.Tutwuri Handayani Sp.N, M.Kes yang bertugas di RSUD R Syamsudin SH, RS Hermina, dan RS Betha Medika. Salah satu resiko terjadi stroke non-hemoragik disebabkan oleh berat badan berlebih atau obesitas. Upaya mengatasi kesalahan dalam mengkonsumsi produk makanan adalah diperlukan sebuah sistem yang dapat menentukan klasifikasi produk makanan yang diperbolehkan bagi pengidap penyakit stroke non-hemoragik.

Mengacu pada contoh penelitian yang dilakukan oleh (**Budianita & Prijodiprodjo, 2013**) menggunakan metode LVQ1 dan LVQ3 berjudul Penerapan Learning Vector Quantization (LVQ) untuk Klasifikasi Status Gizi Anak. Penelitian ini berfokus pada klasifikasi status gizi anak berdasarkan indeks antropometri BB/TB dan beberapa faktor yang mempengaruhi status gizi. Berdasarkan hasil pengujian jumlah data latih pada LVQ1 dan LVQ3 menunjukkan bahwa algoritma LVQ1 mendapatkan nilai rata-rata akurasi sebesar 88% dan LVQ3 mendapatkan nilai rata-rata akurasi sebesar 95.2%. LVQ dapat memecahkan masalah melalui pembelajaran klasifikasi berbasis pola, dimana setiap unit output mewakili kategori tertentu berdasarkan jarak terdekat antara vektor masukan ke bobot yang bersangkutan (**Hilwa, 2021**).

Mengacu pada contoh penelitian yang dilakukan oleh (**Mulkan, 2021**) dengan menggunakan beberapa metode klasifikasi yang berjudul Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, *Random Forest*, SVM, dan Naïve Bayes. Penelitian ini bertujuan untuk membandingkan *performance* beberapa algoritma klasifikasi yaitu C4.5, Random Forest, SVM, dan Naive Bayes. Data penelitian berupa data peserta JISC yang berjumlah sebanyak 200 data. Data training berjumlah 140 (70%) dan data testing berjumlah 60 (30%). Hasil penelitian menunjukkan pada algoritma C4.5 didapatkan akurasi sebesar 86,67%. Pada algoritma Random Forest didapatkan akurasi sebesar 83,33%. Pada algoritma SVM didapatkan akurasi sebesar 95%. Pada algoritma Naive Bayes didapatkan akurasi sebesar 86,67%. Sehingga dapat disimpulkan akurasi algoritma paling tinggi terdapat pada algoritma SVM dan akurasi paling kecil terdapat pada algoritma Random Forest.

Mengacu pada contoh penelitian yang dilakukan oleh (**Yustina, 2015**) menggunakan metode SVM dan LVQ yang berjudul Unjuk Kerja Metode Support Vector Machine (SVM) Dengan Learning Vector Quantization (LVQ) Pada Aplikasi Pengenalan Wajah. Penelitian ini bertujuan untuk membandingkan dan menganalisis unjuk kerja metode SVM dengan LVQ. Pengujian dilakukan dengan 5 citra training dan 5 citra uji pada jumlah kelas yang berbeda yaitu 5, 10, 15, dan 20 kelas. Hasil pengujian terhadap banyaknya kelas, SVM memiliki unjuk kerja 8% lebih baik dari LVQ pada 5 kelas, dan LVQ 42% lebih baik dari SVM pada 15 kelas. Sehingga dapat di simpulkan metode SVM menunjukkan unjuk kerja lebih tinggi pada jumlah kelas yang kecil, namun pada kelas yang besar metode ini memiliki unjuk kerja lebih rendah dari LVQ. Sedangkan metode LVQ dengan semakin bertambahnya jumlah kelas memiliki unjuk kerja yang lebih baik dari SVM.

Berdasarkan data-data yang sudah dijelaskan, penelitian ini dilakukan dengan menggunakan metode LVQ yang memiliki tingkat akurasi tinggi dalam metode klasifikasi dilihat berdasarkan penelitian yang dilakukan **(Budianita & Prijodiprodjo, 2013)**, dan menggunakan metode SVM yang memiliki tingkat akurasi paling tinggi dalam metode klasifikasi dilihat berdasarkan penelitian yang dilakukan **(Mulkan, 2021)**.

Penelitian dilakukan dengan membandingkan metode LVQ dan SVM dalam mengklasifikasi produk makanan untuk pengidap penyakit stroke non-hemoragik. Perbandingan metode dilakukan dengan mencari pasangan parameter yang optimal pada metode LVQ dan menggunakan model polynomial pada metode SVM, dimana pengujian tersebut merupakan bentuk kebaruan atau perbedaan dari penelitian yang dilakukan sebelumnya oleh **(Yustina, 2015)**. Hasil pengujian akan diperlihatkan dalam bentuk perbandingan nilai akurasi, *precision*, *recall*, dan *error goal* dari metode LVQ dan SVM.

## 2. METODE PENELITIAN

### 2.1. Normalisasi

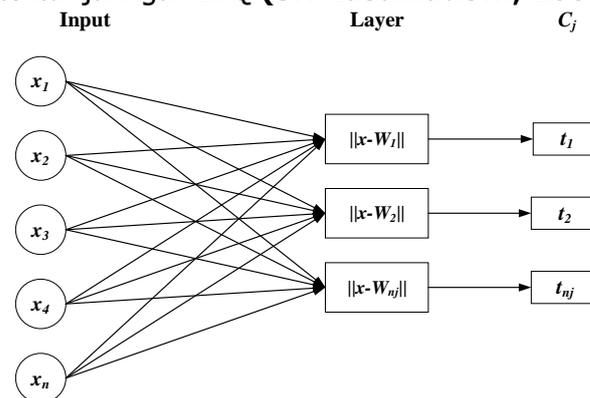
Normalisasi atau yang biasa disebut dengan proses penskalaan nilai atribut dari data sehingga bisa terletak pada rentang tertentu. Normalisasi data bertujuan untuk menempatkan nilai dalam range 0-1 dengan standarisasi. Untuk menskalakan data dalam jangkauan [0-1] menggunakan persamaan sebagai berikut **(Arifin & Sasongko, 2018)**:

$$\bar{x}_{ik} = \frac{\bar{x}_{ik} - \min(x_k)}{\max(x_k) - \min(x_k)} \quad (1)$$

Di mana  $\min(x_k)$  adalah nilai minimal dari semua data dan  $\max(x_k)$  adalah nilai maksimal dari semua data

### 2.2. Learning Vector Quantization (LVQ)

Learning Vector Quantization (LVQ) adalah suatu metode pelatihan untuk melakukan pembelajaran pada lapisan kompetitif yang terawasi (*supervised learning*) yang mana arsitektur jaringannya berlapis tunggal atau (*single layer*). Kelebihan lainnya yang dimiliki LVQ adalah mampu memberikan pelatihan terhadap lapisan-lapisan kompetitif sehingga secara otomatis dapat mengklasifikasikan vektor *input* yang berikan **(Nugroho, 2011)**. Berikut adalah gambar arsitektur jaringan LVQ **(Sri Kusumadewi, 2006)**.



**Gambar 1. Arsitektur LVQ**

Langkah-langkah algoritma pelatihan LVQ terdiri atas **(Wuryandari, 2012)**:

1. Inisialisasi vector referensi yang mewakili inisialisasi *learning rate* ( $\alpha$ ).
2. Masukkan data input (X) dan kelas target (T). Selama syarat henti (belum memenuhi target) tidak terpenuhi maka lakukan langkah ke 3-7.
3. Untuk setiap vektor input training  $x$ , kerjakan langkah 4- 5.
4. Dapatkan nilai  $J$  sedemikian hingga  $\|x-w_j\|$  minimum.
5. Perbarui  $W_j$  sebagai berikut:

Jika  $C_j = T$  maka  $W_j$  (baru) =  $W_j$  (lama) +  $\sigma [x - W_j$  (lama)] (2)

Jika  $C_j \neq T$  maka  $W_j$  (baru) =  $W_j$  (lama) -  $\sigma [x - W_j$  (lama)] (3)

6. Reduksi *learning rate*.
7. Tes kondisi stop atau kondisi akhir.

**2.2.1. Jarak Euclidean**

Menurut (Richard A. Johnson, 2016) jarak Euclid merupakan jarak geometris antara dua obyek data. Semakin dekat jarak maka semakin mirip suatu objek data tersebut. Bentuk umum jarak Euclid (d) dapat diperoleh dengan persamaan berikut (Jannah, 2010):

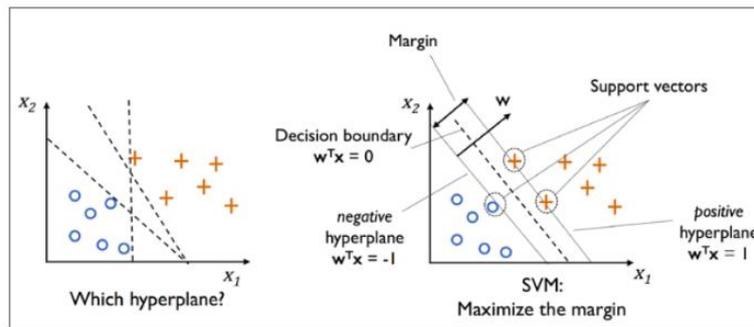
$$x = (x_1, x_2, \dots, x_n)$$

$$y = (y_1, y_2, \dots, y_n)$$

$$(x, y) = \sqrt{(y_1 - x_1)^2 + (y_2 - x_2)^2 + \dots + (y_n - x_n)^2} \tag{4}$$

**2.3. Support Vector Machine (SVM)**

Support Vector Machine (SVM) adalah suatu metode yang berusaha menemukan hyperplane yang terbaik pada input space. Prinsip dasar SVM adalah linear classifier yang selanjutnya dikembangkan agar dapat bekerja pada problem non-linear dengan memasukkan konsep kernel trick pada ruang kerja berdimensi tinggi (Nugroho, dkk, 2013). ini adalah contoh bagaimana SVM mencoba menemukan hyperplane terbaik untuk memisahkan kelas -1 dan +1 yang di tunjukkan pada Gambar 2.



**Gambar 2. Hyperplane SVM**

Hyperplane klasifikasi SVM dinotasikan dengan persamaan berikut:

$$\frac{1}{2} ||w||^2 = \frac{1}{2} (w_1^2 + w_2^2 + \dots + w_n^2) \tag{5}$$

Dari persamaan di atas di dapatkan persamaan berikut:

$$y_i (w_i \cdot x_i + b) \geq 1, i = 1, 2, 3, 4, \dots, n \tag{6}$$

Terdapat beberapa metode pembantu untuk meningkatkan hasil kinerja metode SVM, salah satunya adalah menggunakan metode kernel *trick*. Salah satu jenis yang digunakan dalam penelitian ini adalah kernel polynomial. Kernel polynomial merupakan fungsi kernel yang digunakan ketika data tidak terpisah secara linear. Dalam machine learning, kernel polynomial adalah fungsi kernel yang cocok untuk digunakan dalam SVM dan kernelisasi lainnya, di mana kernel mewakili kesamaan vektor sampel pelatihan dalam ruang fitur. Persamaan untuk fungsi kernel polynomial adalah:

$$K(x, xi) = 1 + \sum (x * xi) ^ d \tag{7}$$

**2.4. Confusion Matrix**

*Confusion matrix* digunakan untuk mengukur kemampuan sistem klasifikasi yang dibangun dengan mencari nilai *precision*, *recall* dan akurasi. Nilai *precision* adalah nilai sensitifitas atau nilai ketepatan sistem antara informasi yang diberikan oleh sistem untuk menunjukkan secara benar data kelas negatif atau kelas positif. Sedangkan nilai *recall* adalah nilai yang menunjukkan tingkat keberhasilan atau spesifisitas untuk mengetahui kembali sebuah informasi secara benar tentang data yang kelas negatif atau pun konten teks positif. Nilai

*precision* dan *recall* dapat dicari dengan menggunakan rumus persamaan 8 dan 9 sebagai berikut (**Challa, 2016**):

$$Precision = \left( \frac{TP}{FP + TP} \right) \times 100\% \quad (8)$$

$$Recall = \left( \frac{TP}{TP + FN} \right) \times 100\% \quad (9)$$

Nilai akurasi adalah nilai yang menunjukkan tingkat kedekatan antara nilai prediksi sistem dengan nilai prediksi manusia. Nilai akurasi dapat dicari dengan persamaan 10 dan error goal pada persamaan 11 (**Challa, 2016**).

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad (10)$$

$$Error = 100\% - Akurasi \quad (11)$$

**Tabel 1. Confusion Matrix**

		Kelas Sebenarnya	
		TRUE	FALSE
Kelas Prediksi	TRUE	TP	FP
	FALSE	FN	TN

Keterangan:

- True = Makanan yang Boleh dikonsumsi
- False = Makanan yang Tidak Boleh dikonsumsi
- TP (True Positive) = Jumlah makanan yang teridentifikasi Boleh
- FP (False Positive) = Jumlah makanan yang terprediksi Boleh tetapi sebenarnya makanan tersebut Tidak Boleh
- FN (False Negative) = Jumlah makanan yang terprediksi Tidak Boleh tetapi sebenarnya makanan tersebut Boleh
- TN (True Negative) = Jumlah makanan yang teridentifikasi Tidak Boleh

## 2.5. Deskripsi Dataset

Dataset yang digunakan mempunyai 588 baris data produk dan 4 kolom fitur yaitu protein, lemak, karbohidrat dan type dengan 2 label yaitu boleh atau tidak boleh. Dataset ini diambil dari Tabel Komposisi Pangan (**Kemenkes RI, 2018**). Berikut adalah beberapa contoh data produk makanan dari tabel komposisi pangan yang diperlihatkan pada Tabel 2.

**Tabel 2. Tabel Komposisi Makanan**

Nama Bahan	Komposisi Zat Gizi Makanan per 100 Gram BDD				BDD (data berat bahan yang dapat dimakan) (%)
	Energi	Protein	Lemak	Karbohidrat	
	(kal)	(g)	(g)	(g)	
Mie Bakso	114	5,3	3,0	16,4	100
Spaghetti	139	7,4	2,1	22,6	100
Keripik Kentang	448	2,7	18,8	67,1	100
Keripik Singkong	478	0,9	20,7	72,0	100
Shabu-Shabu	88	3,2	8,0	0,8	100
Sale Kesemek	296	1,1	0,6	71,4	100
Sale Pisang, Cilacap	295	2,3	2,1	66,9	100

Berdasarkan contoh data produk pangan pada Tabel 2 dibangun dataset yang akan digunakan dengan fitur/indikator protein, lemak, dan karbohidrat ditambah dengan satu fitur yaitu type yang berisi nilai label boleh atau tidak boleh yang didapatkan melalui proses perhitungan gizi ideal menggunakan metode perhitungan *harrist benedict*.

## 2.6. Perhitungan Harrist Benedict

Rumus Harris-Benedict merupakan salah satu rumus yang sering digunakan oleh ahli gizi. Rumus ini memperhitungkan usia, jenis kelamin, berat badan, tinggi badan, dan tingkat aktivitas (**Monika, 2021**). Dalam penelitian ini perhitungan *Harris Benedict* digunakan untuk mencari kebutuhan asupan gizi dari makanan yang dikonsumsi pengidap penyakit stroke seperti protein, lemak, dan karbohidrat yang cukup agar pengidap penyakit tidak mengonsumsi makanan dengan gizi yang lebih dari proporsinya. Berikut beberapa perhitungan *harrist benedict*:

1. Perhitungan Kalori Sementara (BEE)

Untuk rumus perhitungan kalori *Harris Benedict* adalah sebagai berikut :

Laki-laki :  $66,5 + (13,7 \times BB) + (5 \times TB) - (6,8 \times U)$

Perempuan :  $655 + (9,6 \times BB) + (1,8 \times TB) - (4,7 \times U)$

(12)

Keterangan:

**BB** = Berat Badan dalam kg ; **TB** = Tinggi Badan dalam cm ; **U** = Umur dalam tahun

2. Perhitungan Total Energi Expenditure (TEE)

TEE = BEE x Jenis Aktivitas

(13)

Nilai jenis aktivitas menurut (**Supriasa, 2001**) dapat dilihat dalam Tabel 3 sebagai berikut.

**Tabel 3. Level Aktivitas**

Kategori Aktivitas	Laki-Laki	Perempuan
Sangat Ringan	1,30	1,30
Ringan	1,65	1,55
Sedang	1,76	1,70
Berat	2,10	2,00

Level aktivitas dipengaruhi oleh gender, karena kebutuhan dan kondisi fisik laki-laki sedikit berbeda dengan perempuan. Adapun bentuk aktivitas fisik menurut Recommended Dietary Allowance (RDA) dapat dilihat dalam Tabel 4 berikut:

**Tabel 4. Aktifitas Fisik Menurut RDA**

Kategori Aktivitas	Kegiatan
Sangat Ringan	Duduk dan berdiri, melukis, menyetir mobil, mengetik, menyapu, menyetrika.
Ringan	Jogging, membersihkan rumah, mengasuh anak, golf, memancing
Sedang	Berlari, mencabut rumput, mencangkul, bersepeda, berselancar, menari
Berat	Mendaki, menebang pohon, menggali tanah, panjat tebing

**Sumber: (Recommended Dietary Allowance, 2000)**

3. Perhitungan Gizi Ideal

- Syarat Diet (**Neneng, 2019**):

1. Energi cukup yaitu 25-45 kkal/kg BB, pada fase akut energi diberikan 1100-1500 kkal/hari.

2. Protein cukup yaitu 0,8-1 gr/kg BB atau 10-15% dari kebutuhan energi total, apabila pasien berada dalam keadaan status gizi kurang.

3. Lemak cukup yaitu 20-25% dari kebutuhan energi total, diutamakan sumber lemak jenuh yaitu kurang dari 10% dari kebutuhan energi total. Untuk lemak omega 3 dianjurkan pada pasien stroke trombosis.

4. Karbohidrat cukup 60-70 % dari energi total, diutamakan karbohidrat kompleks.

Keterangan protein, lemak, karbohidrat per 1 gram (**Fenny, 2014**):

1 gr Protein = 4 kkal; 1 gr Lemak = 9 kkal; 1 gr Karbohidrat = 4 kkal

Dari syarat diatas menghasilkan perhitungan untuk asupan gizi ideal dengan persamaan berikut (Neneng, 2019):

$$\text{Protein} = \frac{15\% \times TEE}{4} = \dots \text{ gram}$$

$$\text{Lemak} = \frac{20\% \times TEE}{9} = \dots \text{ gram}$$

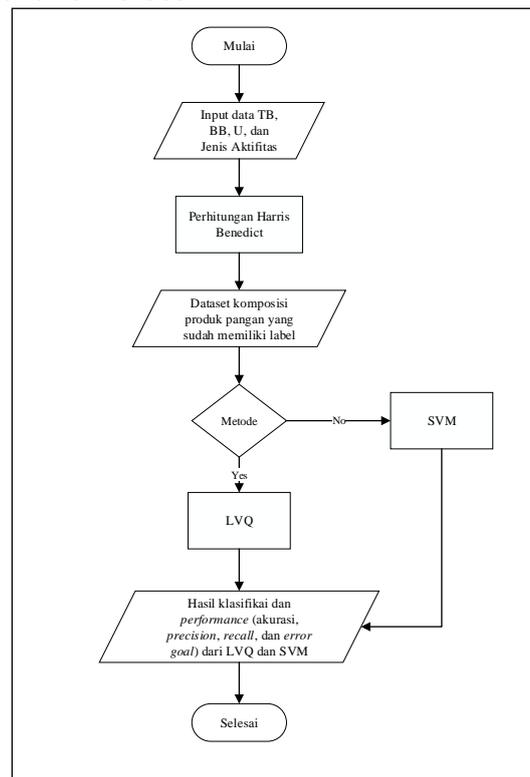
$$\text{Karbohidrat} = \frac{65\% \times TEE}{4} = \dots \text{ gram}$$

(14)

## 2.7. Perancangan Umum

Pada perancangan umum akan dijelaskan proses kerja dari sistem perbandingan metode LVQ dan SVM dalam menentukan status klasifikasi produk makanan yang diperbolehkan untuk pengidap penyakit stroke non-hemoragik melalui *flowchart*.

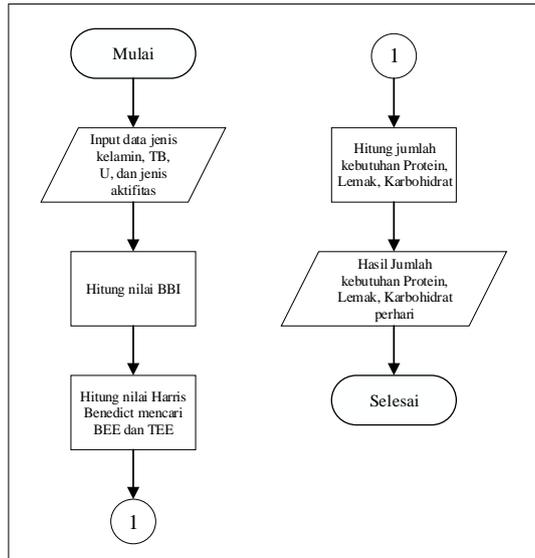
### 2.7.1. Flowchart Keseluruhan Sistem



Gambar 3. Flowchart Keseluruhan Sistem

Gambar 3 menunjukkan flowchart keseluruhan sistem klasifikasi produk makanan untuk pengidap penyakit stroke non-hemoragik dengan penjelasan sebagai berikut:

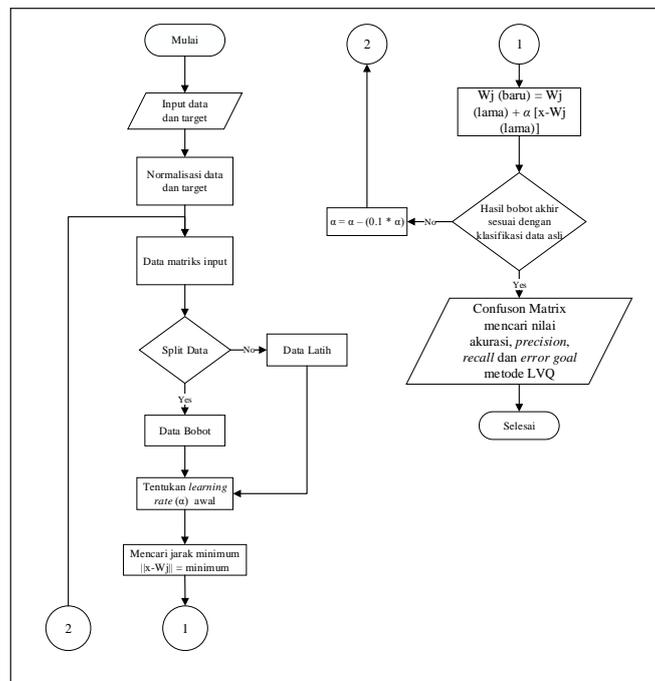
1. Data *input* pengidap penyakit stroke berupa data jenis kelamin, TB, U, dan jenis aktivitas.
2. Perhitungan *harrist benedict* digunakan untuk mencari kebutuhan gizi ideal pengidap penyakit stroke non-hemoragik berdasarkan data *input*.
3. Input dataset produk pangan yang sudah memiliki label berdasarkan kebutuhan gizi sebelumnya.
4. Memilih klasifikasi metode LVQ atau SVM.
5. Keluaran sistem berupa hasil klasifikasi berdasarkan metode yang dipilih sebelumnya serta *performance* berupa akurasi, *precision*, *recall*, dan *error goal*.



**Gambar 4. Perhitungan Harrist Benedict**

Pada Gambar 4 terdapat flowchart dari sub proses perhitungan harrist benedict berikut adalah penjelasannya:

1. Data *input* pengidap penyakit stroke berupa data jenis kelamin, TB, U, dan jenis aktivitas.
2. Dilakukan proses perhitungan mencari nilai BBI (Berat Badan Ideal) berdasarkan inputan.
3. Perhitungan *harrist benedict* dalam mencari nilai BEE (Perhitungan Kalori) dan TEE (Total Energi).
4. Menghitung jumlah kebutuhan Protein, Lemak, dan Karbohidrat berdasarkan nilai komposisi pada produk pangan dan TEE.
5. *Output* berupa jumlah kebutuhan ptotein, lemak, dan karbohidrat perhari dan pemberian label.

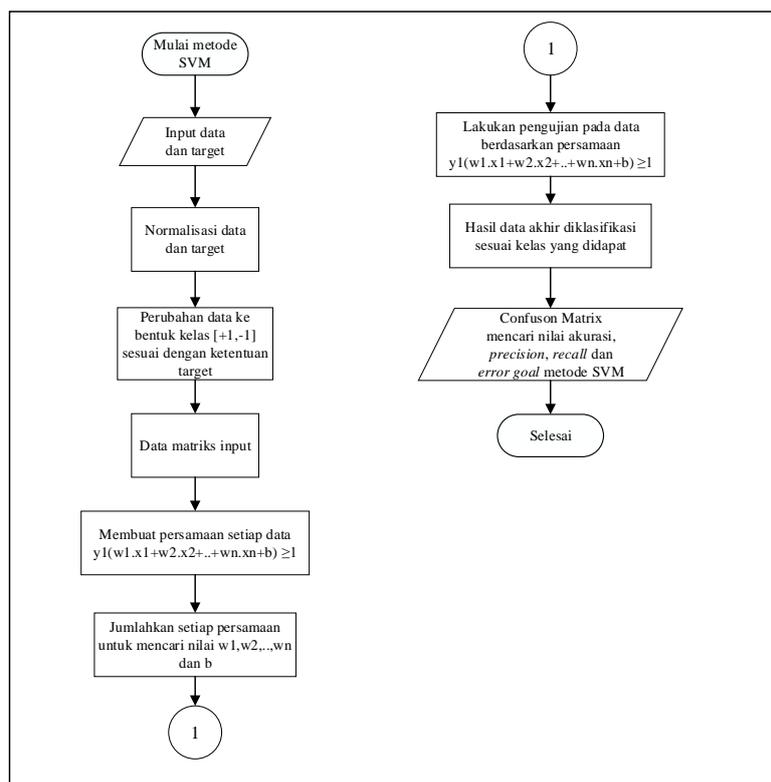


**Gambar 5. Flowchart Klasifikasi LVQ**

Pada Gambar 5 terdapat flowchart dari sub proses model klasifikasi LVQ berikut adalah penjelasannya:

*Perbandingan Metode LVQ dan SVM Dalam Klasifikasi Produk Makanan Untuk Pengidap Penyakit Stroke Non-Hemoragik*

1. Input data dan target berupa dataset produk pangan dan target berdasarkan perhitungan *harrist benedict*.
2. Normalisasi data digunakan untuk menempatkan nilai dalam range (0-1) dengan standarisasi.
3. Data matriks input merupakan dataset yang sudah berubah nilainya ke bentuk range (0-1).
4. Split data menjadi data bobot dan data latih.
5. Menentukan learning rate awal (0-1) yang digunakan untuk menghitung nilai koreksi bobot pada waktu proses training.
6. Mencari jarak minimum dengan rumus  $\|x-W_j\|$ .
7.  $W_j$  baru atau bobot baru akan didapatkan dengan rumus  $W_j$  (baru) =  $W_j$  (lama) +  $\alpha [x-W_j$  (lama)].
8. Hasil bobot akhir adalah klasifikasi status boleh/tidak boleh berdasarkan metode LVQ dan proses pencocokkan dengan data asli sebelumnya.
9. Jika hasil bobot akhir tidak sesuai dengan data asli maka akan dilakukan iterasi selanjutnya dengan mengupdate learning rate lama menjadi baru menggunakan rumus  $\alpha = \alpha - (0.1 * \alpha)$ .
10. Jika hasil bobot akhir sesuai dengan data asli maka dapat dilakukan proses *confusion matrix* untuk mencari nilai akurasi, *precision*, *recall*, dan *error goal*.



**Gambar 6. Flowchart Klasifikasi SVM**

Pada Gambar 6 terdapat flowchart dari sub proses model klasifikasi SVM berikut adalah penjelasannya:

1. Input data dan target berupa dataset produk pangan dan target berdasarkan perhitungan *harrist benedict*.
2. Normalisasi data digunakan untuk menempatkan nilai dalam *range* (+1, -1) dengan standarisasi.
3. Data matriks input merupakan dataset yang sudah berubah nilainya ke bentuk (+1, -1).

4. Membuat persamaan dari setiap data dengan rumus  $y_1(w_1.x_1 + w_2.x_2 + \dots + w_n.x_n + b) \geq 1$ .
5. Menjumlahkan setiap persamaan yang didapatkan untuk mencari nilai  $w_1, w_2, \dots, w_n$ , dan  $b$ .
6. Melakukan pengujian data berdasarkan persamaan yang sudah didapatkan sebelumnya.
7. Hasil data akhir diklasifikasi sesuai kelas yang didapat seperti boleh/tidak boleh.
8. Confusion matrix digunakan untuk mencari nilai akurasi, *precision*, *recall*, dan *error goal* dari metode SVM.

### 3. HASIL DAN PEMBAHASAN

#### 3.1. Penggunaan Dataset

Dataset yang digunakan pada penelitian ini berbentuk file dalam format csv dengan jumlah 588 data produk, 4 fitur/indikator yaitu protein, lemak, karbohidrat dan type dengan 2 label boleh atau tidak boleh. Pembagian dataset 80% untuk training dan 20% untuk testing. Berikut adalah detail pembagian dataset diperlihatkan pada Tabel 5.

**Tabel 5. Pembagian datasets training, dan testing**

Label	Training	Testing
<b>Boleh (0)</b>	240	64
<b>Tidak Boleh (1)</b>	230	54
<b>Jumlah</b>	470	118

Berdasarkan Tabel 6, pada data training berjumlah 470 data dengan label boleh (0) sebesar 240 data dan label tidak boleh (1) sebesar 230 data. Pada data testing berjumlah 118 data dengan label boleh (0) 64 data dan label tidak boleh (1) 54 data.

#### 3.2. Training Model

Berdasarkan penelitian yang dilakukan terdapat 2 metode yang digunakan sehingga pada tahapan training dibagi menjadi 2 proses yaitu training metode LVQ dan training metode SVM. Pada tahap training metode LVQ dilakukan menggunakan parameter *learning rate* = 0.1 dan epoch = 100. Sedangkan tahap training metode SVM menggunakan model kernel hyperparameter (default). Training metode LVQ menghasilkan akurasi 0.8362 sedangkan training metode SVM menghasilkan akurasi 0.9511.

#### 3.3. Testing Model

Berdasarkan penelitian yang dilakukan terdapat 2 metode yang digunakan sehingga pada tahapan testing dibagi menjadi 2 proses yaitu testing metode LVQ dan testing metode SVM. Pada tahapan testing atau pengujian model menggunakan 20% dari dataset yang ada seperti pada Tabel 6. Pengujian ini menghasilkan *confusion matrix* untuk mengukur performa model klasifikasi LVQ dan SVM. Proses testing juga menghasilkan tingkat akurasi, *precision*, *recall*, dan *error goal*. Performa model klasifikasi LVQ dan SVM tersebut dapat dilihat pada Tabel 6 dan Tabel 7.

**Tabel 6. Confusion Matrix Model LVQ dan SVM**

Model	TP	TN	FP	FN
LVQ	56	36	18	8
SVM	60	48	4	6

**Tabel 7. Performa Pengujian Model LVQ dan SVM**

Model	Accuracy	Precision	Recall	Error Goal
LVQ	0.7797	0.7568	0.875	0.2203
SVM	0.9153	0.9375	0.9091	0.0847

#### 4. KESIMPULAN

Pada proses pembentukan model klasifikasi LVQ dengan parameter *learning rate* = 0.1 dan *epoch* = 100 dihasilkan nilai akurasi sebesar 0.7797, *precision* sebesar 0.7568, *recall* sebesar 0.8750, dan *error goal* sebesar 0.2203. Sedangkan proses pembentukan model klasifikasi SVM dengan model *hyperparameter (default)* dihasilkan nilai akurasi sebesar 0.9153, *precision* sebesar 0.9375, *recall* sebesar 0.9091, dan *error goal* sebesar 0.0847.

Tahap eksperimen dilakukan pada pengujian model klasifikasi LVQ dan SVM, dimana LVQ menggunakan parameter *learning rate* (0.1, 0.5, 0.01, 0.05, dan 0.001) dan *epoch* (10, 50, 100, 250, dan 500) menghasilkan pasangan parameter yang optimal adalah *learning rate* = 0,001 dengan *epoch* = 10 dengan nilai akurasi 0.9068, *precision* 0.9077, *recall* 0.9219, dan *error goal* 0.0883. Sedangkan SVM menggunakan model kernel polynomial dengan parameter *cost* (1, 10, 100, 1000) menghasilkan parameter yang optimal adalah *cost* = 100 dengan nilai akurasi 0.9661, *precision* 0.9844, *recall* 0.9545, dan *error goal* 0.0339. Berdasarkan hasil pengujian yang dilakukan tersebut metode klasifikasi SVM lebih baik dari metode klasifikasi LVQ.

#### DAFTAR PUSTAKA

- Afifah, L. (2021, Januari 25). *2 Teknik Reduksi Dimensi Populer dengan Python*. Retrieved from ilmudatapy: <https://ilmudatapy.com/teknik-reduksi-dimensi/>
- AJIDARMA, M. (2019). APLIKASI PERHITUNGAN KEBUTUHAN KALORI DAN PERHITUNGAN KALORI DARI MAKANAN YANG DIKONSUMSI. 1-18.
- Arvianti, R. V. (2019). KLASIFIKASI PENYAKIT DIABETES MELLITUS TIPE 2 MENGGUNAKAN LEARNING VECTOR QUANTIZATION (LVQ). 1-56.
- A. S. Nugroho, A. B. Witarto, and D. Handoko, "SVM," 2003.
- ATHIFAH, H. (2021). IMPLEMENTASI PRINCIPAL COMPONENT ANALYSIS DAN LEARNING VECTOR QUANTIZATION UNTUK KLASIFIKASI GIZI MAKANAN ANAK DI MASA PANDEMI. 1-225.
- Cunningham, J. P., & Ghahramani, Z. (2015). *Linear dimensionality reduction: Survey, insights, and generalizations*. *Journal of Machine Learning Research*, 16, 2859–2900.
- DELSAL, I. (2018). PENERAPAN METODE HARRIS-BENEDICT DALAM MEMBANTU MENENTUKAN GIZI IDEAL DAN SARAN MAKANAN MENURUT STANDAR UKURAN RUMAH TANGGA. *Institutional Repository State Islamic University Of Sultan Syarif Kasim Riau*.
- Dito Putro Utomo, M. (2020). Analisis Komparasi Metode Klasifikasi Data Mining dan Reduksi Atribut Pada Data Set Penyakit Jantung. *MEDIA INFORMATIKA BUDIDARMA Volume 4, Nomor 2, April 2020, Page 437-444*, 437-444.
- Dr.Fenny Amalia M.Gizi. 2015. "Power Point tentang Karbohidrat, Lemak, Protein" (diakses pada 27 februari 2022)
- Elvia Budianita, W. P. (2013). Penerapan Learning Vector Quantization (LVQ) untuk Klasifikasi Status Gizi Anak. *IJCCS, Vol.7, No.2, July 2013*, 155-166.
- Kusumadewi S, d. H. (2006). Neuro Fuzzy : Integrasi Sistem Fuzzy dan Jaringan Syaraf. *Graha Ilmu, Yogyakarta*.
- MARHAMAH. (2019). IMPLEMENTASI ALGORITMA SUPPORT VECTOR MACHINE (SVM) DALAM MEMREDIKSI KELULUSAN JALUR MASUK PERGURUAN TINGGI BANDA ACEH. 1-65.
- Mulkan Azhari, Z. S. (2021). Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes. *JURNAL MEDIA INFORMATIKA BUDIDARMA, Volume 5, Nomor 2*, 640-651.
- Nanda, M. (2021, Juni 21). *Cara Menghitung Kebutuhan Kalori Harian Anda*. Retrieved from hellosehat.com: <https://hellosehat.com/nutrisi/cara-menghitung-kebutuhan-kalori/>

- Nurkhozin A, I. M. (2011). Klasifikasi Penyakit Diabetes Mellitus Menggunakan Jaringan Syaraf Tiruan Backpropagation dan Learning Vector Quantization. *Prosiding Seminar Nasional Penelitian, Pendidikan, dan Penerapan MIPA*.
- R. A. Permana and S. Sahara, "Metode SVM Sebagai Penentu Kelulusan Mahasiswa pada Pembelajaran Elektronik," vol. VII, no. 1, pp. 50–58, 2019.
- RI, K. K. (2018). Tabel Komposisi Pangan Indonesia 2017. *Kementerian Kesehatan RI*, 1-135.
- Rifwan Hamid1, M. T. (2017, Desember). Implementasi Learning Vector Quantization (LVQ) untuk Klasifikasi Kualitas Air Sungai. *Jurnal Pengembangan Teknologi Informasi dan Ilmu Komputer, Vol. 1, No. 12*, 1758-1763.
- SUFMELA, N. G. (2019). STUDI KASUS PROSES ASUHAN GIZI TERSTANDAR PADA PASIEN STROKE HEMORAGIK DAN NON HEMORAGIK RAWAT INAP DI RUANG ANGGREK RSUD PROF. DR. W. Z. JOHANES KUPANG. 1-137.
- Supriasa. Gizi dan Ruang Lingkupnya Dalam Memenuhi Kebutuhan Sehari-hari, Yogyakarta, 2001.
- T. Arifin, "Metode Data Mining Untuk Klasifikasi Data Sel Nukleus Dan Sel Radang Berdasarkan Analisa Tekstur," *Informatika*, Vol. Ii, No. 2, Pp. 425–433, 2015.
- Veratamala, A. (2021, Maret 25). *Cara Menghitung BMR: Kalori Minimum yang Dibutuhkan Tubuh*. Retrieved from hellosehat.com: <https://hellosehat.com/nutrisi/cara-menghitung-bmr/>
- Z. Hao, L. Shaohong, and S. Jinping, "Unit Model of Binary SVM with DS Output and its Application in Multi-class SVM," in *2011 Fourth International Symposium on Computational Intelligence and Design*, 2011, vol. 1, pp. 101–104, doi: 10.1109/ISCID.2011.34.
- Z. Yan, "A SVM model for data mining and knowledge discovering of mine water disasters," in *2010 8th World Congress on Intelligent Control and Automation*, 2010, pp. 2730–2734, doi: 10.1109/WCICA.2010.5554830.