

Fuzzy Logic Training for Predicting Age of Rats

Douglas E. Dow
Dept. of Electrical Engineering and Technology
Wentworth Institute of Technology
550 Huntington Avenue
Boston, Massachusetts
02115, USA
+1-617-989-4134
dowd@wit.edu

Isao Hayashi
Faculty of Informatics
Kansai University
2-1-1, Ryozenji-cho
Takatsuki, Osaka
569-1095, Japan
+81-72-690-2448
ihaya@cbii.kutc.kansai-u.ac.jp

ABSTRACT

Physiological measurements may contain data with nonlinear relations, non-normal distribution and large signal-to-noise ratio. Fuzzy logic has been utilized to analyze and classify physiological data. Age in mammals is reflected in physiological properties, such as skeletal muscle function. A fuzzy logic algorithm with self-tuning mechanism was developed in this study to make a model from physiological measurements, and predict age. The system was developed using scalar number sets having linear and nonlinear relations. Then the system was applied toward data of body mass and skeletal muscle function to predict the age of rats as a scalar value. The algorithm was developed using the python programming language. The results of the developed fuzzy logic system were compared with other machine learning algorithms using the Weka platform. The developed fuzzy logic model had a lower mean for relative absolute error (RAE) for the tested set of linear and nonlinear relations compared to the results of the tested machine learning algorithms in Weka. For prediction of rat age, the RAE of the fuzzy logic system was 22% compared with values of 23-33% for the other tested algorithms. Further testing and development of the fuzzy logic system on physiological data relations will be necessary to verify these promising results.

CCS Concepts

- **Theory of computation**→Theory and algorithms for application domains→Machine learning theory
- **Computing methodologies**→Knowledge representation and reasoning→Vagueness and fuzzy logic

Keywords

Machine learning; fuzzy logic; supervised training; tuning; python; aging; muscle; maximum force; muscle function; Weka

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

BICT 2017, March 15-16, Hoboken, United States

Copyright © 2017 ACM 978-1-63190-148-5

1. INTRODUCTION

Measurements of physiological and biomedical phenomenon typically require signal conditioning and statistical analysis. The measured data may have a non-normal distribution and large signal-to-noise ratio. The underlying physiological phenomenon may also have nonlinear relations. Fuzzy logic has been utilized to analyze and make predictions of physiological data [1, 8, 11].

As the distribution of age in the population of many societies shifts toward older ages [3], there is growing interest in developing potential therapeutic treatments to delay aging or prolong healthy function and quality of life [13, 14]. Age in mammals is reflected in certain physiological properties, such as skeletal muscle function [5, 14]. Methods to characterize the age from physiological measurements, such as muscle function, would be useful to determine the efficacy of potential therapies, treatments or lifestyle modifications.

A fuzzy logic algorithm with self-tuning mechanism [9, 11-13] was developed in this study to make a model from physiological measurements for characterization of age. The system was developed using scalar number sets having linear and nonlinear relations. Then the system was applied toward data of body mass and skeletal muscle function to predict the age of rats as a scalar value. Such a machine learning model that would characterize age based on physiological measurements would assist studies developing potential therapeutic treatments for maintaining healthy function during old age.

2. MATERIALS AND METHODS

2.1 Development of Algorithm

The fuzzy logic reasoning and self-tuning algorithm for supervised training that were utilized for this study were based on prior models [10, 12, 15, 16]. Key details and modifications are described here.

One vector of input numerical values \mathbf{x} has m values, which were expressed as x_0, x_1 through x_{m-1} . The output numerical value was expressed as y . Figure 1 shows the utilized membership function, which was an isosceles triangle centered on a_{ij} and having a width of b_{ij} . For an input value of x_j , the membership value of A_{ij} was calculated as follows.

$$A_{ij}(x_j) = 1 - \frac{2|x_j - a_{ij}|}{b_{ij}} \quad (1)$$

Where A_{ij} was the membership value for an x_j input value. The j indicated which x input (x_0, x_1, \dots, x_{m-1}), and the i indicated which

fuzzy rule (f_0, f_1, \dots, f_{q-1}). Any value of A_{ij} less than 0 was nullified (set to 0).

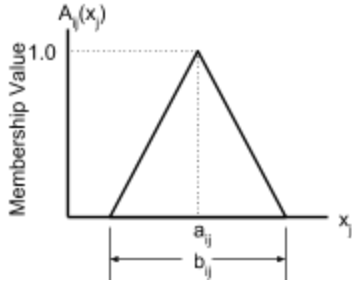


Figure 1. Fuzzy logic membership function. Function maps an input x_j value to the membership value that ranges from 0 to 1. Shape of the function was an isosceles triangle with height of 1.0, width of b_{ij} and centered at a_{ij} . Values of x_j that were outside of $a_{ij} \pm (b_{ij}/2)$ were mapped to 0.

Each of the x_j inputs had p membership functions to classify the x_j value. The number of fuzzy rules in the model was determined as follows.

$$q = p^m \quad (2)$$

where q was the number of fuzzy rules. The output of applying all q fuzzy rules was determined as follows.

$$\mu_i = \prod_{j=0}^{m-1} A_{ij}(x_j) \quad (3)$$

$$y = \frac{\sum_{i=0}^{q-1} \mu_i w_i}{\sum_{i=0}^{q-1} \mu_i} \quad (4)$$

The method of training the membership functions for supervised learning was denoted as self-tuning [1-4], and was based on maximum descent [4]. A training set of n \mathbf{x} vectors was expressed as \mathbf{x}_k , which would include vectors $\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_{n-1}$. The output of applying \mathbf{x}_k to all the fuzzy rules was y_k . The error value for an input \mathbf{x}_k vector applied to one rule to determine y_k was calculated as follows.

$$E_k = \frac{1}{2} \cdot (y_k - y_k^r) \quad (5)$$

where y_k^r was the desirable or correct output value, used for supervised training.

The gradients in the error for each tuning parameter were calculated based on an input \mathbf{x}_k vector, as follows.

$$\frac{\delta E_k}{\delta w_i} = \frac{\mu_{ki}}{n-1} \cdot (y_k - y_k^r) \quad (6)$$

$$\frac{\delta E_k}{\delta a_{ij}} = \frac{\mu_{ki}}{n-1} \cdot (y_k - y_k^r) \cdot (w_i - y_k) \cdot \text{sgn}(x_{kj} - a_{ij}) \cdot \frac{2}{b_{ij} \cdot A_{ij}(x_{kj})} \quad (7)$$

$$\frac{\delta E_k}{\delta b_{ij}} = \frac{\mu_{ki}}{n-1} \cdot (y_k - y_k^r) \cdot (w_i - y_k) \cdot \frac{1 - A_{ij}(x_{kj})}{b_{ij} \cdot A_{ij}(x_{kj})} \quad (8)$$

where $\text{sgn}(g)$ was +1 if g was greater or equal to 0, and -1 if g was less than 0.

During training, one \mathbf{x}_k vector was processed during each timing step. The tuning parameters were updated after processing each input \mathbf{x}_k vector from one time step (t) to the next ($t+1$) as follows.

$$w_i(t+1) = w_i(t) - K_w \cdot \frac{\delta E_k}{\delta w_i} \quad (9)$$

$$a_{ij}(t+1) = a_{ij}(t) - K_a \cdot \frac{\delta E_k}{\delta a_{ij}} \quad (10)$$

$$b_{ij}(t+1) = b_{ij}(t) - K_b \cdot \frac{\delta E_k}{\delta b_{ij}} \quad (11)$$

Each \mathbf{x}_k vector in the training set was processed by the fuzzy rules. After processing each \mathbf{x}_k vector, the fuzzy membership parameters (w_i, a_{ij}, b_{ij}) were adjusted according to equations 8-10. The sequence of these tuning steps is further described below.

An *iteration* was one pass through all n of the \mathbf{x}_k vectors in the training set. A total error, S , for one iteration was calculated by summing up all of the absolute values of the E_k values as follows.

$$S = \sum_{k=0}^{n-1} |2 \cdot E_k| \quad (12)$$

The improvement in S from one iteration to the next iteration is I , where I is the difference of the value of S from the prior iteration and the value of S from this iteration.

$$I(t) = S(t-1) - S(t) \quad (13)$$

The iterations of training stop when the value of the improvement, I , is less than a threshold value T .

The term *wave* was used to denote a series of iterations until the level of improvement, I , did not meet the threshold, T . A wave had two parts. During the first part, the fuzzy parameter of w_i values were tuned according to eq. 6 and 9. During this first half-wave, iterations of tuning occurred until the level of improvement, I , was less than the threshold, T . Then, during the second half-wave, the fuzzy parameters of a_{ij} and b_{ij} were tuned according to eq. 7-8 and 10-11. Thus, constituting a full wave, the first half-wave tuned w_i , and the second half-wave tuned a_{ij} and b_{ij} .

Following completion of a whole wave, another wave was allowed to commence. This process was repeated until either the target number of waves was completed or a whole wave did not result in an improvement, I , above the threshold level, T .

The Fuzzy Logic model consisted of the \mathbf{a} matrix containing the membership center points, the \mathbf{b} matrix containing the membership widths, and the \mathbf{w} array containing the weights for each rule. During training, the value of S was monitored. Models at the end of an iteration with an S value that did not show sufficient improvement, were discarded. The final trained model was the one with the lowest S value that did show sufficient improvement. For this model, the \mathbf{a} and \mathbf{b} matrix, and \mathbf{w} array were saved to be used during testing.

The initial position of the membership functions was determined by the range of values of each \mathbf{x} input variable to be trained. The membership functions were distributed evenly across the range with about one quarter overlap on each side and also beyond the maximum and minimum values. The initial value of the \mathbf{w} array was the inverse of the number of \mathbf{x} vectors.

2.2 Generation of Scalar Data Sets for Training and Testing

Sets of input \mathbf{x} vectors were generated. Five equations were chosen to relate each \mathbf{x} vector with an output y value.

Ten sets of data values were generated by a custom python program to use as the input \mathbf{x} -vectors, labeled as set s_0 through s_9 . Each set consisted of 100 \mathbf{x} vectors. Each \mathbf{x} vector had three values, x_0 , x_1 and x_2 . The values were randomly generated floating point numbers, but within specific ranges as follows. Values for x_0 were randomly distributed between -10 and 10, values for x_1 were randomly distributed between 1000 and 5000, and values for x_2 were randomly distributed between 0 and 150. The purpose of having heterogeneous ranges was to mimic certain properties of physiological data sets, such as differences in the symmetry of the data about 0, size of the offsets, and scale of the range.

Of the five equations that formed a relation between the \mathbf{x} vectors and y outputs, two were linear and three were nonlinear. The output y values were generated by the equations shown in Table 1. For each set of input \mathbf{x} vectors (s_0 to s_9), five different correct output y^f arrays were calculated based on the five equations shown in Table 1. Having a varied assortment of relations was intended to challenge the developed system in ways that may mimic some of the properties of physiological applications. In many applications, part of the relation is linear but other parts are nonlinear.

Table 1. Equations that define the five relations used

Name	Type	Equation
Eq1	linear	$y = -12 \cdot x_0 + 0.017 \cdot x_1 + 8 \cdot x_2 + 350$
Eq2	linear	$y = 25 \cdot x_0 + 0.04 \cdot x_1 - 6 \cdot x_2 - 95$
Eq3	nonlinear	$y = \sin\left(\frac{x_0}{3}\right) \cdot \left(2^{\left(\frac{x_1}{1000}\right)} + x_2\right)$
Eq4	nonlinear	$y = (x_0 + \text{sqrt}(x_1))^{\left(\frac{x_2}{50}\right)}$
Eq5	nonlinear	$y = x_0^3 + (0.004 \cdot x_1)^2 - (3 \cdot x_2)$

2.3 Experimental Trials with Scalar Sets

The parameters of the fuzzy logic system and tuning algorithm were varied over ranges and tested in order to verify the learning function, gain insight into the how the parameter values affected the model for different relations, and to progress toward determining optimal parameters that work well for all of the tested relations. The parameters that were tested over a range included the number of membership functions, tuning constants K_w , K_a and K_b , and parameters that control the flow of iterations and when to stop the learning process.

2.3.1 Training

For any one set of values for the parameters of the fuzzy membership and tuning algorithm, supervised training was done independently for the five relations shown in Table 1, resulting in five independent fuzzy models. For each of the five relations (Table 1), the training data consisted of 1) \mathbf{x} vector set s_0 , and 2) the corresponding y^f array that was calculated by the equation of

Table 1. The result of the training would be five independent fuzzy models, each with an \mathbf{a} matrix containing the membership center points, the \mathbf{b} matrix containing the membership widths, and the \mathbf{w} array containing the weights for each rule.

2.3.2 Testing

The five tuned fuzzy models that were determined by the supervised training were then independently tested on nine sets of testing data that consisted of 1) one of the nine \mathbf{x} vector set $s_1 - s_9$, and 2) the corresponding y^f array that was calculated by the utilized \mathbf{x} vector set and the equation of Table 1.

For each set of \mathbf{x} and y , the results of testing were calculated as average error (AE), average correct (AC), and relative absolute error (RAE) in the following way.

$$AE = \frac{\sum_{k=0}^{n-1} |y_k - y_k^a|}{\sum_{k=0}^{n-1} |y_k^a|} \quad (14)$$

$$AC = 1.0 - AE \quad (15)$$

$$RAE = \frac{\sum_{k=0}^{n-1} |y_k - y_k^a|}{\sum_{k=0}^{n-1} |y_k - y^a|} \quad (16)$$

where y^f was the desirable or correct output value, and y^a was the average of the y_k values. These calculations were repeated for the nine testing data sets, the mean value was found, and the results were reported as a percentage in the form of mean \pm standard deviation. The mean AC value was termed percent correct.

2.3.3 Comparison to other machine learning

The results of the developed fuzzy logic algorithm were compared to results of other machine learning algorithms. Weka is an open source, publicly available machine learning workbench (University of Waterloo, <http://www.cs.waikato.ac.nz/ml/weka/>) [7, 9]. Several of the available machine learning algorithms were selected in Weka, and listed in Table 2. These algorithms were used with their default parameters on the same training (s_0) and testing ($s_1 - s_9$) data sets for each of the relations in Table 1. The resulting RAE was recorded. The mean RAE value for the testing data sets was found, and reported as a percentage in the form of mean \pm standard deviation.

Table 2. Machine learning algorithms that were utilized from the Weka open source library.

Type	Machine Learning Algorithm
function	Linear Regression
function	Multilayer Perceptron
lazy	KStar
rule	Decision Table
tree	Random Forest

2.4 Physiological Data and Analysis

To develop a model for the prediction of age, the developed fuzzy logic system was applied to physiological measurements of the body mass and skeletal muscle function of rats whose age varied from 8 months (fully developed adults) to 33 months (old age). Body mass and muscle function change with age. A total of 54 rats were utilized in the study: 14 rats at age of 8 months old, 10 rats at 18 months, 10 rats at 23 months, 10 rats at 28 months, and 10 rats at 33 months old.

Rats were obtained and muscle properties analyzed as has previously been reported [2, 5, 6]. Fifty-four Male Fisher Brown Norway rats were utilized. All procedures were conducted in accordance with the guidelines established in the United States Public Health Service Guide for the Care of Laboratory Rats (NIH Publication 85-23) and with the approval of the University Committee on the Use and Care of Rats. For operative procedures, rats were anesthetized with an initial intraperitoneal injection of a mixture of ketamine (100 mg/kg) and xylazine (10 mg/kg). The contractile properties of the extensor digitorum longus (EDL) muscles were measured *in vitro*. With rats anesthetized deeply, each EDL muscle was removed from the hindlimb and immersed in Ringer's solution in a tissue bath. A muscle was secured between a fixed post and a force transducer. Muscle contractions were generated by electrical stimulation. The voltage and muscle length were each adjusted to produce a maximum isometric twitch contraction. For this maximum twitch contraction, the half-relaxation time (HRT) were measured. Maximum force was achieved by increasing the frequency of stimulation during successive contractions until a maximum isometric tetanic contraction was reached.

The purpose of the fuzzy logic system was to predict the age of the rats in months (y value). The input (x vectors) values consisted for each rat of the body mass (g), the mass of the EDL muscle (mg), the HRT (ms) value, and the maximum force (mN).

Data for the 54 rats was divided into 2 groups, such that a training set contained measurements for 32 rats (~60%), and a testing set contained measurements for the remaining 22 rats (~40%). The fuzzy logic system used the training set to form the fuzzy model using supervised learning. Then the testing set was used to determine the error for age. The age was treated as a scalar number, not grouped into a classification age group. The RAE values were used as primary outcome.

Comparison of how other machine learning algorithms were able to predict rat age based on the same training and testing set were made using the Weka library. The same Weka algorithms were used as previously for the scalar data sets in this study (Table 2).

3. RESULTS

3.1 Scalar Data Sets

The developed fuzzy logic and tuning system was applied to scalar data sets having linear or nonlinear relations as described in Table 1. To explore function of the system, various ranges of the parameters were utilized for training (on scalar set s_0), and then tested (on scalar sets $s_1 - s_9$). Figure 2 shows the resulting percent correct for different numbers of fuzzy membership functions. About 4 or 5 membership functions appears to result in the best percent correct for most functions.

The tuning variable of K_w was tested for a range of values and plotted in Figure 3. Values for K_w of 0.1 to 0.05 seemed to work the well for most of the relations.

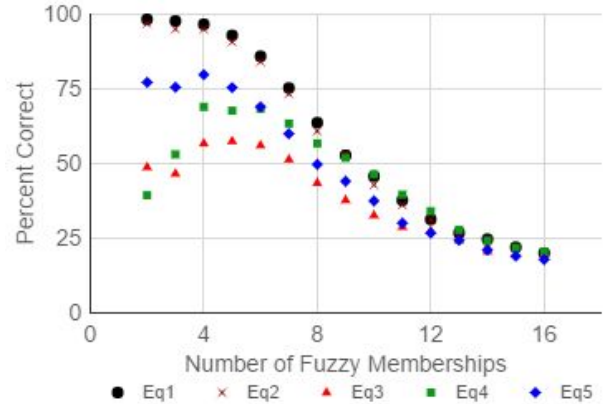


Figure 2. Number of Fuzzy Membership Function and resulting Percent Correct. The type of equations (Eq1 - Eq5) are listed in Table 1. For these trials, the threshold was 0.05 with a 5 count to achieve that improvement, and the maximum number of iterations per wave was 500. The tuning variable K_w was 0.005 and both K_a and K_b were 0.002.

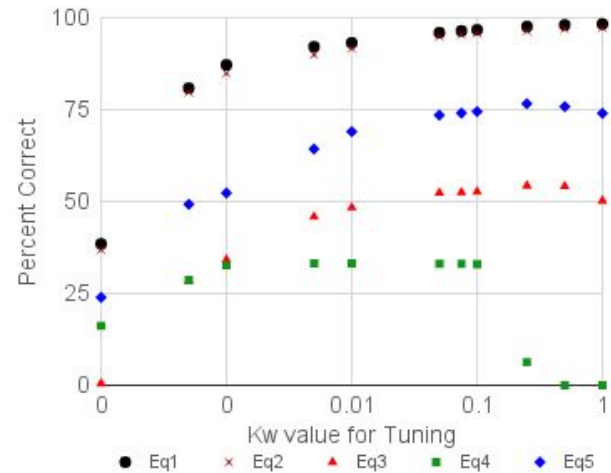


Figure 3. Tuning variable K_w and resulting Percent Correct. The type of equations (Eq1 - Eq5) are listed in Table 1. For these trials, the number of membership functions was 4, the threshold was 0.01 with a 1 count to achieve that improvement, and the maximum number of iterations per wave was 10k. Both of the tuning variable K_a and K_b were 0.005.

The threshold in how much is required for Percent Error to improve for each training pass through all of the x vectors is plotted in Figure 3 with the resulting Percent Correct values. A value for threshold of 0.1 or smaller seemed to work the well for most of the relations.

To verify function and compare with other machine learning algorithms, the same data sets (s_0 for training, $s_1 - s_9$ for testing) were used with selected machine learning algorithms (Table 2) using the open source Weka library. Results are shown in Figure 4. Some of the algorithms worked comparatively better for some of the relations, but worse for the other relations. For example, both linear regression and multilayer perceptron performed better for the two linear data relations compared with the fuzzy logic model, but worse for two of the nonlinear relations.

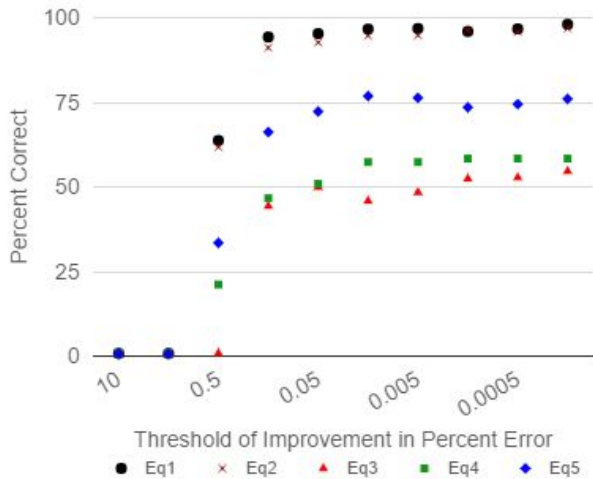


Figure 3. The threshold for how much the Percent Error needs to improve for each pass through all of the input x vectors during training and tuning is plotted with the resulting Percent Correct. The type of equations (Eq1 - Eq5) are listed in Table 1. For these trials, the number of membership functions was 4, and the maximum number of iterations per wave was 10k. The tuning variable of K_w was 0.005, and both of the tuning variable K_a and K_b were 0.002.

Toward evaluating which algorithm performed better as a whole for all of the five relations, the mean of the five RAE values for the relations (Table 1) was found and plotted in Figure 5. The fuzzy logic system had the lowest mean value of 22% compared with values of 23-33% for the other tested algorithms. This supports that the developed fuzzy logic system appears to work as good or better for the 5 tested relations.

3.2 Physiological Data Sets

Following developing and testing of the fuzzy logic system on the scalar data sets, the algorithm was applied to a training and test data set of physiological data for which muscle mass and skeletal muscle properties were used to predict the age of the rats.

Figure 6 shows the changes with aging of the four properties analyzed. Rats become young adult at 2 or 3 months old. By the age of the first time point at 8 month old, the rats are fully mature adults. After about 2 years old the rats progressively undergo the changes of old age, so the last three time points of 23, 28 and 33 months show changes with old age.

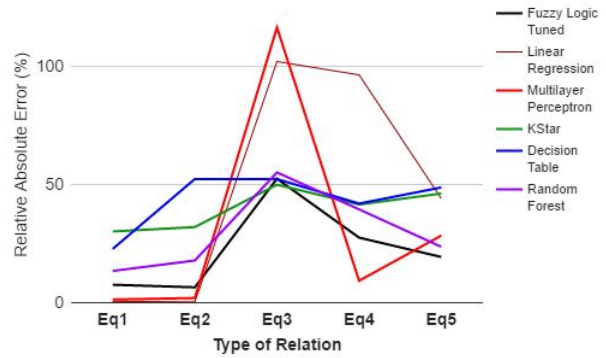


Figure 4. Comparison of the developed Fuzzy Logic System to other Machine Learning algorithms for the same data sets. The other machine learning algorithms were run on the Weka library and described in Table 2. The parameters of the fuzzy logic system for these trials included the threshold was 0.05 with a 5 count to achieve that improvement, the number of membership functions was 4, and the maximum number of iterations per wave was 100. The tuning variable of K_w was 0.1, and both of the tuning variable K_a and K_b were 0.0005.

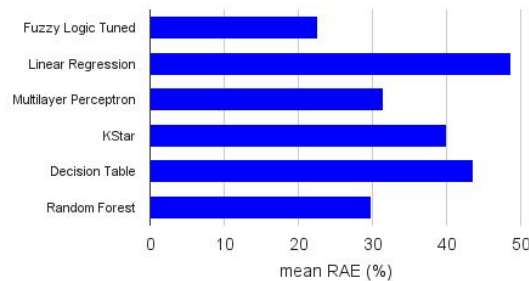


Figure 5. For each type of machine learning algorithm, the mean of RAE values for the 5 relations is plotted.

The predicted age for training and testing of the algorithm was a scalar value in months. The results were compared to several other machine learning algorithms (Table 2) using the Weka library. The results are shown in Figure 7.

4. Discussion

A fuzzy logic and tuning system was developed and tested on both a set of five scalar relations (2 linear and 3 nonlinear), and on physiological data of the age of a rat as reflected in the values for body mass and properties of the skeletal muscles. The performance of the system was affected by the parameters of the model and tuning, such as the number of fuzzy logic membership functions, tuning variable K_w , and threshold level necessary to continue the training process.

Just 2 membership functions were sufficient for the linear relations (Eq1 and Eq2) to result after training in the highest levels of performance according to percent correct during testing. In contrast, some of the nonlinear relations had relatively low performance with 2 membership functions, but did better with 4-6 membership functions. The physiological data for age and muscle

properties was best modeled with 3 membership functions. More than 6 membership functions resulted in lower performance for all the tested scalar relations. This decline with higher numbers may be due to the effect of overfitting, where the trained model matches the exact points of the training data, but does not model the overall relationship. The optimal number probably reflects the primary frequency components of the input x parameters.

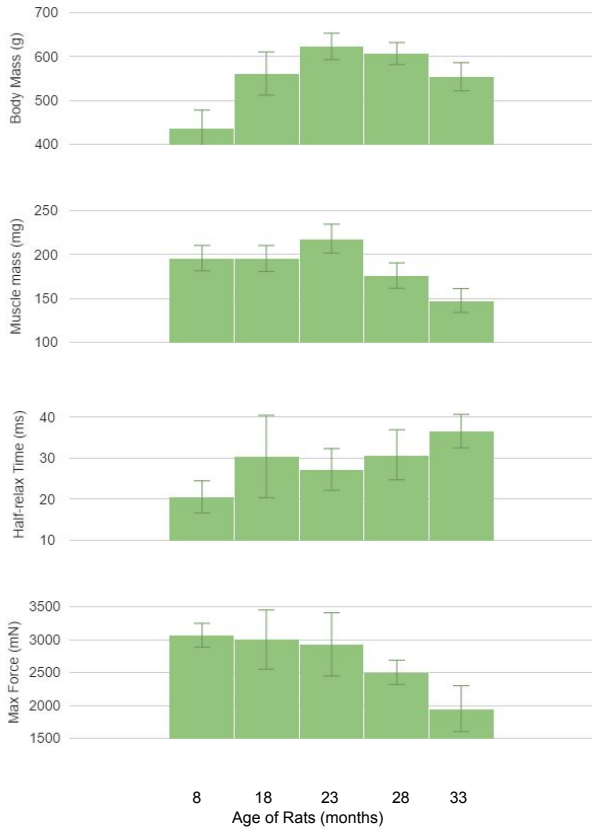


Figure 6. Changes in body mass and muscle properties of rats with age. Body mass was the weight of the rat on the day of the evaluation. Muscle mass was for the EDL muscle after excision. Half-relaxation time was the time required for the generated muscle force after a twitch contraction to decline 50% from peak force. Maximum Force was the highest force generated during a tetanic contraction. The error bars show standard deviation. Some of the properties increase with old age, others decrease and yet others seem to increase and then decrease with old age.

The order of tuning for training the fuzzy model optimized the membership rule weights (w) with K_w first, and then optimized the membership location (a) and width (b) values with K_a and K_b . All of the values affected performance, but K_w had a more primary role than K_a and K_b in the tested relations. K_w values of 0.05 and smaller seemed to result in the best performance for the scalar set with the linear and nonlinear relations. Similarly, the threshold for improvement seemed to result in the best performance for values of 0.01 or smaller. The developed fuzzy logic and tuning system was compared with several machine learning algorithms using the open source Weka library. Though some of the tested algorithms resulted in a lower error in

prediction as indicated by the RAE values for some of the relations compared to the developed fuzzy logic system, the fuzzy logic system did seem to perform as well or better overall, when the mean of RAE for all of the relationships were compared.

The developed system was then applied for training of physiological measurements of body mass and muscle properties for prediction of the age of the rats. The values of the fuzzy logic and tuning system developed for the scalar values and five linear and nonlinear relations resulted in predictions of age, but the system made better prediction after further adjustment of the parameters. This observation may support the notion that one machine learning system is not ideal for all types of data sets and relations. For the rat and muscle physiological data and prediction of age, a lower number of membership functions (3), and higher values for tuning constants (K_w of 1.0; K_a and K_b of 0.5), and threshold for improvement (0.25 with a count of 5 iterations to achieve that) resulted in better prediction of the age.

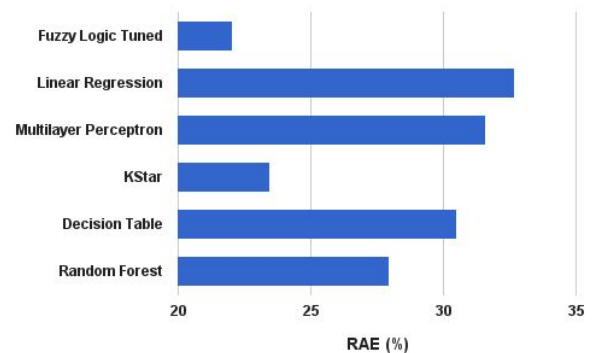


Figure 7. Comparison of the developed Fuzzy Logic System to other Machine Learning algorithms for prediction of the age of rats in months using body mass and properties of skeletal muscles. The other machine learning algorithms were run on the Weka library and described in Table 2. The parameters of the fuzzy logic system for these trials included the threshold was 0.25 with a 5 count to achieve that improvement, the number of membership functions was 3, the maximum number of iterations per wave was 10k, and the maximum number of full waves of tuning was 1. The tuning variable of K_w was 1, and both of the tuning variable K_a and K_b were 0.5. The Relative Absolute Error (RAE) for the fuzzy logic model was lower than the values for the other machine learning algorithms.

Consideration of potential weakness in the developed fuzzy logic system might relate to cases where the test data set had values of x that were far outside of the range of the values in the training set. The isosceles triangle membership functions may not be the most robust for the boundary conditions. Improvements to the algorithm might employ another type of membership function for the boundaries of the ranges or as offset constant.

The developed fuzzy logic and tuning system does appear to have promise for learning a relation of parameters with an outcome value, such as for physiological data. Further testing and development of this algorithm will show how this system could be suitable for application to other physiological relationships and data.

5. REFERENCES

- [1] BHATLA, N. AND JYOTI, K. 2012. A Novel Approach for heart disease diagnosis using Data Mining and Fuzzy logic. *International Journal of Computer Applications*, 54.
- [2] CARLSON, B.M., BORISOV, A.B., DEDKOV, E.I., DOW, D. AND KOSTROMINOVA, T.Y. 2002. The biology and restorative capacity of long-term denervated skeletal muscle. *Basic Appl Myol*, 12, 247-254.
- [3] CHRISTENSEN, K., DOBLHAMMER, G., RAU, R. AND VAUPEL, J.W. 2009. Ageing populations: the challenges ahead. *The Lancet*, 374, 1196-1208.
<http://www.sciencedirect.com/science/article/pii/S0140673609614604>.
- [4] CURRY, H.B. 1944. The Method of Steepest Descent for Non-linear Minimization Problems. *Quarterly of Applied Mathematics*, 2, 258-261.
<http://www.jstor.org/stable/43633461>.
- [5] DOW, D.E., DENNIS, R.G. AND FAULKNER, J.A. 2005. Electrical Stimulation Attenuates Denervation and Age-Related Atrophy in Extensor Digitorum Longus Muscles of Old Rats. *The Journals of Gerontology Series A: Biological Sciences and Medical Sciences*, 60, 416-424.
<http://biomedgerontology.oxfordjournals.org/content/60/4/416>.
- [6] DOW, D.E., CARLSON, B.M., HASSETT, C.A., DENNIS, R.G. AND FAULKNER, J.A. 2006. Electrical stimulation of denervated muscles of rats maintains mass and force, but not recovery following grafting. *Restorative Neurology and Neuroscience*, 24, 41-54.
<http://content.iospress.com/articles/restorative-neurology-and-neuroscience/rnn00329>.
- [7] FRANK, E., HALL, M., TRIGG, L., HOLMES, G. AND WITTEN, I.H. 2004. Data mining in bioinformatics using Weka. *Bioinformatics*, 20, 2479-2481.
<http://bioinformatics.oxfordjournals.org/content/20/15/2479>.
- [8] GRANT, P. AND NAESH, O. 2005. Fuzzy logic and decision-making in anaesthetics. *Journal of the Royal Society of Medicine*, 98, 7-9.
- [9] HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P. AND WITTEN, I.H. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.* 11, 10-18.
<http://doi.acm.org/10.1145/1656274.1656278>.
- [9] HALL, M., FRANK, E., HOLMES, G., PFAHRINGER, B., REUTEMANN, P., WITTEN, I.H. 2009. The WEKA Data Mining Software: An Update. *SIGKDD Explor. Newsl.*, 11, 10-18.
- [10] HAYASHI, I., NAITO, E. AND WAKAMI, N. 1993. Proposal for fuzzy connectives with a learning function using the steepest descent method. *Japanese Journal of Fuzzy Theory and Systems*, 5.
- [11] HO, K., TSUCHIYA, N., NAKAJIMA, H., KURAMOTO, K., KOBASHI, S. AND HATA, Y. August 2009. Fuzzy logic approach to respiration detection by air pressure sensor. *IEEE International Conference on Fuzzy Systems*, 911-915.
- [12] ICHIHASHI, H. 1991. Iterative fuzzy modeling and a hierarchical network. In *Proceedings of the Fourth IFSA Congress, Engineering, Brussels*, 52.
- [13] KVORNING, T., CHRISTENSEN, L.L., MADSEN, K., NIELSEN, J.L., GEJL, K.D., BRIXEN, K. AND ANDERSEN, M. 2013. Mechanical Muscle Function and Lean Body Mass During Supervised Strength Training and Testosterone Therapy in Aging Men with Low-Normal Testosterone Levels. *Journal of the American Geriatrics Society* 61, 957-962.
<http://onlinelibrary.wiley.com/doi/10.1111/jgs.12279/abstract>.
- [14] LANG, T., STREEPER, T., CAWTHON, P., BALDWIN, K., TAAFFE, D.R. AND HARRIS, T.B. 2010. Sarcopenia: etiology, clinical consequences, intervention, and assessment. *Osteoporosis International*, 21, 543-559.
<http://link.springer.com/article/10.1007/s00198-009-1059-y>.
- [15] NOMURA, H., HAYASHI, I. AND WAKAMI, N. March 1992. A learning method of fuzzy inference rules by descent method. In *IEEE International Conference on Fuzzy Systems*, 1992. IEEE, 203-210.
- [16] NOMURA, H., HAYASHI, I. AND WAKAMI, N. 1991. A self-tuning method of fuzzy control by descent method. In *Proc. IFSA*, 155-158.