AIMS Mathematics

AIMS Mathematics, 4(4): 1078–1086.
DOI: 10.3934/math.2019.4.1078
Received: 11 July 2019
Accepted: 31 July 2019
Published: 09 August 2019

http://www.aimspress.com/journal/Math

Research article

Fuzzy gradient descent for the linear fuzzy real number system

Frank Rogers*

Department of Mathematics, University of West Alabama, Livingston, Alabama 35470, USA

* **Correspondence:** Email: frogers@uwa.edu.

Abstract: Many problems in education, finance, and engineering design require that decisions be made under uncertainty. In these fields, Machine Learning is often used to search for patterns and information from data. To find patterns in Fuzzy Data, Fuzzy Machine Learning techniques can be used. In this paper, we focus on solving and manipulating Fuzzy Nonlinear problems in the Linear Fuzzy Real (LFR) number system using the Gradient Descent. The Gradient Descent is the most often used learning algorithm in Machine Learning. Thus, we propose the LFR Gradient Descent method for solving nonlinear equations in the LFR number system.

Keywords: fuzzy nonlinear optimization; fuzzy numbers; linear fuzzy real numbers; machine learning; gradient descent
Mathematics Subject Classification: 00A69

1. Introduction

Problems in production planning and scheduling, location, transportation, finance, and engineering design sometimes require that decisions be made under uncertainty. In education, there exist Fuzzy Educational datasets, which are useful sources of information for educational software developers, students, parents, researchers and other educational stakeholders [2]. Fuzzy decision making was initially introduced by Bellman and Zadeh [1]. This concept was then adopted to mathematical programming [14], uncertain programming [12,14], fuzzy posets [6], linear programming [3,7,8,10,13,14] and many other concepts.

Linear Fuzzy Real (LFR) numbers are a system of numbers that have properties comparable to the set of real numbers and the set of fuzzy numbers [9]. LFR numbers are used in the study of fuzzy random variables [4,9] and in the linear optimization problem [8]. The set of LFR numbers is a set that shows intermediate properties which are unique to the set and not to those of either the real numbers or the “general” fuzzy numbers. Because of the unique properties of LFR numbers, we can solve fuzzy linear and nonlinear problems [11].

At the core of machine learning techniques, there are many types of cost functions. The process of increasing the accuracy of our model by iterating over a training data involves searching for the lowest error in the cost function. The cost function is used to check the error in predictions of a Machine Learning model. In many learning algorithms the Gradient Descent is used to minimize the cost function [5].

In this paper, the Gradient Descent method is used to find both the extreme values and find the solutions of fuzzy equations.

The Crisp Gradient Descent algorithm begins with an initial value x^0 and generates the sequence $\{x^n\}_{n=0}^{\infty}$, by

$$x^n = x^{(n-1)} - \gamma \nabla F.$$

The algorithm finds the minimum of a function f by taking small steps proportion to the negative of the gradient of the function f. The symbol γ is the measure of the small steps and ∇F is the gradient of f.

Solving fuzzy equations also implies that the Gradient Descent method can be used to minimize fuzzy cost functions since the values of x that satisfies

$$f(x) = b \Longrightarrow f(x) - b = 0.$$

Will also be the global minimizer of

$$\|f(x) - b\|_2^2.$$

The paper is outlined as follows: Linear Fuzzy Real numbers are reviewed in Section 2. In Section 3, we look at nonlinear equations in LFR. In Section 4, we illustrate LFR Gradient Descent method to solve systems of nonlinear equations in LFR. Finally, conclusions are made in Section 5.

2. Linear fuzzy real numbers

In this section, we describe the Linear Fuzzy Real numbers [7,10]. Considering the set of all real numbers R, one way to associate a fuzzy number with a fuzzy subset of real numbers is as a function $\mu: R \to [0,1]$, where the value $\mu(x)$ is to represent a degree of belonging to the subset of R.

Definition 2.1. *(Linear fuzzy real number) Let $\mu: R \to [0,1]$ be a function such that:*

1. $\mu(x) = 1$ *if* $x = b$;
2. $\mu(x) = 0$ *if* $x \leq a$ *or* $x \geq c$;
3. $\mu(x) = (x - a)/(b - a)$ *if* $a < x < b$;
4. $\mu(x) = (c - x)/(c - b)$ *if* $b < x < c$.

Then $\mu(a,b,c)$ is called a linear fuzzy real number with associated triple of real numbers (a,b,c) where $a \leq b \leq c$ shown in Figure 1.

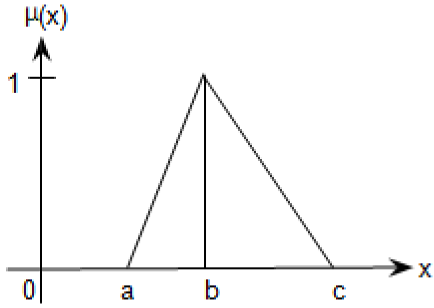

Figure 1. Linear fuzzy real number $\mu(a,b,c)$.

We let *LFR* be the set of all linear fuzzy real numbers. We note that any real number b can be written as a linear fuzzy real number, $r(b)$, where $r(b) = \mu(b,b,b)$ and so $R \subseteq LFR$. As a linear fuzzy real number, we consider $r(b)$ to represent the real number b itself. A linear fuzzy real number $\mu(a,b,c)$ is defined to be positive if $a > 0$, negative if $c < 0$, and zeroic if $a \leq 0$ and $c \geq 0$. The hybrid nature of LFR allows one to map the set to an isomorphic set where we can use inverse operations to solve fuzzy linear equations [10].

Operations, functions, and linear equations are also defined in *LFR* as follows:

Definition 2.2. *(Operations in LFR) For given two linear fuzzy real numbers* $\mu_1 = \mu(a_1,b_1,c_1)$ *and* $\mu_2 = \mu(a_2,b_2,c_2)$, *we define addition, subtraction, multiplication, and division by*

1. $\mu_1 + \mu_2 = \mu(a_1 + a_2, b_1 + b_2, c_1 + c_2)$;
2. $\mu_1 - \mu_2 = \mu(a_1 - c_2, b_1 - b_2, c_1 - a_2)$;
3. $\mu_1 \cdot \mu_2 = \mu(\min\{a_1a_2,a_1c_2,a_2c_1,c_1c_2\},b_1b_2,\max\{a_1a_2,a_1c_2,a_2c_1,c_1c_2\})$.;
4. $\frac{\mu_1}{\mu_2} = \mu_1 * \frac{1}{\mu_2}$; where $\frac{1}{\mu_2} = \mu(\min\left\{\frac{1}{a_2},\frac{1}{b_2},\frac{1}{c_2}\right\}, \text{median}\left\{\frac{1}{a_2},\frac{1}{b_2},\frac{1}{c_2}\right\}, \max\left\{\frac{1}{a_2},\frac{1}{b_2},\frac{1}{c_2}\right\})$.

Definition 2.3. *(Function in LFR) Given real-valued function* f: $R \rightarrow R$ *and an LFR* $\mu(a,b,c)$, *the LFR-valued function* $\ddot{f}$: *LFR* $\rightarrow$ *LFR is defined as*

$$\ddot{f}\,(\mu(a,b,c)) = \mu(a^*,b^*,c^*),$$

where $a^* = \min\{f(a),f(b),f(c)\}$, $b^* = \text{median}\{f(a),f(b),f(c)\}$, $c^* = \max\{f(a),f(b),f(c)\}$. *We note that if* $a = b = c$ *then* $a^* = b^* = c^*$, *i.e.,* $\bar{f}(r(b)) = r(f(b))$. *Hence* $\ddot{f}$ *is an extension of the function f.*

Definition 2.4. *(Linear equation in LFR) A linear equation over LFR is an equation of the form*

$$\mu_1 \cdot \mu_x + \mu_2 = \mu_3,$$

where the μ_i *are LFR's for* $i = 1,2,3$ *and* $\mu_x = \mu(\alpha,\beta,\gamma)$ *is an unknown LFR with a triple of unknown real numbers* (α,β,γ).

Definition 2.5. *(Ordering Properties of LFR) Given* $\mu_1, \mu_2 \in LFR$, $\mu_1 \leq \mu_2$ *provided that*

$$a_1 \leq a_2, b_1 \leq b_2, c_1 \leq c_2.$$

If $0 \leq \mu(a,b,c)$, *then* $0 \leq a \leq b \leq c$, *hence* μ *is a non-negative linear fuzzy real number. Therefore, if* μ *is non-negative and zeroic, then* $a = 0$ *precisely.*

(LFR,≤) is a complete ordered set. However, it is not linearly ordered. If we let $\mu_1 = \mu(3,4,5)$ and let $\mu_2 = \mu(a,5,6)$ and state that $a < 3$, then it is not true that $\mu_1 \leq \mu_2$ nor is it true that $\mu_1 \geq \mu_2$. Therefore, μ_1 and μ_2 are incomparable in this order [10].

3. Nonlinear equations in LFR

Nonlinear equations can be found in many applications, all the way from light diffraction to planetary orbits for example [10]. In this section, we discuss how to solve nonlinear equations in *LFR* so that we may locate the maximum or minimum of nonlinear functions. It is important that we discuss locating fuzzy zeros of nonlinear functions. Thus, we will find a μ_x in *LFR* such that a nonlinear equation $f(\mu_x) = 0$, where f: $LFR \rightarrow LFR$ is a nonlinear function.

3.1. Support for LFR gradient descent for solving nonlinear equations

Solving and optimizing nonlinear equations over linear fuzzy real numbers is possible with a modification of the Gradient Descent method. This method is also known as the Steepest Descent method over real numbers.

Definition 3.1. A function f satisfies Lipschitz if there is a real number N such that $|f(x)-f(y)| \leq N|x-y|$.

We can apply Definition 3.1 to the function $g = \nabla f$, such that there is a real number P such that $|g(x)-g(y)| \leq P|x-y|$. Thus, the function g satisfies Lipschitz.

Proposition 3.1. The function f satisfies Lipschitz condition in the LFR environment for a real number N.

Proof: Given the existence of f *and* $f'(\mu)$ where f is continuous in a compact interval. Let us suppose that $|f^{'}(\mu)| \leq k$ for all $\mu_x \in [\mu_a, \mu_b]$ such that $\mu_x \neq \mu_y$,

Thus, we have $\left|\frac{f(\mu_x)-f(\mu_y)}{(\mu_x-\mu_y)}\right| = |f'(\mu_c)| \leq k.$

For some $\mu_c \in [\mu_x, \mu_y]$ by the mean value theorem. Then $\left|f(\mu_x) - f(\mu_y)\right| \leq k\left|\mu_x - \mu_y\right|$.

Therefore, the function f satisfies Lipschitz condition on any interval $[\mu_a, \mu_b]$.

Since we can apply Taylor's theorem to f in LFR [10], Proposition 3.1 coupled with the application of the mean value theorem, there is a constant M, such that f *is* Lipschitz continuous in LFR.

Proposition 3.2. Suppose the function $\dddot{f}: LFR^n \rightarrow LFR$, where $\dddot{f}$ is an extension of the real function f. $\dddot{f}$ is convex and differentiable and the gradient $\nabla\dddot{f}$ is an extension of the gradient and real function ∇f. ∇f is Lipschitz continuous with constant $K > 0$. Then if we run gradient descent for j iterations with a fixed step $t \leq \frac{1}{K}$, it will yield a solution that satisfies

$$\dddot{f}\left(\mu_x^{(k)}\right) - \dddot{f}(\mu_x^*) \leq \frac{\left\|\mu_x^{(0)}-\mu_x^*\right\|^2}{2tj} \tag{1}$$

Where $\dddot{f}(\mu_x^*)$ is the optimal value. μ_x^* is the value that minimizes the function, $\dddot{f}$.

Proof

Because $\nabla\dddot{f}$ is Lipschitz continuous with constant K, $\nabla^2\dddot{f}(\mu_x)$-KI is a negative semidefinite fuzzy matrix. $\nabla^2\ddot{f}$ is an extension of the hessian and real function $\nabla^2 f$. The identity matrix, I, is a matrix in LFR[10].

Because of LFR's hybrid nature we can expand around $\ddot{f}(\mu_x)$ apply the rules of differential calculus[10]. This yields the following inequality.

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x) + \nabla\ddot{f}(\mu_x)^T(\mu_y - \mu_x) + \frac{1}{2}\nabla^2\ddot{f}(\mu_x)\,\|\mu_y - \mu_x\|^2 \tag{2}$$

In general, we wish to minimize the increase in the expansion of the area of a fuzzy number. For this reason, in the case of the expansion, we use b, the crisp maximum value of μ_x in the Jacobian and Hessian. Thus, we use a modified expansion

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x) + \nabla\ddot{f}(b)^T(\mu_y - \mu_x) + \frac{1}{2}\nabla^2\ddot{f}(b)\,\|\mu_y - \mu_x\|^2 \tag{3}$$

Where $\nabla^2\ddot{f}(b)$-KI is a semidefinite matrix. Thus

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x) + \nabla\ddot{f}(b)^T(\mu_y - \mu_x) + \frac{1}{2}K\,\|\mu_y - \mu_x\|^2 \tag{4}$$

Taking the next step in the gradient descent update by letting $\mu_y = \mu_x - t\nabla\ddot{f}(b)$.

Then

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x) + \nabla\ddot{f}(b)^T\left(-t\nabla\ddot{f}(b)\right) + \frac{1}{2}K\,\|-t\nabla\ddot{f}(b)\|^2 \tag{5}$$

$$= \ddot{f}(\mu_x) - \nabla\ddot{f}(b)^T\left(t\nabla\ddot{f}(b)\right) + \frac{1}{2}K\,\|t\nabla\ddot{f}(b)\|^2$$

$$= \ddot{f}(\mu_x) - t\|\nabla\ddot{f}(b)\|^2 + \frac{1}{2}Kt^2\,\|\nabla\ddot{f}(b)\|^2$$

$$= \ddot{f}(\mu_x) - (1 - \frac{1}{2}Kt)t\,\|\nabla\ddot{f}(b)\|^2$$

If we set $t \leq \frac{1}{K}$, we find that $-(1 - \frac{1}{2}Kt\ ) \leq -\frac{1}{2}$, thus we have

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x) - \frac{1}{2}t\,\|\nabla\ddot{f}(b)\|^2 \tag{6}$$

This inequality implies that the objective value decreases with each iteration of the gradient descent. Since $\ddot{f}$ is convex.

$$\ddot{f}(\mu_x^*) \geq \ddot{f}(\mu_x) + \nabla\ddot{f}(b)^T(\mu_x^* - \mu_x) \tag{7}$$

which implies that

$$\ddot{f}(\mu_x) \leq \ddot{f}(\mu_x^*) + \nabla\ddot{f}(b)^T(\mu_x - \mu_x^*). \tag{8}$$

Thus

$$\ddot{f}(\mu_y) \leq \ddot{f}(\mu_x^*) + \nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) - \frac{1}{2}t\,\|\nabla\ddot{f}(b)\|^2 \tag{9}$$

$$\ddot{f}(\mu_y) - \ddot{f}(\mu_x^*) \leq \nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) - \frac{1}{2}t\,\|\nabla\ddot{f}(b)\|^2$$

$$\leq \frac{1}{2t}(2t\nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) - t^2\,\|\nabla\ddot{f}(b)\|^{\,2})$$

$$\leq \frac{1}{2t}(2t\nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) - t^2\,\|\nabla\ddot{f}(b)\|^{\,2} - \|\mu_x - \mu_x^*\|^{\,2} + \|\mu_x - \mu_x^*\|^{\,2})$$

$$= \frac{1}{2t}(2t\nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) + \|\mu_x - \mu_x^*\|^{\,2} - t^2\,\|\nabla\ddot{f}(b)\|^{\,2} - \|\mu_x - \mu_x^*\|^{\,2})$$

$$= \frac{1}{2t}(2t\nabla\ddot{f}(b)^T(\mu_x - \mu_x^*) + \|\mu_x - \mu_x^*\|^{\,2} - t^2\,\|\nabla\ddot{f}(b)\|^{\,2} - \|\mu_x - \mu_x^*\|^{\,2})$$

By factoring, we have

$$= \frac{1}{2t}(\|\mu_x - \mu_x^*\|^{\,2} - \left(\|\mu_x - \mu_x^*\| - t\|\nabla\ddot{f}(b)\|\right)^2). \tag{10}$$

And thus, the triangle inequality yields

$$\ddot{f}(\mu_y) - \ddot{f}(\mu_x^*) \leq \frac{1}{2t}(\|\mu_x - \mu_x^*\|^{\,2} - \|\mu_x - \mathrm{t}\nabla\ddot{f}(b) - \mu_x^*\|)^2. \tag{11}$$

If we note that another step of the gradient descent, by definition yields $\mu_y = \mu_x - t\nabla\ddot{f}(b)$.

Then $\mu_y - \mu_x = -t\nabla\ddot{f}(b)\ and$

$$\ddot{f}(\mu_y) - \ddot{f}(\mu_x^*) \leq \frac{1}{2t}\left(\|\mu_x - \mu_x^*\|^{\,2} - \|\mu_y - \mu_x^*\|^{\,2}\right) \tag{12}$$

This holds for μ_y on every iteration of the modified gradient descent. If we sum over the iterations.

$$\sum_{i=1}^{j}\ddot{f}\left(\mu_x^{(i)}\right) - \ddot{f}(\mu_x^*) \leq \sum_{i=1}^{j}\frac{1}{2t}(\left\|\mu_x^{(i-1)} - \mu_x^*\right\|^{\,2} - \left\|\mu_x^{(i)} - \mu_x^*\right\|^{\,2}) \tag{13}$$

$$= \frac{1}{2t}(\left\|\mu_x^{(0)} - \mu_x^*\right\|^{\,2} - \left\|\mu_x^{(k)} - \mu_x^*\right\|^{\,2}$$

$$\leq \frac{1}{2t}(\left\|\mu_x^{(0)} - \mu_x^*\right\|^{\,2})$$

Given the update (6) and using the fact that *f* is decreasing. We can conclude the following:

$$\ddot{f}\left(\mu_x^{(j)}\right) - \ddot{f}(\mu_x^*) \leq \frac{1}{j}\sum_{i=1}^{j}\ddot{f}\left(\mu_x^{(i)}\right) - \ddot{f}(\mu_x^*) \tag{14}$$

$$\leq \frac{\left\|\mu_x^{(0)} - \mu_x^*\right\|^{\,2}}{2tj}$$

Which is what we sought to prove. Thus, the minimal value $\ddot{f}(\mu_x^*)$ is reached where both μ_x^* and $\ddot{f}(\mu_x^*)$ are Linear Fuzzy Real (LFR) number as defined by Definition 2.1. It follows that Proposition 3.2 applies to (1). As well as the modified form (3).

3.2. LFR gradient descent for searching for extreme values of nonlinear equations

The Modified Gradient Descent method begins with an initial approximation *LFR*, $\mu^{(0)}{}_x = \mu(a^{(0)},b^{(0)},c^{(0)})$, and generates the sequence $\{\mu_x^{(n)}\}_{n=0}^{\infty}$ with $\mu(a^{(n)},b^{(n)},c^{(n)})$, by

$$\mu_x^{(n)} = \mu_x^{(n-1)} - \frac{1}{\alpha}\nabla f(b^{n-1})^T, \text{ for } n \geq 1.$$

The stopping criterion of this method is $|b^n - b^{n-1}| < \epsilon$, where ϵ is a preset small value.

Example 3.1. *Find the local minimum of an LFR nonlinear equation* $\mu^4_x - 3\,\mu^3_x + 2$.
This is an equation with fuzzy coefficients. With typical calculation, it is expected that the local minimum occurs "around" 2.25.
If we set $f(\mu_x) = \mu^4_x - 3\,\mu^3_x + 2$, *then* $f\,'(\mu_x) = 4\mu^3_x - 9\mu^2_x$. *If we use an initial approximation* $\mu^{(0)}_x = \mu(4.5,5,5.5)$. If we set $\gamma = \frac{1}{\alpha}$. *Then we can set* $\gamma = 0.001$ *and* $\epsilon = 0.00001$. *Using Python, after 356 iterations, we find a fuzzy solution* $\mu^{(*)}_x = \mu(1.7505, 2.2505, 2.7505)$.

Example 3.2. *Find the extreme values of an LFR nonlinear equation*

$$5*(\mu_{x1})^2 + (\mu_{x2})^2 + 4\mu_{x1}*\mu_{x2} - 14\mu_{x1} - 6\mu_{x2} + 20$$

This is a multivariable equation with fuzzy coefficients. With typical calculation, it is expected that the optimal solution occurs "around" (1,1).

Let us set $F(x) = 5*(\mu_{x1})^2 + (\mu_{x2})^2 + 4\mu_{x1}*\mu_{x2} - 14\mu_{x1} - 6\mu_{x2} + 20$

then we have

$$\nabla F = \begin{bmatrix} 10*\mu_{x1} + 4*\mu_{x2} - 14 \\ 4*\mu_{x1} + 2*\mu_{x2} - 6 \end{bmatrix}$$

If we use an initial approximation $\mu^{(0)}_{x1} = \mu(-0.5,0,0.5)$, and $\mu^{(0)}_{x2} = \mu(9.5,10,10.5)$. If we *set* $\gamma = 0.02$ *and* $\epsilon = 0.000001$. *Using Python, after 59583 iterations, we find fuzzy solutions* $\mu^{(*)}_{x1} = \mu(0.44704, 0.944244, 1.44704)$ and $\mu^{(*)}_{x2} = \mu(0.63582, 1.13461, 1.63582)$.

4. LFR gradient descent for solving system of nonlinear equations

In this section, we discuss using Gradient Descent to solve system of nonlinear equations in *LFR*. To illustrate, suppose we solve the following system of two *LFR* nonlinear equations:

$$\begin{Bmatrix} f_1(\mu_{x1}, \mu_{x2}) = 0 \\ f_2(\mu_{x1}, \mu_{x2}) = 0 \end{Bmatrix} \tag{15}$$

Let the vector function F and the Jacobian matrix $J = \nabla F$ be the followings:

$$F(\mu_{x1}, \mu_{x2}) = \begin{bmatrix} f_1(\mu_{x1}, \mu_{x2}) \\ f_2(\mu_{x1}, \mu_{x2}) \end{bmatrix} \tag{16}$$

And

$$J = \begin{bmatrix} \frac{\partial f_1}{\partial \mu_{x1}} & \frac{\partial f_1}{\partial \mu_{x2}} \\ \frac{\partial f_2}{\partial \mu_{x1}} & \frac{\partial f_2}{\partial \mu_{x2}} \end{bmatrix} \tag{17}$$

Then LFR Gradient Descent method generates the solution sequences $\{\mu^{(n)}_{x1}\}^{\infty}_{n=0}$ and $\{\mu^{(n)}_{x2}\}^{\infty}_{n=0}$ such as:

$$\begin{bmatrix} \mu^{(n)}_{x1} \\ \mu^{(n)}_{x2} \end{bmatrix} = \begin{bmatrix} \mu^{(n-1)}_{x1} \\ \mu^{(n-1)}_{x2} \end{bmatrix} - \gamma\left(J\left(\mu^{(n-1)}_{x1}, \mu^{(n-1)}_{x2}\right)\right) \tag{18}$$

Where $\mu^{(n)}_{x1} = \mu(a^{(n)}_{x1}, b^{(n)}_{x1}, c^{(n)}_{x1})$ and $\mu^{(n)}_{x2} = \mu(a^{(n)}_{x2}, b^{(n)}_{x2}, c^{(n)}_{x2})$.

The stopping criterion of this method is $\min\left\{\left|b_{x1}^{(n)} - b_{x1}^{(n-1)}\right|, \left|b_{x2}^{(n)} - b_{x2}^{(n-1)}\right|\right\} < \epsilon$, where ϵ is a preset small value.

Example 4.1. *Solve an LFR system of nonlinear equations:*

$$\begin{cases} \mu_{x1} + \mu_{x2} + \mu_{x3} = 6 \\ \mu_{x1}^2 + \mu_{x2} = 5 \\ \mu_{x2}^2 + \mu_{x3} = 4 \end{cases}$$

We easily find F and J as

$$F(\mu_{x1}, \mu_{x2}, \mu_{x3}) = \begin{bmatrix} \mu_{x1} + \mu_{x2} + \mu_{x3} - 6 \\ \mu_{x1}^2 + \mu_{x2} - 5 \\ \mu_{x2}^2 + \mu_{x3} - 4 \end{bmatrix}$$

And

$$J = \begin{bmatrix} 1 & 1 & 1 \\ 2\mu_{x1} & 1 & 0 \\ 0 & 2\mu_{x2} & 1 \end{bmatrix}$$

With typical calculation, it is expected that the optimal solution occurs "around" (2,1,3). If we use initial approximations $\mu_{x1}^{(0)} = \mu_{x2}^{(0)} = \mu_{x3}^{(0)} = \mu(-1,2,4)$. *And set* $\gamma = 0.02$ *and* $\epsilon = 0.000001$.

Then in Python after 121193 iterations, LFR Gradient Descent Method yields

$\mu^{(*)}{}_{x1} = \mu(-1.004, 1.996, 3.996)$

$\mu^{(*)}{}_{x2} = \mu(-1.978, 1.022, 3.022)$

$\mu^{(*)}{}_{x3} = \mu(-0.035, 2.965, 4.965)$

5. Conclusion

We often do not know the consequence of each possible decision with precision, sometimes we can only know this case with uncertainty. In many cases the LFR approach may be a good approach. The simplest case for example would be tolerance, where all possible values within the interval are equally likely. LFR presents an advantage where it may represent an interval, a crisp point, or a fuzzy number that can be written discretely or continuously. This is an advantage in the case of certain problems where it is already known that "crisp optima" in the purest sense do not exist, but projecting to the middle $\mu(a,b,c) \rightarrow \mu(b,b,b) = b$ produces a "crisp good choice" for an optimal value in optimization [12] and machine learning. It is also possible to approximate other fuzzy number types, using LFR. In this paper, we used LFR coupled with the Gradient Descent to solve and optimize Fuzzy Nonlinear problems and systems of fuzzy problems. As has been shown, LFR is a viable environment for machine learning techniques.

Acknowledgments

This work has been financially supported by Project Engage: ROAR a grant funded by the United States Department of Education. Grant number: P120A160122.

Conflict of interest

The author declares that there is no conflict of interests.

References

1. R. E. Bellman, L. A. Zadeh, *Decision making in a fuzzy environment*, Manage. Sci., **17** (1970), 141–164.
2. J. Cheng, *Data-Mining Research in Education*, Report, International School of Software, Wuhan University, 2017.
3. D. Dubois, H. Prade, *System of linear fuzzy constraints*, Fuzzy Set. Syst., **3** (1980), 37–48.
4. B. Monk, *A Proposed Theory of Fuzzy Random Variables*, Dissertation, University of Alabama, 2001.
5. A. C. Muller, S. Guido, *Introduction to Machine Learning with Python: A Guide for Data Scientists*, O'Reilly Media, 2016.
6. J. Neggers, H. Kim, *Fuzzy posets on sets*, Fuzzy Set. Syst., **117** (2001), 391–402.
7. J. Neggers, H. Kim, *On Linear Fuzzy Real Numbers*, Manuscript for book under development, 2007.
8. C. V. Negoita, *Fuzziness in management*, OPSA/TIMS, Miami, 1970.
9. R. Prevo, *Entropies of families of fuzzy random variables: an introduction to an in-depth exploration of several classes of important examples*, Dissertation, University of Alabama, 2002.
10. F. Rogers, *Optimal Choices in an LFR System*, Dissertation, University of Alabama, 2005.
11. F. Rogers, Y. Jun, *Fuzzy Nonlinear Optimization for the Linear Fuzzy Real Number System*, International Mathematical Forum, **4** (2009), 587–596.
12. N. V. Sahindas, *Optimization under uncertainty: state-of-the-art and opportunities*, Comput. Chem. Eng., **28** (2004), 971–983.
13. H. Tanaka, K. Asai, *Fuzzy linear programming problems with fuzzy numbers*, Fuzzy Set. Syst., **13** (1984), 1–10.
14. H. J. Zimmermann, *Fuzzy mathematical programming*, Comput. Oper. Res., **10** (1983), 291–298.