

# **Laporan Tugas Akhir Project**

## **Jaringan Saraf Tiruan**

**Dosen Pengampu:**

Dr. Eng. Budi Darma Setiawan, S.Kom., M.Cs.



**Nama:**

Firman Maulana	(225150200111001)
Arion Syemael Siahaan	(225150207111060)
Muhammad Razi Al Kindi Nadra	(225150207111107)

**Program Studi Teknik Informatika**  
**Jurusan Teknik Informatika**  
**Universitas Brawijaya**  
**2024**

# 1. Penjelasan Dataset

Dataset MHEALTH (Mobile HEALTH) adalah kumpulan data mentah dengan 1.215.745 baris dan 14 kolom yang merekam gerakan tubuh dan tanda vital dari 10 relawan yang memiliki profil beragam saat melakukan 12 aktivitas fisik yang berbeda. Data ini dikumpulkan menggunakan sensor wearable Shimmer 2 yang ditempatkan di tiga lokasi tubuh, yaitu: Dada (Chest), Pergelangan tangan kanan (Right wrist), Pergelangan kaki kiri (Left ankle).

Dataset mhealth\_raw\_data.csv berisi. dan memiliki beberapa fitur, yaitu:

## 1. 12 kolom sensor

- Accelerometer left : alx, aly, alz.
- Accelerometer right : arx, ary, arz.
- Gyroscope left : glx, gly, glz.
- Gyroscope right : grx, gry, grz.

## 2. 12 jumlah Activity

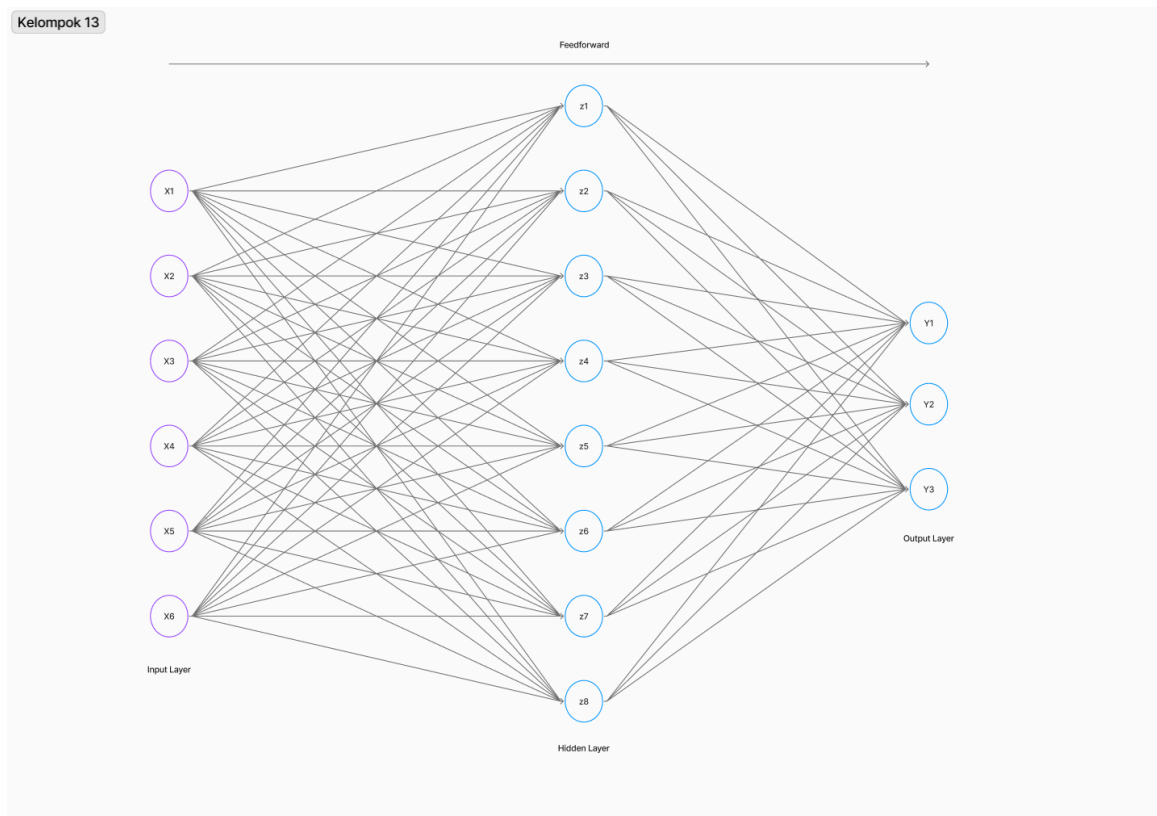
- L1: Berdiri diam (Standing still, 1 menit).
- L2: Duduk santai (Sitting and relaxing, 1 menit).
- L3: Berbaring (Lying down, 1 menit).
- L4: Berjalan (Walking, 1 menit).
- L5: Naik tangga (Climbing stairs, 1 menit).
- L6: Membungkuk ke depan (Waist bends forward, 20x).
- L7: Mengangkat lengan ke depan (Frontal elevation of arms, 20x).
- L8: Membungkuk lutut (Knees bending (crouching), 20x).
- L9: Bersepeda (Cycling, 1 menit).
- L10: Jogging (Jogging, 1 menit).
- L11: Berlari (Running, 1 menit).
- L12: Melompat maju & mundur (Jump front & back, 20x).

Dalam implementasinya, kami menggunakan target aktivitas sebagai berikut:

- Target Aktivitas : 1(standing), 2(Sitting) dan 4(Walking)
- Sensor yang Digunakan : 6 fitur gyroscope yang terdiri glx, gly, glz (kaki kiri), grx, gry, grz (lengan kanan).
- Total Data Setelah Pemrosesan: 92.160 entri data, kemudian digabungkan per 20 entri data dengan total akhir 4608 entri data.

Tujuan utama dataset ini adalah untuk mempelajari pengenalan aktivitas manusia (Human Activity Recognition) dengan menggunakan sensor gerakan dan data vital.

## 2. Penjelasan dan Gambar Arsitektur



### Struktur Arsitektur :

- **Input Layer:** Terdiri dari 6 input yang diperoleh dari menggabungkan setiap 20 entri data dimana setiap entri merepresentasikan 50z atau per 1 detiknya yang menghasilkan, kemudian dirata-ratakan berdasarkan fitur yang sesuai.
- **Hidden Layers:** Terdiri dari satu hidden layer dengan ukuran 64 neuron, dengan menggunakan fungsi aktivasi RELU untuk proses feedforward dan RELU derivative untuk proses backpropagation.
- **Output Layer:** 3 Output Layer dengan fungsi Softmax untuk klasifikasi multi-kelas.

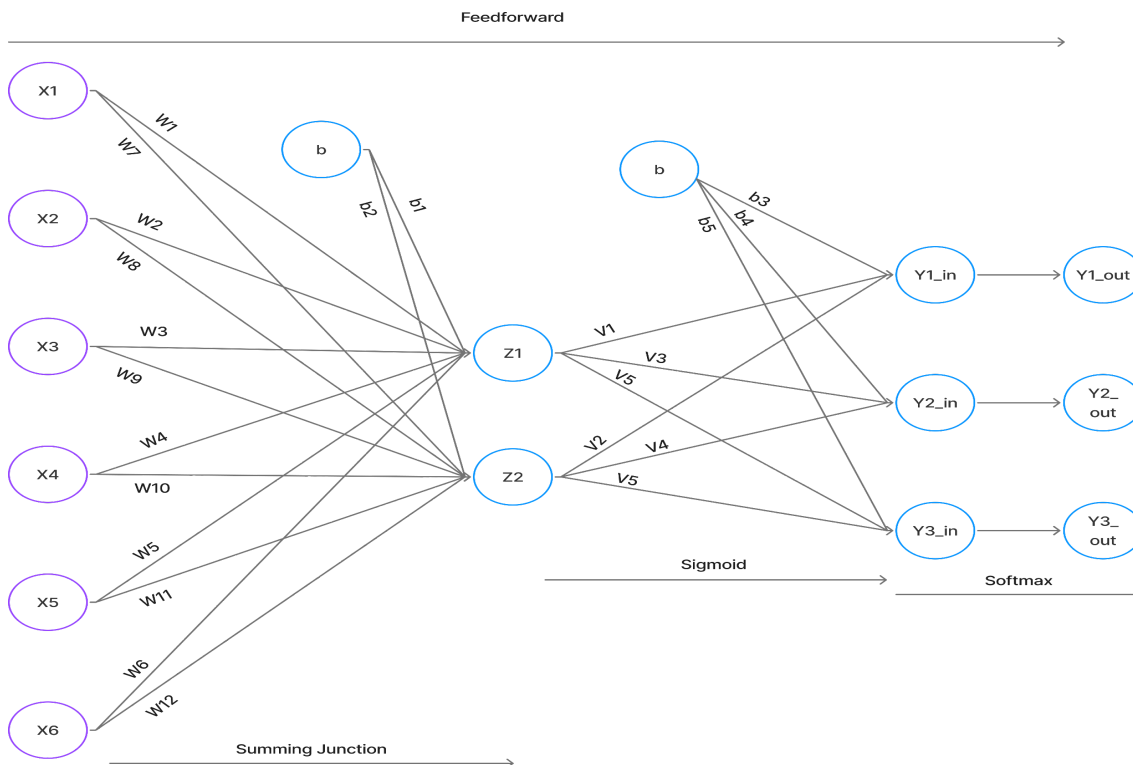
Alasan menggunakan 1 hidden layer karena dari dataset yang kami pakai nampaknya cukup sederhana dan sudah memiliki label. Jadi untuk menghindari overfitting dan meminimalkan resource yang ada, kami memilih untuk menggunakan 1 hidden layer.

### 3. Contoh Perhitungan Manual

#### 3.1 Inisialisasi Bobot dan Bias

- Inisialisasi semua nilai bobot, biasanya acak antara 0 dan 1
- W1 - W12, v1-v6, b1-b5 diinisialisasi random dari range 0,1 - 1

#### 3.2 FeedForward



Selama kondisi belum tercapai, lakukan:

- Setiap neuron menerima data latih x dan meneruskan ke hidden layer
- Setiap neuron hidden menghitung jumlah terboboti setiap input  $z_{j\_in} = \sum_i x_i w_{ij} + b_{ij}$
- Hitung aktivasi  $z_{j\_out} = f(z_{j\_in})$
- Untuk setiap neuron output, menghitung jumlah terboboti  $z_{j\_out}$  dengan

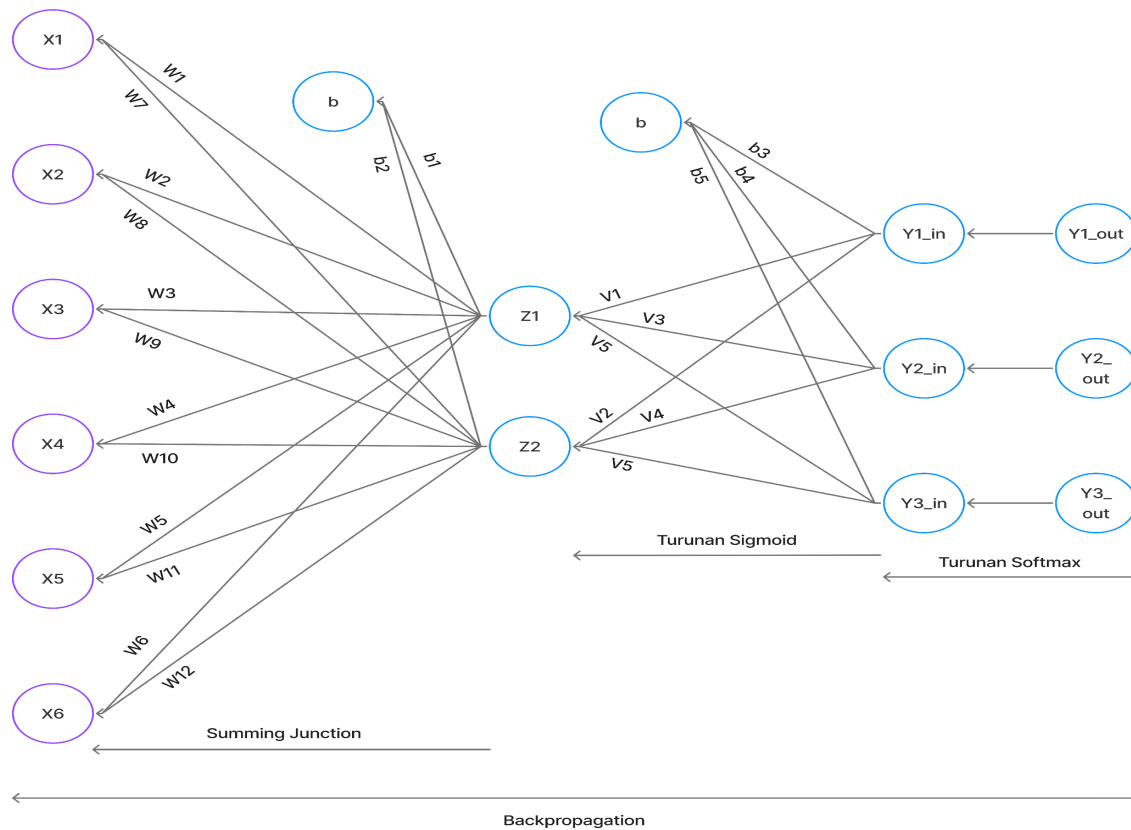
$$y_{k\_in} = \sum_j z_j v_{jk} + b_{jk}$$

- Hitung aktivasi  $y_{k\_out} = f(y_{k\_in})$

Proses pada perhitungan manualnya dalam excel :

- Proses summing junction dari setiap neuron yang menjadi input.
- Proses Feedforward dari Kolom W1 hingga Y3\_Out  
Pada hidden layer, digunakan fungsi aktivasi ReLU dengan rumus berikut dalam Excel: **(=MAX(0;Zn\_in))**
- Selanjutnya, untuk output layer digunakan fungsi aktivasi Softmax yang dirumuskan dalam Excel sebagai berikut:  
**(=EXP(Y1\_In)/(EXP(Y1\_In)+EXP(Y2\_In)+EXP(Y3\_In)))**

### 3.3 Backpropagation



- Hitung perubahan bobot dan bias yang menuju output neuron
  - $\Delta v_{jk} = \alpha \delta_k (z_{1out})$
  - $\Delta b_{jk} = \alpha \delta_k$
- Hitung perubahan bobot dan bias yang menuju hidden neuron
  - $\Delta w_{ij} = \alpha \sum_k (\delta_k v_{jk}) f'(z_{1in}) x_i$
  - $\Delta b_{ij} = \alpha \sum_k (\delta_k v_{jk}) f'(z_{1in})$
- di mana  $\delta_k = 2(t1 - y1out)(-1)f'(y1in)$  (tergantung loss function yang digunakan)

**Proses pada perhitungan manualnya dalam excel :**

- **Proses Backpropagation dari Kolom f'(y1\_in) hingga db2**

Pada proses backpropagation ini, perhitungan manual tidak mencakup nilai error. Untuk pembaruan bobot dan bias, digunakan rumus berikut dalam Excel:

- **Bobot**

**Rumus :  $(=H2 - (A16*BC2*A2) )$**

Keterangan: Bobot baru = Bobot lama – (Learning rate×Gradient bias×Input)

- Bias

$$\text{Rumus : } (=T2 - (A16*BO2) )$$

Keterangan: Bias baru = Bias lama – (Learning rate × Input)

### Fungsi Aktivasi Backpropagation:

- Turunan Softmax:

$$(\text{=IF}(G2 < \text{AL2}; \text{AL2} * (1 - \text{AL2}); \text{AL2} * (-\text{AL2})))$$

Keterangan: Menghitung derivatif Softmax berdasarkan output yang sesuai.

- Turunan RELU:

$$(\text{=IF}(\text{Zn\_in} > 0; 1; 0))$$

Keterangan: Menghitung derivatif fungsi aktivasi RELU.

## 3.4 Update Bobot dan Bias

### Rumus:

- $w' = w - \Delta w$
- $v' = v - \Delta v$
- $b' = b - \Delta b$

Semua proses di atas dilakukan sampai model mencapai maksimal epoch yang ditetapkan.

Hasil Perhitungan Excel:

w1	w2	w3	w4	w5	w6	w7	w8	w9	w10	w11	w12	b1	b2
0,3	0,3	0,3	0,3	1	0,9	0,5	0,7	0,6	1	0,3	0,2	0,6	0,7
0,300000000	0,300000000	0,300000000	0,300000000	1,000000000	0,900000000	0,500000000	0,700000000	0,600000000	1,000000000	0,300000000	0,300000000	0,600000000	0,700000000
0,300000000	0,300000000	0,300000000	0,300000000	1,000000000	0,900000000	0,500000000	0,700000000	0,600000000	1,000000000	0,300000000	0,300000000	0,600000000	0,700000000
0,300000000	0,300000000	0,300000000	0,300000000	1,000000000	0,900000000	0,500000000	0,700000000	0,600000000	1,000000000	0,300000000	0,300000000	0,600000000	0,700000000
0,300000000	0,300000000	0,300000000	0,300000000	1,000000000	0,900000000	0,500000000	0,700000000	0,600000000	1,000000000	0,300000000	0,300000000	0,600000000	0,700000000
0,300017582	0,299970473	0,299981645	0,300006140	0,999917867	0,900041570	0,500048551	0,699885849	0,599939177	1,000033910	0,299959175	0,300022959	0,600114577	0,700189839
0,300033269	0,299943504	0,299964921	0,300011844	0,999842974	0,900079278	0,500095268	0,699773417	0,599879416	1,000067881	0,299919032	0,299981629	0,600219287	0,700376932
0,300050170	0,299914562	0,299946772	0,300018060	0,999762314	0,900120243	0,500142829	0,699659420	0,599818141	1,000102865	0,299878184	0,299942077	0,600331826	0,700566928
0,300067103	0,299885713	0,299928821	0,300024178	0,999681734	0,900161235	0,500190834	0,699544957	0,599757085	1,000137553	0,299837076	0,299901411	0,600443718	0,700757234
0,300212733	0,299668391	0,299867301	0,299860520	0,999140868	0,900787175	0,500190834	0,699544957	0,599757085	1,000137553	0,299837076	0,299837076	0,601368878	0,700757234
0,300445116	0,299443058	0,300087873	0,299597015	0,998339924	0,901825310	0,500444446	0,699200402	0,600046058	0,999561863	0,299679634	0,300063423	0,602842626	0,701722577
0,300800474	0,299057475	0,300220039	0,299262757	0,997142267	0,903082128	0,500581841	0,698991414	0,600107401	0,999303001	0,299596184	0,299776579	0,604745649	0,702164309

v1	v2	v3	v4	v5	v6	b3	b4	b5
0,4	0,2	0,5	0,3	0,7	0,3	0,8	1	0,5
0,400000000	0,200000000	0,500000000	0,300000000	0,700000000	0,300000000	0,800000000	1,000000000	0,500281283
0,400000000	0,200000000	0,500000000	0,300000000	0,700000000	0,300000000	0,800000000	1,000000000	0,500562586
0,400000000	0,200000000	0,500000000	0,300000000	0,700000000	0,300000000	0,800000000	1,000000000	0,500843909
0,400000000	0,200000000	0,500000000	0,300000000	0,700000000	0,300000000	0,800000000	1,000000000	0,501125252
0,400046814	0,200079037	0,500055704	0,300112854	0,700016228	0,300023484	0,803333059	1,003172780	0,501785477
0,400089464	0,200156764	0,500106458	0,300223848	0,700030995	0,300046546	0,806665290	1,006345087	0,502444824
0,400135429	0,200235896	0,500161149	0,300336834	0,700046898	0,300070003	0,809997314	1,009516887	0,503103687
0,400181125	0,200315188	0,500215522	0,300450045	0,700062692	0,300093478	0,813328837	1,012688195	0,503761874
0,401274190	0,200315188	0,500215522	0,300450045	0,700441674	0,300093478	0,820680975	1,012688195	0,505219029
0,403463674	0,200879357	0,502865612	0,301271664	0,701226318	0,300266999	0,828074737	1,019867265	0,506737007
0,404984479	0,201061809	0,504713552	0,301538435	0,701756034	0,300321655	0,835438912	1,027046713	0,508212847

Link sheet: [📄 contoh perhitungan manual](#)

## 4. Hasil Analisis Performa Model

### 4.1 Pengaturan parameter:

- **Proporsi Data:** 70% data digunakan untuk pelatihan dan 30% data digunakan untuk pengujian.
- **Hidden layer:** 64 neuron
- **Learning Rate:** 0.1
- **Jumlah Epochs:** 50 epoch
- **Loss Function:** Cross-Entropy Loss, yang sesuai untuk klasifikasi multi-kelas.
- **Fungsi Aktivasi:**
  1. RELU digunakan di hidden layer karena kemampuannya memperkenalkan non-linearitas, kemudian RELU derivative untuk proses backpropagation nya.
  2. Softmax digunakan di lapisan output untuk mengubah output menjadi probabilitas untuk setiap kelas, kemudian Softmax derivative untuk proses backpropagation nya.

### 4.2 Hasil Evaluasi

- **Normalisasi:**

Data yang kita gunakan menggunakan normalisasi MinMax Scaler yang bertujuan untuk mengubah nilai-nilai dalam dataset ke dalam rentang tertentu (biasanya antara 0 dan 1) tanpa mengubah hubungan proporsional antara data. Normalisasi ini membantu algoritma pembelajaran mesin atau jaringan saraf untuk berkinerja lebih baik.
- **Jumlah Data Uji**

Data uji yang digunakan untuk evaluasi model terdiri dari 1383 sampel. data ini terbagi dalam tiga kelas aktivitas yang berbeda yaitu Standing (Berdiri), Sitting (Duduk), dan Walking (Berjalan). Berikut adalah distribusi jumlah data per kelas:

  - Standing (Berdiri): 472 data
  - Sitting (Duduk): 460 data
  - Walking (Berjalan): 451 data
- **Prediksi Data Uji**

Setelah pelatihan model selesai, selanjutnya kami melakukan prediksi pada data uji menggunakan model yang telah dilatih. Hasil Prediksi kemudian didekode untuk mendapatkan label kelas dengan one-hot encoding.
- **Akurasi Model: 98.77%**

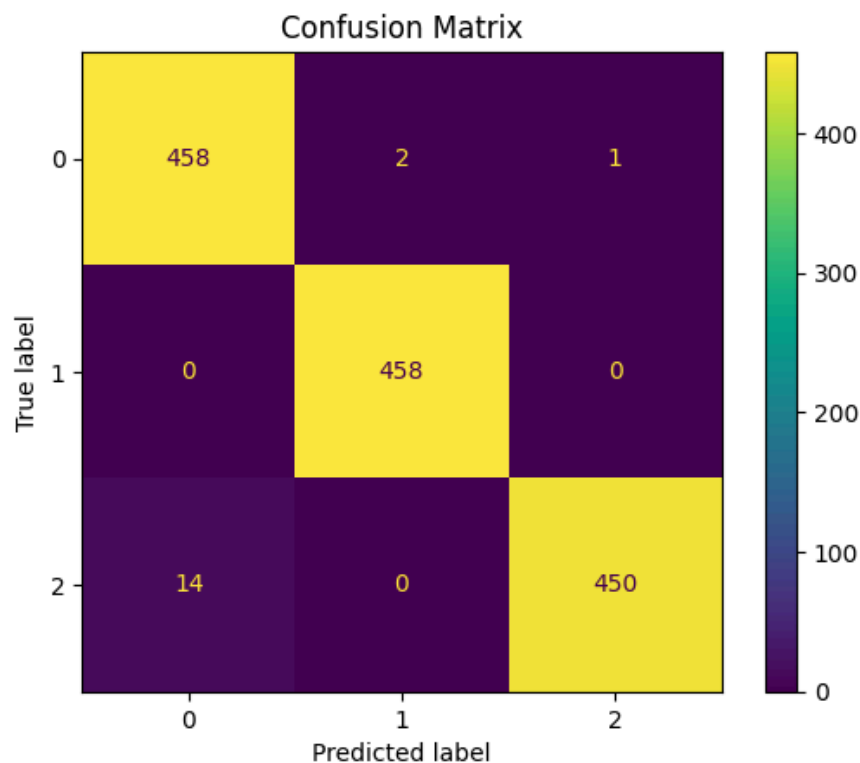
Akurasi menunjukkan bahwa model berhasil memprediksi dengan benar sekamir 99.06% data pengujian. Ini merupakan performa yang sangat baik, mengindikasikan kemampuan model untuk mengenali pola data pelatihan dan menggeneralisasi pada data pengujian dengan tingkat kesalahan yang sangat kecil.



- **F1-Score (Keseluruhan): 0.9877**  
F1-Score merupakan rata-rata harmonis antara precision dan recall. Dengan skor **0.9877**, model menunjukkan keseimbangan yang sangat baik antara kedua metrik ini, memastikan performa yang konsisten di semua kelas.
- **F1-Score per Kelas:**
  - **Standing (Berdiri): 0.9817**  
Model memiliki performa yang sangat baik untuk mengenali aktivitas ini, meskipun ada beberapa kesalahan minor yang menyebabkan prediksi salah ke kelas lain.
  - **Sitting (Duduk): 0.9978**  
Hampir sempurna. Model mengenali aktivitas ini dengan tingkat kesalahan yang sangat rendah.
  - **Walking (Berjalan): 0.9836**  
Meskipun performanya sedikit lebih rendah dibandingkan kelas "Sitting," model tetap mampu mengenali aktivitas ini dengan akurasi tinggi.
- **Precision dan Recall (Keseluruhan):**
  - **Precision: 0.9879 (98.79%)**  
Precision yang tinggi menunjukkan bahwa ketika model memprediksi suatu aktivitas (misalnya, "Walking"), kemungkinan besar prediksi tersebut benar. Hal ini berarti model jarang menghasilkan prediksi positif palsu (False Positives).
  - **Recall: 0.9878 (98.78%)**  
Recall yang tinggi menunjukkan bahwa model mampu mengenali hampir semua instance dari setiap aktivitas, dengan tingkat data yang terlewat (False Negatives) yang sangat rendah.
- **Precision dan Recall per Kelas:**
  - **Standing (Berdiri):**
    - **Precision: 97.03%**  
Beberapa data dari kelas lain diprediksi sebagai "Standing," tetapi sebagian besar prediksi tetap akurat.
    - **Recall: 99.35%**  
Hampir semua data "Standing" dikenali dengan benar, dengan hanya sebagian kecil data yang salah diklasifikasikan.
  - **Sitting (Duduk):**
    - **Precision: 99.56%**  
Prediksi aktivitas "Sitting" hampir selalu benar.
    - **Recall: 100%**  
Semua data "Sitting" berhasil dikenali tanpa kesalahan sama sekali.
  - **Walking (Berjalan):**
    - **Precision: 99.78%**  
Ketika model memprediksi aktivitas ini, hampir semua prediksi benar.
    - **Recall: 96.98%**  
Beberapa data "Walking" salah diklasifikasikan sebagai "Standing," tetapi secara keseluruhan performa tetap sangat baik.



### 4.3 Confusion Matrix:



- 0: Standing (Berdiri)
- 1: Sitting (Duduk)
- 2: Walking (Berjalan)

Setiap elemen dalam matriks mewakili jumlah prediksi model dibandingkan dengan label sebenarnya. Berikut adalah interpretasi elemen-elemen utama dalam matriks:

#### 1. Diagonal utama (True Positive):

Di diagonal utama menunjukkan jumlah prediksi benar (*True Positives*):

- (0,0): 458 data Standing diprediksi benar sebagai Standing.
- (1,1): 458 data Sitting diprediksi benar sebagai Sitting.
- (2,2): 450 data Walking diprediksi benar sebagai Walking.

#### 2. Off-diagonal (Misclassifications):

Elemen di luar diagonal utama menunjukkan prediksi salah (*False Positives* dan *False Negatives*):

- (0,2): 2 data Standing salah diprediksi sebagai Walking.
- (2,0): 14 data Walking salah diprediksi sebagai Standing.
- (0,1): 1 data Standing salah diprediksi sebagai Sitting.
- Tidak ada kesalahan prediksi untuk kelas Sitting (kolom dan baris 1 selain diagonal adalah nol).

## 4.5 Kesimpulan

Model memiliki performa yang sangat baik untuk tugas klasifikasi aktivitas fisik dengan:

- Akurasi: **98.77%**
- Precision dan Recall secara makro di atas **98%**, menunjukkan bahwa model dapat membuat prediksi akurat di setiap kelas.
- Performa sempurna pada kelas "Sitting" tanpa kesalahan prediksi.
- Kesalahan minor pada kelas "Standing" dan "Walking" disebabkan oleh kemiripan pola data antar kelas.

Model ini sangat cocok untuk diterapkan dalam skenario nyata, seperti aplikasi pemantauan aktivitas fisik atau deteksi kesehatan, dengan tingkat kesalahan yang sangat kecil dan keandalan yang tinggi.