# Body Postures and Movements classification with Neural Network

Hasan MUTLU
*Hacettepe University*
*Computer Engineering Department*
Ankara, Turkey
hasanmutlu9@gmail.com

***Abstract*—In this work, We tried to develop a Artificial Neural Network to predict body postures and movements with wearable accelerometers dataset which collected by Pontifical Catholic University of Rio de Janeiro. To improve accuracy of the constructed neural network, Several optimization algorithms are used and compared with each other.**

***Index Terms*—Artificial Neural Network, Multi Classification, Wearable accelerometers dataset**

## I. Introduction

Nowadays, With the development of technology, Human activity recognizing is used in many areas such as sport activities and health care systems so on.

Two approaches are commonly used for detection human body postures and movements, Computer vision or wearable devices can be used but these approaches have some difficulties.For example, if you are using computer vision approach, you need to have a camera or if you are using wearable devices, you need to calibrate sensors of devices. In this work, We used wearable accelerometers dataset to detect five movements and postures (Sitting, Sitting down, Standing, Standing up and Walking).

## II. Related Works

### Accelerometers' Data Classification of Body Postures and Movements [1]

This research uses wearable accelerometers' data and for classification Ross Quinlan's [2] C4.5 decision tree was used in connection with the AdaBoost ensemble method [3]. Accuracy of this approach is approximately 98%.

### Multisensor Data Fusion for Physical Activity Assessment [4]

This research presents a sensor fusion method for assessing physical activity of human subjects, based on support vector machines. Specifically, acceleration and ventilation measured by a wearable multisensor device on 50 test subjects performing 13 types of activities of varying intensities are analyzed, from which activity type and energy expenditure are derived.The results show that the method correctly recognized the 13 activity types 88.1% of the time.

### Detection of daily movements from data collected with two tri-axial accelerometers [5]

This research presents a design for a classifier using detection and classification of eight daily movements data collected with two tri-axial accelerometers, one mounted on the right part of the hip and the other one mounted on the lower part of the right leg. This classifier gave good accuracy of 99.8% in controlled laboratory experiments, in which four healthy subjects carried out a set of eight basic movements.

## III. Dataset

In this work, We used same dataset of the article Accelerometers' Data Classification of Body Postures and Movements. The accelerometers were respectively positioned in the waist, left thigh, right ankle, and right arm. This dataset was collected during 8 hours of activities, 2 hours with each one of the 4 subjects: 2 men and 2 women, all adults and healthy.

| Genre | Age | Height | Weight | Instances |
|---|---|---|---|---|
| Female | 46 y.o | 1,62m | 67kg | 51.577 |
| Female | 28 y.o | 1,58m | 53kg | 49.797 |
| Male | 31 y.o | 1,71m | 83kg | 51.098 |
| Male | 75 y.o | 1,67m | 67kg | 13.161 |

Table 1. The profile of each subject

The Dataset consist of 17 dimensional data which are gender, age, height, weight, body mass index and x,y,z angle values for each accelerometers.

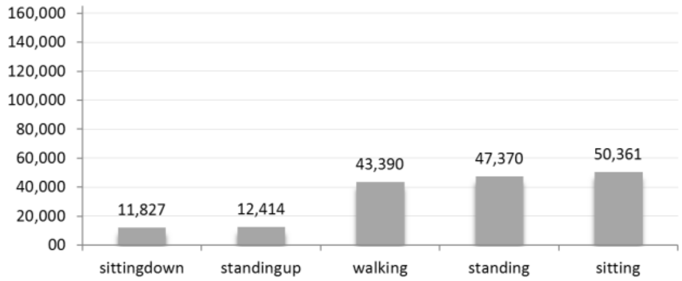


Fig. 1. Frequency of classes between collected data

## IV. Solution

First of all, To adapt the dataset to neural network,class names are converted to 5 dimensional integer array which consist of 0's and 1's and genre names are converted to 0 for Woman and 1 for Man.

| **Class** | | | | | |
|---|---|---|---|---|---|
| Sitting | 1 | 0 | 0 | 0 | 0 |
| Sitting down | 0 | 1 | 0 | 0 | 0 |
| Standing | 0 | 0 | 1 | 0 | 0 |
| Standing up | 0 | 0 | 0 | 1 | 0 |
| Walking | 0 | 0 | 0 | 0 | 1 |

Table 2. Output value of each classes

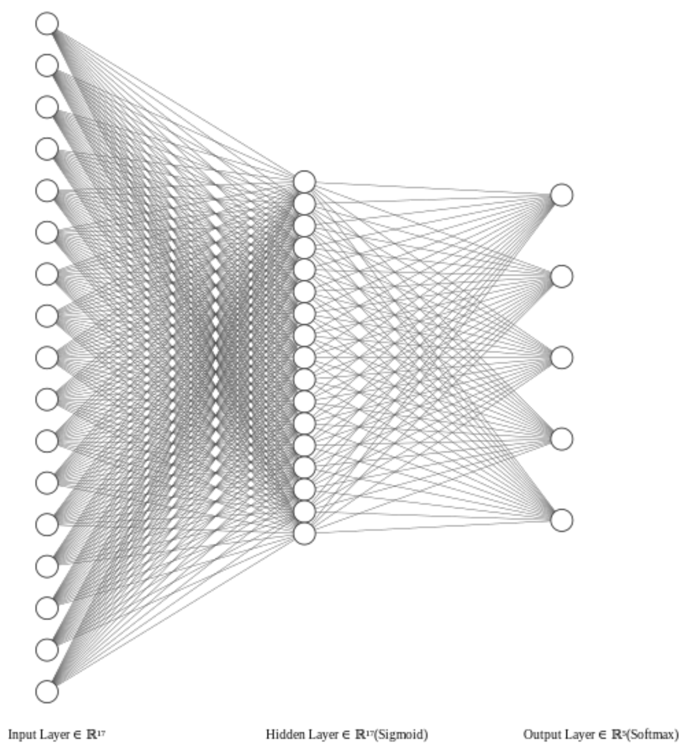


Fig. 2. Network Structure

The constructed neural network for solution consist of 3 layers which are input layer, hidden layer and output layer. Hidden layer consists of 17 neurons and output layer consists of 5 neurons. Sigmoid activation function is used for hidden layer. Since the problem is multi classification problem, Softmax activation function is used for output layer and Cross entropy loss function is used to train network.(Mean square loss function with sigmoid activation function on output layer was also used but accuracy and training results was very bad so We decide to use softmax and cross entropy)

To improve accuracy and training time, several optimization algorithms are used which are Momentum, Adaptive learning rate and Adam optimization as well as data normalization.

After trained network first time, the loss was very small but also accuracy was small so this means network has overfitting problem. To solve overfitting problem We was need to use generalized data so I was used half of dataset and this solved overfitting problem and finally accuracy was increased up to 98%

### A. *Optimization Algorithms*

**Momentum Optimization**

Momentum optimization behaves like an filter and helps accelerate gradients vectors in the right directions, thus leading to faster converging also It provides better results with noisy data. It is one of the most popular optimization algorithms and many state-of-the-art models are trained using it.

$$\Delta w_{ij}^k(t) = \mu \Delta w_{ij}^k(t-1) - \eta \frac{\partial J}{\partial w_{ij}^k} \qquad \text{where } 0 < \boldsymbol{\mu} < 1$$

Fig. 3. Momentum Optimization

**Adaptive Learning Rate**

Learning Rate Adaptation is another optimization method which changes learning rate according to loss function. If loss is increasing, learning rate will be decreased. If loss is decreased, learning rate will be increased.

$$\eta(t) = \begin{cases} \eta(t-1) + \gamma & J(t) < J(t-1) \\ \beta\eta(t-1) & J(t) > J(t-1) \\ 0 & \text{otherwise} \end{cases} \qquad \text{where } 0 < \boldsymbol{\beta}, \boldsymbol{\gamma} < 1$$

Fig. 4. Adaptive Learning Rate

**Adam optimization [6]**

Adaptive Moment Estimation (Adam) is another method that computes adaptive learning rates for each parameter. In addition to storing an exponentially decaying average of past squared gradients, Adam also keeps an exponentially decaying average of past gradients, similar to momentum. Algorithm computes the updating term as follows:

$$m_t = \beta_1 m_{t-1} + (1 - \beta_1) g_t$$
$$v_t = \beta_2 v_{t-1} + (1 - \beta_2) g_t^2$$

$$\hat{m}_t = \frac{m_t}{1 - \beta_1^t}$$
$$\hat{v}_t = \frac{v_t}{1 - \beta_2^t}$$

$$\theta_{t+1} = \theta_t - \frac{\eta}{\sqrt{\hat{v}_t} + \epsilon} \hat{m}_t.$$

Fig. 5. Adam optimization updating term

### B. Comparison Optimization Algorithms

After these optimization algorithms added to Matlab implementation, Network is trained 1000 epoch times and loss value is calculated for each epoch as shown in figure 6 and accuracy rate of each optimization as shown in table 3.

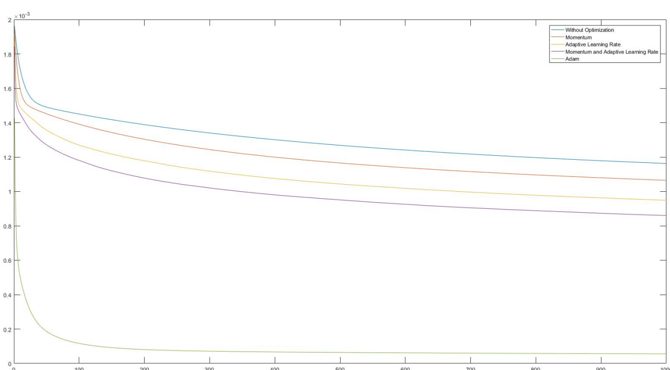


Fig. 6. Epoch and Loss values according to Optimizations

| Optimization Method | Accuracy |
|---|---|
| None | 57.20% |
| Momentum | 60.61% |
| Adaptive Learning Rate | 65.63% |
| Momentum and Adaptive Learning Rate | 68.37% |
| Adam | 98.07% |

Table 3. Accuracy rates of each optimization methods

| Output/Desired | Sitting^ | Sitting down^ | Standing^ | Standing up^ | Walking^ |
|---|---|---|---|---|---|
| Sitting | **99.85%** | 0.40% | 0% | 0.75% | 0.01% |
| Sitting down | 0.05% | **93.81%** | 0.18% | 2.70% | 0.41% |
| Standing | 0% | 1.88% | **98.92%** | 2.49% | 1.43% |
| Standing up | 0.09% | 2.63% | 0.40% | **92.31%** | 0.39% |
| Walking | 0% | 1.25% | 0.48% | 1.73% | **97.73%** |

Table 4. Confusion Matrix of Network with Adam optimization

As We can see from figure 6, table 3 and 4, Adam optimization gives us best result also between 400 and 500 epoch is enough to train network. This means that with Adam optimization, Network can be trained faster than other algorithms.

## V. Conclusion

In this work, We develop a Neural Network with several optimization methods and These methods are compared with each other. This Works also shows that we can solve this prediction human posture and movement problem with a Neural Network with Adam optimization.

For future works, We can change neuron count of hidden layer or add more hidden layer to gain more accuracy also We can use another dataset. Apart from these, we can obtain a system that is easier and more practical by using the data we collect from mobile phones, smart watches and by training our network with these data.

## References


[1] Ugulino, Wallace & Cardador, Débora & Vega, Katia & Velloso, Eduardo & Milidiú, Ruy & Fuks, H. (2012). Wearable computing: Accelerometers' data classification of body postures and movements. Advances in Artificial Intelligence-SBIA 2012. 52-61.

[2] Salzberg, S.L.: C4.5: Programs for Machine Learning by J. Ross Quinlan. Morgan Kauf-mann Publishers, Inc. (1993) Machine Learning 16(3), 235–240 (1994) ISSN: 0885-6125, doi: 10.1007/bf00993309

[3] Freund, Y., Schapire, R.E.: Experiments with a New Boosting Algorithm. In: International Conference on Machine Learning, pp. 148–156 (1996).

[4] S. Liu, R. X. Gao, D. John, J. W. Staudenmayer and P. S. Freedson, "Multisensor Data Fusion for Physical Activity Assessment," in IEEE Transactions on Biomedical Engineering, vol. 59, no. 3, pp. 687-696, March 2012.doi: 10.1109/TBME.2011.2178070

[5] F. Ioana-Iuliana and D. Rodica-Elena, "Detection of daily movements from data collected with two tri-axial accelerometers," 2011 34th International Conference on Telecommunications and Signal Processing (TSP), Budapest, 2011, pp. 376-380.

[6] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization.