Raport Badawczy
Research Report

RB/4/2017

Interval based, non-parametric approach for resampling of fuzzy numbers

M. Romaniuk, O. Hryniewicz

Instytut Badań Systemowych
Polska Akademia Nauk

Systems Research Institute
Polish Academy of Sciences

POLSKA AKADEMIA NAUK

Instytut Badań Systemowych

ul. Newelska 6

01-447 Warszawa

tel.: (+48) (22) 3810100

fax: (+48) (22) 3810105

Kierownik Zakładu zgłaszający pracę:
Prof. dr hab. inż. Olgierd Hryniewicz

Warszawa 2017

Interval based, non-parametric approach for resampling of fuzzy numbers

Maciej Romaniuk* Olgierd Hryniewicz†

September 19, 2017

Keywords: *fuzzy numbers; alpha-cuts; resampling method; bootstrap; generation of fuzzy numbers; Monte Carlo method.*

Abstract

In the paper we propose two new non-parametric methods for the simulation of bootstrap-like samples of fuzzy numbers. The generated secondary samples are based on an input set (i.e., a primary sample) consisting of LFRNs (Left-Right Fuzzy Numbers). The proposed approaches utilize Monte Carlo method in a way, which, to some extent, resembles a bootstrap. However, contrary to the classical bootstrap approach, the discussed methods are based on alpha-cuts of fuzzy numbers, which are generated in a new non-parametric way. Therefore, these procedures give us an opportunity to create "not exactly

*Systems Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01–447 Warszawa, Poland, The John Paul II Catholic University of Lublin, ul. Konstantynów 1H, 20–708 Lublin, Poland, *e-mail: mroman@ibspan.waw.pl*

†Systems Research Institute, Polish Academy of Sciences, ul. Newelska 6, 01–447 Warszawa, Poland, *e-mail: olgierd.hryniewicz@ibspan.waw.pl*

the same as previous" fuzzy numbers, and also lead to greater variability of the obtained output.

1 Introduction

Monte Carlo simulations are widely used for the numerical analysis of complex phenomena. For example, they are used in solving optimization problems, simulation of complex probabilistic models, evaluation of different properties of statistical tests, etc. A good description of main problems and applications of Monte Carlo simulations can be found in well-known textbooks, such as, e.g., Robert and Casella [23]. In general, Monte Carlo simulations are used when considered mathematical models are too complex for analytical evaluation. We face such problems very frequently when we have to deal with complex phenomena that are characterized not only by randomness, but also by fuzzy imprecision, as well. In all such cases for the description of real-life problems we may use the concept of fuzzy random variables.

There exist several models of fuzzy random variables having different practical interpretation. They are described in a very good monograph by Couso *et al.* [3] or in overview papers [4, 5]. One of the two most popular approaches is called "ontic". This model was proposed in the seminal paper by Puri and Ralescu [19], and is based on the notion of set-valued mapping and random sets. Simulation methods of fuzzy random variables having "ontic" interpretation have attracted recently attention of many researchers. For example, Colubi *et al.* [2] consider simulation methods for different types of both one- and multidimensional fuzzy variables. They use these methods for the analysis of asymptotic behavior of a fuzzy arithmetic mean, expressed in terms of the strong law of large numbers, and of the law of iterated logarithm. The process of simulation itself was thoroughly examined in the paper by González-Rodríguez *et al.* [7]. They have proposed two different approaches based on the concept of support functions. The first one is related to simulations of Hilbert space-valued random elements with a projection on the cone of all fuzzy sets. The second one imitates the representation of elements of a separable Hilbert space for an orthonormal basis directly on the space of fuzzy sets. Both of these approaches were compared, and their comparison has shown that the second method is more adequate for modeling realistic situations.

The second most popular interpretation of fuzzy random variables is historically older, and is based on the model proposed in the papers by Kwakernaak [14, 15]. In this model a fuzzy random variable describes imprecise (fuzzy) perception of an unobserved crisp random variable. This model has been applied for solving many real-life problems. Its applications have been described in numerous papers. For example, the authors of this paper used it to solve problems from such different areas as: pricing of financial and insurance instruments [17, 18], estimation of the maintenance costs of a water distribution system [24] or Bayesian statistical decisions in reliability [11]. For nearly all such problems, Monte Carlo simulations of fuzzy random variables have been extensively used. They usually consisted in the generation of random hidden crisp origins, and respective membership functions (e.g., in a form of triangles with edges of random length).

In this paper we focus on the one aspect of simulations related to the first, "ontic", aproach, namely resampling of sets of fuzzy numbers. The widely used in statistics resampling method , known as the bootstrap approach, has been applied in the analysis of random fuzzy data. For example, in [9], the weighted bootstrap approach for fuzzy numbers is used to calculate the standard error of the minimum inaccuracy estimator, and to construct an appropriate confidence interval. The bootstrap is also an important tool in evaluation of a distribution of a test statistic, if such a distribution is too complex for a direct calculation. For example, it is used for tests about the expected value of a fuzzy random variable (see, e.g., [6, 8, 16]), and other types of statistical tests in a fuzzy environment (see, e.g., [20, 21]). In the aforementioned papers bootstrap samples enable the authors of such tests to estimate a nominal significance level of a statistical test via an empirical percentage of rejections of a true null hypothesis, and this bootstrap-based estimator serves then as an empirical benchmark for the considered statistical test. In our approach, described in this paper, we propose a more general resampling method, which consists in the modification of the existing bootstrap procedure with the aim to improve statistical properties of bootstrap estimators, especially for small available samples of fuzzy observations. In contrast to the classical bootstrap, our method generates fuzzy numbers that may be different from the numbers included in the original sample. Therefore, our new method of resampling may be considered as a bootstrap-like generation method.

Our contribution in this article is fourfold. Firstly, two numerical methods for simulation of the left-right fuzzy numbers (LRFNs) are considered. These algorithms, similarly to the classical bootstrap method, utilize a primary (initial) sample of random fuzzy numbers in order to generate secondary (bootstrap) fuzzy random samples. But, contrary to the classical bootstrap, these simulated secondary sets consist of values, which are "not exactly the same" as in the initial sample. In the first method, a modified direct method (coined the *d-method*, and described by a discrete probability distribution $d(x)$), an overall information about the α-cuts of the LRFNs from the primary set is used. In the second method (coined the *w-method*) a mixed discrete-uniform probability distribution $w(x)$ is used for generation purposes. In this approach the information about the α-cuts of the observations from the primary sample is modified in a certain way using a non-informative uniform distribution. Both proposed methods are used to generate LRFNs, whose variability is in a certain sense greater than the variability of observations from the primary sample. However, this greater variability has been achieved without incorporation of any additional and specific assumptions about the general probability model for the initial population. Hence, both of these approaches are strictly non-parametric ones.

Secondary, the outputs for these two methods are analyzed, using the most important statistical measures. For both small and moderate primary samples, and two types of the triangular fuzzy numbers, we check, if the generated secondary (bootstrap-like) samples well imitate statistical behavior of the initial population. In order to do this, mean and standard deviation are calculated, and applicability of the strong law of large numbers and the law of iterated logarithm has been confirmed. We also compare the simulated secondary samples for the two introduced methods with the output of the classical bootstrap approach. It seems, that the application of $d(x)$ and

$w(x)$ distributions in bootstrapping is very promising, because the generated triangular numbers "mimic" the values from the initial sample very well. Moreover, if the previously mentioned statistical measures are taken into account, these generated values sometimes behave even better than in the case of classical bootstrap approach applied for the same primary samples.

Thirdly, for the same sizes of samples, and two types of the triangular fuzzy numbers, we check if the simulated values are "close enough" to the fuzzy numbers from the initial set. A level of this proximity is measured using four types of measures (the supremum measure, the l_1 metric, the Hausdorf distance extended to the metric, and the measure proposed by Tran and Duckstein [25]). Once again, the obtained results are compared with the outcomes for the classical bootstrap approach. The performed analysis confirms a thesis that fuzzy numbers generated using $d(x)$ and $w(x)$ distributions are very close to observations from the primary sample, used in the classical bootstrap. Therefore, the two simulation procedures introduced in this paper can be used to form the secondary (bootstrap-like) sample, which is "similar", but also, in some way, different, in comparison to the initial set of observations.

Fourthly, we check if these two new simulation algorithms can be successfully applied for solving some practical statistical problems. As an example, we have applied them for two statistical tests about the mean value of a population of fuzzy numbers. In these two tests, outputs for both small and moderate primary samples have been analyzed for three types of the triangular fuzzy numbers. As previously, we have compared three simulation procedures (the classical bootstrap, and our two methods based on the $d(x)$

and $w(x)$ distributions). In all considered cases, a difference between a nominal significance level of the test, and an empirical percentage of rejections of the true null hypothesis is used as a benchmark. Once again, the algorithms introduced in this paper show their promising potential, because the mentioned difference is usually lower for the proposed bootstrap-like procedures, based on the $d(x)$ or $w(x)$ distributions, than for the classical bootstrap of fuzzy random variables.

The paper is organized as follows. In Section 2 basic definitions of fuzzy sets and random fuzzy sets have been recalled. Moreover, we have presented the descriptions of statistical tests used for testing hypotheses about the expected value. Next, in Section 3, we describe the proposed algorithms for the generation of bootstrap-like secondary samples. Then, in Section 4 we describe the results of the experimental verification of the properties of the proposed procedures. The application of the proposed new bootstrap procedures in statistical testing has been presented in Section 5. The paper is concluded in its last section.

2 Mathematical preliminaries

2.1 Fuzzy sets and random fuzzy sets

Let us present basic definitions and notation concerning the simulation of fuzzy random variables which will be used in this paper. Additional details can be found in, e.g., [4, 5].

For a fuzzy subset $\tilde{A}$ of the set of real numbers $\mathbb{R}$ we denote by $\mu_{\tilde{A}}$ its membership function $\mu_{\tilde{A}} : \mathbb{R} \rightarrow [0, 1]$, and by $\tilde{A}(\alpha) = \{x : \mu_{\tilde{A}}(x) \geq \alpha\}$

the α-level set of $\tilde{A}$ for $\alpha \in (0,1]$. Then, $\tilde{A}(0)$ is the closure of the set $\{x : \mu_{\tilde{A}}(x) > 0\}$.

A fuzzy number $\tilde{a}$ is a fuzzy subset of $\mathbb{R}$ for which $\mu_{\tilde{a}}$ is a normal, upper-semicontinuous, fuzzy convex function with a compact support. Then, for each $\alpha \in [0,1]$, the α-level set $\tilde{a}(\alpha)$ is a closed interval of the form $\tilde{a}(\alpha) = [a^L(\alpha), a^R(\alpha)]$, where $a^L(\alpha), a^R(\alpha) \in \mathbb{R}$ and $a^L(\alpha) \leq a^R(\alpha)$.

A left-right fuzzy number (which is further abbreviated as LRFN) is a fuzzy number with the membership function of the form

$$\mu_{\tilde{a}}(x) = \begin{cases} L\left(\frac{x-a}{b-a}\right) & \text{if } x \in [a,b] \\ 1 & \text{if } x \in [b,c] \\ R\left(\frac{d-x}{d-c}\right) & \text{if } x \in [c,d] \\ 0 & \text{otherwise} \end{cases},$$

where $L, R : [0,1] \to [0,1]$ are non-decreasing functions such that $L(0) = R(0) = 0$ and $L(1) = R(1) = 1$.

A triangular fuzzy number $\tilde{a}$, denoted further by $\left[a^L, a^C, a^R\right]$, is a LRFN with the membership function of the form

$$\mu_{\tilde{a}}(x) = \begin{cases} \frac{x-a^L}{a^C-a^L} & \text{if } x \in \left[a^L, a^C\right] \\ \frac{a^R-x}{a^R-a^C} & \text{if } x \in \left[a^C, a^R\right] \\ 0 & \text{otherwise} \end{cases},$$

where a^L is a left end of its support, a^C – its core, and a^R – a right end of its support.

2.2 Measures of similarity

To compare some properties of two fuzzy numbers, like their shape or location, various measures of similarity can be used. In this paper, we focus on classical measures: the supremum, the l_1 metric, the Hausdorff metric for fuzzy sets (see, e.g., [26] for additional details), and the distance measure introduced in [25]. Of course, other types of measures of similarity can be also used to compare differences between the LRFNs generated according to methods described in Section 3.

If $\tilde{a}$ and $\tilde{b}$ are fuzzy sets, then the supremum similarity measure is defined for their membership functions $\mu_{\tilde{a}}(x)$ and $\mu_{\tilde{b}}(x)$, as

$$m_{\infty}\left(\tilde{a},\tilde{b}\right) = \sup_x \left|\mu_{\tilde{a}}(x) - \mu_{\tilde{b}}(x)\right| \ .$$

In the case of the l_1 metric, an appropriate measure is given by

$$m_{l_1}\left(\tilde{a},\tilde{b}\right) = \int_{-\infty}^{\infty} \left|\mu_{\tilde{a}}(x) - \mu_{\tilde{b}}(x)\right| dx \ .$$

There are various ways to extend the Hausdorf distance to a metric for fuzzy sets (see, e.g., [26]). In the following, we use

$$m_H\left(\tilde{a},\tilde{b}\right) = \int_0^1 \max\left\{\left|a^L(\alpha) - b^L(\alpha)\right|, \left|a^R(\alpha) - b^R(\alpha)\right|\right\} d\alpha \ .$$

This measure was proposed by Ralescu and Ralescu in [22].

The fourth distance measure was introduced by Tran and Duckstein [25], and it is given by

$$m_{TD}\left(\tilde{a},\tilde{b}\right) = \int_0^1 \left(\left(\frac{a^L(\alpha)+a^R(\alpha)}{2} - \frac{b^L(\alpha)+b^R(\alpha)}{2}\right)^2 \right.$$
$$\left. + \frac{1}{3}\left(\left(\frac{a^L(\alpha)-a^R(\alpha)}{2}\right)^2 + \left(\frac{b^L(\alpha)-b^R(\alpha)}{2}\right)^2\right)\right) \times w(\alpha)d\alpha$$
$$/ \int_0^1 w(\alpha)d\alpha \ , \quad (1)$$

where $w(\alpha)$ is a certain weighting function. In the following $w(\alpha) = 1$ is used, so each α-cut in the measure (1) has the same significance (see [25] for other possible types of the weighting function).

2.3 Tests of a fuzzy mean value

There are many types of statistical tests for an expected value of a fuzzy random variable (see, e.g., [6, 8, 12, 16]). We focus on only two of them, which will be used in Section 5 as examples of an application of the introduced non-parametric simulation methods.

The first considered test is an asymptotic test introduced in [12]. Let us assume, that $\tilde{a}$ is a LRFN with a core, which is given by a single value. Then, we have

$$m_a = a^L(1) = a^R(1), l_a = m_a - a^L(0), r_a = a^R(0) - m_a \ .$$

The d_2 distance between two LRFNs $\tilde{a}$ and $\tilde{b}$ is defined as

$$\begin{aligned} d_2^2\left(\tilde{a}, \tilde{b}\right) = & \left|m_a - m_b\right|^2 + R_2\left|r_a - r_b\right|^2 + L_2\left|l_a - l_b\right|^2 \\ & + 2\left(m_a - m_b\right)\left(R_1\left(r_a - r_b\right) - L_1\left(l_a - l_b\right)\right) \ , \end{aligned}$$

where

$$L_2 = \frac{1}{2}\int_0^1 \left|L^{(-1)}(\alpha)\right|^2 d\alpha \ , \ L_1 = \frac{1}{2}\int_0^1 L^{(-1)}(\alpha) d\alpha$$

and R_1, R_2 are defined similarly (see [12]).

Using this distance, we have the following corollary, which was proved in [12]:

Corollary 1. *Let $X_1, X_2, \ldots, X_n$ be a sample of LRFNs. Then*

$$n d_2^2\left(\bar{X}, \mathbb{E}\, X\right) \xrightarrow[n \to \infty]{} \lambda_1 \xi_1^2 + \lambda_2 \xi_2^2 + \lambda_3 \xi_3^2 \ ,$$

where ξ_1, ξ_2, ξ_3 *are independent* $N(0,1)$*-distributed random variables, and* $\lambda_1, \lambda_2, \lambda_3$ *are the eigenvalues of the matrix*

$$\begin{pmatrix} C_{m_X m_X} - L_1 C_{l_X m_X} + R_1 C_{r_X m_X} & L_2 C_{l_X m_X} - L_1 C_{m_X m_X} & R_1 C_{m_X m_X} + R_2 C_{r_X m_X} \\ C_{l_X m_X} - L_1 C_{l_X l_X} + R_1 C_{r_X l_X} & L_2 C_{l_X l_X} - L_1 C_{l_X m_X} & R_1 C_{l_X m_X} + R_2 C_{r_X l_X} \\ C_{r_X m_X} - L_1 C_{r_X l_X} + R_1 C_{r_X r_X} & L_2 C_{r_X l_X} - L_1 C_{r_X m_X} & R_1 C_{r_X m_X} + R_2 C_{r_X r_X} \end{pmatrix}$$

where $C_{zy} = \mathbb{E}(z - \mathbb{E}\, z)\,\mathbb{E}(y - \mathbb{E}\, y)$ *for* $z, y \in \{m_X, l_X, r_X\}$*. Moreover, an asymptotic test of the hypothesis*

$$H_0 : \mathbb{E}\, X = \tilde{V} \text{ against } H_1 : \mathbb{E}\, X \neq \tilde{V}$$

is formulated as follows: reject H_0*, if*

$$n d_2^2 \left(\bar{X}, \tilde{V} \right) > \omega_{1-p}^2 \ ,$$

where w_q^2 *is the* q*-th quantile of a* ω^2 *distribution with respect to the eigenvalues* $\lambda_1, \lambda_2, \lambda_3$*.*

The above mentioned ω^2 distribution has a rather complex structure, which is known only for some special cases (see [12]).

The second considered test was developed in [8, 16]. It is based on a metric introduced in [1], which was generalized in [13]. The D_W^{φ} metric for two LRFNs $\tilde{a}, \tilde{b}$ is defined as

$$D_W^{\varphi} \left(\tilde{a}, \tilde{b} \right) = \sqrt{\int_0^1 d_w^2 \left(\tilde{a}(\alpha), \tilde{b}(\alpha) \right) d\varphi(\alpha)} \ , \tag{2}$$

where

$$d_w^2 \left(\tilde{a}(\alpha), \tilde{b}(\alpha) \right) = \int_0^1 \left(f_{\tilde{a}}(\alpha, \lambda) - f_{\tilde{b}}(\alpha, \lambda) \right)^2 dW(\lambda)$$

with $f_{\tilde{a}}(\alpha, \lambda) = \lambda a^R(\alpha) - (1 - \lambda) a^L(\alpha)$, and W, φ are two weighting normalized measures (see [1] for some examples of W, φ and further details).

Then, we have the following corollary, which was established in [8]:

Corollary 2. *Let $X_1, X_2, \ldots, X_n$ be a sample of LRFNs. In testing the null hypothesis*

$$H_0 : \mathbb{E}X = \tilde{V}$$

at the nominal significance level p, H_0 should be rejected, if

$$\frac{D_W^{\varphi}\left(\bar{X}, \tilde{V}\right)^2}{\hat{S}^2} > z_{1-p} \ ,$$

where z_q is the q-th empirical quantile of the bootstrap distribution, which is given by

$$\frac{D_W^{\varphi}\left(\bar{X}^*, \bar{X}\right)^2}{\hat{S}_*^2}$$

and with

$$\bar{X}^* = \frac{1}{n}\sum_{i=1}^{n} X_i^* \ , \ \hat{S}_*^2 = \frac{1}{n-1}\sum_{i=1}^{n} D_W^{\varphi}\left(X_i^*, \bar{X}^*\right)^2 \ ,$$

where $X_1^*, X_2^*, \ldots, X_n^*$ is a bootstrap sample obtained from the initial sample $X_1, X_2, \ldots, X_n$.

3 Generation of a secondary (bootstrap) sample

Let $\mathcal{A} = \{\tilde{a}_1, \ldots, \tilde{a}_m\}$ be a primary sample of LRFNs. These values are treated as an input set for the method proposed further in this paper. We assume that we do not have (and, moreover, we do not need) any additional information about a source (or a model) of the fuzzy numbers belonging to $\mathcal{A}$. Note, however, that in many cases known from literature such additional information is often assumed (see, e.g., [2, 10, 11, 17, 18, 24] for various approaches to the problem of fuzzy numbers modeling). Therefore, only a strictly non-parametric way should be used to build a secondary (bootstrap) sample $\mathcal{B} = \{\tilde{b}_1, \ldots, \tilde{b}_n\}$ of LRFNs, which should be, in some way, "similar" to the fuzzy numbers from A.

Let $\tilde{a}_j(\alpha) = \left[a_j^L(\alpha), a_j^R(\alpha)\right]$ be an α-cut of $\tilde{a}_j$ for some $\alpha \in [0,1]$. For simplicity, we assume that there are $k+1$ possible values of α, so we have $\alpha \in \{\alpha_0, \alpha_1, \ldots, \alpha_k\}$, where $\alpha_0 = 0 < \alpha_1 < \ldots < \alpha_k = 1$. We also assume that $a_j^L(1) = a_j^R(1) = a_j(1)$ for each $\tilde{a}_j$. However, this requirement can be easily relaxed in a simulation procedure presented further.

During the first step of an initialization procedure (a setup of simulation, see Algorithm 1), a set of cores $\mathcal{C}(1)$ is found, based on $\mathcal{A}$. Hence, we have

$$\mathcal{C}(1) = \{a_1(1), \ldots, a_m(1)\} \ .$$

For simplicity of notation, we assume that the set $\mathcal{C}(1)$ is already ordered, i.e., $a_1(1) \leq a_2(1) \leq \ldots \leq a_k(1)$.

During the second step of a initialization procedure, sets of incremental spreads for all of possible α-cuts are constructed. Let

$$s_j^L(\alpha_i) = a_j^L(\alpha_{i+1}) - a_j^L(\alpha_i) \tag{3}$$

be a difference between left ends of α-cuts for α_{i+1} and α_i, for the given fuzzy number $\tilde{a}_j$. We call such a difference an *incremental left spread* for the level i. In the same manner, we have

$$s_j^R(\alpha_i) = a_j^R(\alpha_i) - a_j^R(\alpha_{i+1}) \ , \tag{4}$$

which is a difference between right ends of α-cuts for α_i and α_{i+1}, for the given fuzzy number $\tilde{a}_j$. It will be called as an *incremental right spread* for the level i. Then the sets of left and right incremental spreads, given by

$$\mathcal{S}^L(\alpha_i) = \{s_1^L(\alpha_i), \ldots, s_m^L(\alpha_i)\} \ , \mathcal{S}^R(\alpha_i) = \{s_1^R(\alpha_i), \ldots, s_m^R(\alpha_i)\} \tag{5}$$

Algorithm 1: Initialization procedure

Input: A primary sample $\mathcal{A} = \{\tilde{a}_1, \ldots, \tilde{a}_m\}$, a number of possible α-cuts $k+1$.

Output: Sets of the cores and the spreads for $\mathcal{A}$.

Find a set of values of the cores $\mathcal{C}(1) = \{a_1(1), \ldots, a_m(1)\}$ *and order it increasingly*;

for $i \leftarrow k-1$ **to** 0 **do**

  for $j \leftarrow 1$ **to** m **do**

    Find an incremental left spread $s_j^L(\alpha_i) = a_j^L(\alpha_{i+1}) - a_j^L(\alpha_i)$;

    Find an incremental right spread $s_j^R(\alpha_i) = a_j^R(\alpha_i) - a_j^R(\alpha_{i+1})$;

    Append $s_j^L(\alpha_i)$ *to a set* $\mathcal{S}^L(\alpha_i)$ *and* $s_j^R(\alpha_i)$ *to a set* $\mathcal{S}^R(\alpha_i)$;

  end

  Order sets $\mathcal{S}^L(\alpha_i)$ *and* $\mathcal{S}^R(\alpha_i)$ *increasingly*;

end

for $\alpha_{k-1}, \alpha_{k-2}, \ldots, \alpha_0$ can be found. It should be noted, that the construction of (5) has to be made from the highest value of α to the lowest one (i.e. from a core of a fuzzy number to its support). We also assume, in the same manner as for the set of cores $\mathcal{C}(1)$, that each of the sets (5) is already ordered, so

$$0 \leq s_1^L(\alpha_i) \leq \ldots \leq s_m^L(\alpha_i)\ ,\ 0 \leq s_1^R(\alpha_i) \leq \ldots \leq s_m^R(\alpha_i)$$

for all α_i.

Let us illustrate this initialization procedure with a numerical toy-example.

Example 1. Suppose that our primary sample consists of only three triangular fuzzy numbers $[0, 1, 3], [1, 2.5, 5]$ and $[1, 3.5, 5]$. We consider only 3 different α-levels: $\alpha_2 = 1$ (cores), $\alpha_1 = 0.5$, and $\alpha_0 = 0$ (supports). For these data the set of cores is $\mathcal{C}(1) = \{1, 2.5, 3.5\}$. The ordered sets of incremental left spreads are $\mathcal{S}^L(\alpha_1) = \{0.5, 0.75, 1.25\}$, and $\mathcal{S}^L(\alpha_0) = \{0.5, 0.75, 1.25\}$, respectively. The ordered sets of incremental right spreads are $\mathcal{S}^R(\alpha_1) = \{0.75, 1, 1.25\}$, and $\mathcal{S}^R(\alpha_0) = \{0.75, 1, 1.25\}$, respectively.

Now, the secondary sample $\mathcal{B}$, which consists of n fuzzy numbers, can be generated. In order to do this, we use one of further proposed two methods, based on two kinds of distributions.

3.1 The d-method based on a discrete distribution $d(x)$

Let us start from a description of a generation procedure in case of the d-method, based on a discrete probability distribution $d(x)$. In the proposed procedure two steps are necessary to construct a fuzzy number $\tilde{b}_j \in \mathcal{B}$, if $j = 1, \ldots, n$ (see also Algorithm 2).

Firstly, a value of a core $b_j(1)$ is found, using a uniform discrete distribution for the values from the set $\mathcal{C}(1)$. It means, that the generated value $b_j(1) = C$ is a random element taken from the set $\mathcal{C}(1)$ according to the probability distribution $d(x)$. In this paper we assume that $d(x)$ is uniform on $\mathcal{C}(1)$, i.e.

$$\Pr\left(C = a_l(1)\right) = d(a_l(1)) = \frac{1}{m} ,$$

if $l = 1, \ldots, m$. Therefore, we randomly (and uniformly) pick up a single value from the set $\mathcal{C}(1)$ and treat it as the core of the new, constructed LRFN $\tilde{b}_j$.

Secondly, consecutive α-cuts of the given $\tilde{b}_j$ are found, starting from its core and ending at its support. Thus, we proceed from $\tilde{b}_j(\alpha_{k-1})$ until $\tilde{b}_j(0)$, For each α_i, a value of a left end of the α-cut of $\tilde{b}_j$ is found, using

$$b_j^L(\alpha_i) = b_j^L(\alpha_{i+1}) - S^L(\alpha_i) , \tag{6}$$

where $S^L(\alpha_i)$ is an independently drawn random value from the set $\mathcal{S}^L(\alpha_i)$. Once again, the uniform discrete distribution $d(x)$ is used, for which

$$\Pr\left(S^L(\alpha_i) = s_l^L(\alpha_i)\right) = d(s_l^L(\alpha_i)) = \frac{1}{m} ,$$

where $l = 1, \ldots, m$. In the same manner, a right end of each α-cut of $\tilde{b}_j$ is constructed, using

$$b_j^R(\alpha_i) = b_j^R(\alpha_{i+1}) + S^R(\alpha_i) , \tag{7}$$

where $S^R(\alpha_i)$ is independently drawn from the set $\mathcal{S}^R(\alpha_i)$, using the same uniform discrete distribution $d(x)$. Formulas (6) and (7) mean, that the new left (or right, respectively) end of α_i-cut is constructed, based on subtracting (or adding) a random element from the set $\mathcal{S}^L(\alpha_i)$ (or $\mathcal{S}^R(\alpha_i)$) from (to) the previously generated left (right) end of α_{i+1}-cut. Therefore, this new fuzzy number $\tilde{b}_j$ is approximated using intervals for the cosecutive values of α (from 1 at the top to 0 at the bottom).

The generated in this way fuzzy number $\tilde{b}_j$ is, in a certain sense, similar to the LRFNs from the primary sample $\mathcal{A}$. The core of $\tilde{b}_j$ is one of the "true" cores from $\mathcal{C}(1)$, and its spreads are drawn from the "true" spreads belonging to $\mathcal{S}^L(\alpha_i)$ or $\mathcal{S}^R(\alpha_i)$. It is easily seen that we have

$$\mathbb{E}C = \frac{1}{m}\sum_{l=1}^{m} a_l(1) = \bar{a}(1) , \; \mathbb{E}S^L(\alpha_i) = \frac{1}{m}\sum_{l=1}^{m} s_l^L(\alpha_i) = \bar{s}^L(\alpha_i) ,$$

$$\mathbb{E}S^R(\alpha_i) = \bar{s}^R(\alpha_i) ,$$

so the expected values of core and spreads of $\tilde{b}_j$ are precisely equal to the respective means for LRFNs from $\mathcal{A}$. In the same way,

$$\mathrm{Var}\, C = \frac{1}{m}\sum_{l=1}^{m}\left(a_l(1) - \bar{a}(1)\right)^2 = s^2_{a(1)}\ , \ \ \mathrm{Var}\, S^L(\alpha_i) = s^2_{s^L(\alpha_i)}\ ,$$

$$\mathrm{Var}\, S^R(\alpha_i) = s^2_{s^R(\alpha_i)}\ ,$$

then $\tilde{b}_j$ exactly "imitates" statistical behaviour of the samples from $\mathcal{A}$, without necessity of introducing any additional knowledge about the model, which (perhaps) creates the primary sample.

Now, let us continue our example by showing how the secondary bootstrap-like sample is constructed. We will show the construction of only one element of this sample. The remaining elements are constructed in the same way.

Example 1 (Continued)**.** The core of a new element of the secondary sample is, in this example, randomly chosen (with equal probabilities 1/3) from the set $\{1, 2.5, 3.5\}$, and let this chosen value be equal to $b_1^L(1) = b_1^R(1) = 1$. Then, we take randomly (also with equal probabilities 1/3) the left and right incremental spreads on the remaining two α-levels. Suppose that for $\alpha = 0.5$ we have chosen $S_1^L(0.5) = 0.75$, $S_1^R(0.5) = 1.25$, and for $\alpha = 0$ we have chosen $S_1^L(0) = 1.25$, $S_1^R(0.5) = 1$. Thus, the respective α-cuts of the new element $\tilde{b}_1$ of the secondary sample, calculated according to (6)–(7), are defined by the following limits: $b_1^L(0.5) = -0.25, b_1^R(0.5) = 2.25$, and $b_1^L(0) = -1.5, b_1^R(0) = 3.25$.

3.2 The w-method based on a mixed discrete-uniform distribution $w(x)$

We can be also interested in an additional level of "freedom" in creation of the secondary sample $\mathcal{B}$. It is easy to see that if $\tilde{b}_j$ is generated using the method described in Section 3.1 (i.e., using the uniform discrete distribution $d(x)$), its core is exactly equal to one of the values from $\mathcal{C}(1)$. Also its spreads are given by the respective values from the sets $\mathcal{S}^L(\alpha_i)$ or $\mathcal{S}^R(\alpha_i)$.

But creation of a more diversified sample $\mathcal{B}$ can be fruitful in some cases. Because of such diversification the values from $\mathcal{B}$ could be "closer" to the (unknown) hidden model, than the samples from $\mathcal{A}$, especially if a number of the elements in $\mathcal{A}$ is strictly limited. Consider, for example, the case when there are only two fuzzy numbers in $\mathcal{A}$ described by only two α-cuts. The random numbers $\tilde{b}_j$, generated using the method described in Section 3.1, have no more than two possible values of a core and four possible left / right ends of its support. Moreover, if a more classical resampling method is taken into account (like the "classical" bootstrap), then these two elements from $\mathcal{A}$ are repeated many times during a construction of the LRFNs from $\mathcal{B}$. Thus, no new "knowledge" about other possible outcomes which can be possibly "produced" by the unknown model can be obtained.

Of course, apart from the introduction of the diversification, the secondary sample $\mathcal{B}$ should be still enough "similar" to the primary set $\mathcal{A}$. If such a requirement is not fulfilled, then our knowledge resulting from $\mathcal{B}$ can be misleading, and our suppositions about the original source (i.e., the model of $\mathcal{A}$) can be incorrect. But no strict prior knowledge about the model for the primary sample was previously assumed in this paper. Therefore, a proposed generation method should be a strictly non-parametric one, without any additional, more detailed, assumptions.

If we do not want to introduce in statistics any prior knowledge, we have to use so called non-informative probability distributions. A commonly used model of such a distribution is a uniform density for an interval $[c, d]$, denoted further by $U([c, d])$. We will use this density in the construction of the probability distribution used for generation purposes.

3.2.1 The $w(x)$ distribution, and its properties

Let

$$x_1 < x_2 < \ldots < x_m \tag{8}$$

be a strictly increasing sequence of m values, without their repetitions. We propose a density $w(x)$ which is the composition of discrete (atomic) and continuous parts given by the following formula

$$\begin{aligned} w(x) = \frac{1}{2m}\mathbb{1}(x = x_1) + \frac{1}{m}w_{1,2}(x) + \frac{1}{m}w_{2,3}(x) + \ldots \\ + \frac{1}{m}w_{m-1,m}(x) + \frac{1}{2m}\mathbb{1}(x = x_m) \ , \end{aligned} \tag{9}$$

where

$$w_{l-1,l}(x) = \frac{1}{x_l - x_{l-1}}\mathbb{1}(x \in [x_{l-1}, x_l]) \ . \tag{10}$$

If $X \sim w(x)$, where $w(x)$ is given by (9), then $X = x_1$ or $X = x_m$ with an atomic probability $\frac{1}{2m}$. Hence, the first value x_1 or the last one x_m from the sequence (8) are selected with equal probabilities. Otherwise, one of the intervals $[x_{l-1}, x_l]$, for $l = 1, \ldots, m - 1$, is designated with identical atomic probabilities $\frac{1}{m}$. When such a single interval is selected, we have $X \sim$

$w_{l-1,l}(x)$, so the output x is generated using the uniform density $U([x_{l-1}, x_l])$, which is described by (10).

Therefore, $w(x)$ can be seen as a certain generalization of the discrete distribution discussed in Section 3.1. The pdf $w(x)$ also generates values from the same interval $[x_1, x_m]$, but they are more diversified – apart from values directly equal to the ones from the sequence (8), all $x \in [x_1, x_m]$ can now be obtained.

Statistical characterizations of the density $w(x)$ are summarized in the following lemma:

Lemma 1. *Let $X \sim w(x)$, where $w(x)$ is a pdf described by* (9) *and* (10). *Then*

$$\mathbb{E}X = \frac{1}{m}\sum_{i=1}^{m} x_i = \bar{x} \ ,$$

and

$$\operatorname{Var} X = \frac{1}{m}\left(\frac{5}{6}x_1^2 + \frac{1}{3}x_1x_2 + \frac{2}{3}x_2^2 + \frac{1}{3}x_2x_3 + \ldots + \frac{1}{3}x_{m-1}x_m + \frac{5}{6}x_m^2\right) - (\bar{x})^2 = s_w^2 \ .$$

Proof. From (9) and (10), we have

$$\mathbb{E}X = \frac{1}{2m}x_1 + \frac{1}{2m}x_m + \frac{1}{m}\sum_{i=1}^{m-1}\frac{x_i + x_{i+1}}{2}$$

and

$$\operatorname{Var} X = \frac{1}{2m}x_1^2 + \frac{1}{2m}x_m^2 + \frac{1}{m}\sum_{i=1}^{m-1}\frac{x_i^2 + x_i x_{i+1}^2 + x_{i+1}^2}{3} - (\bar{x})^2 \ ,$$

which concludes the proof. □

From Lemma 1 we see that if $X \sim w(x)$, then the expected value of X is precisely equal to its mean $\bar{x}$. But the variance s_w^2 of X is not equal to a classical estimator, i.e. the standard sample variance s^2. A difference between the variances s_w^2 and s^2 can be important for the intended diversity of the LRFNs in the second sample $\mathcal{B}$. We have

$$s_w^2 - s^2 = \frac{1}{m}\left(-\frac{1}{6}x_1^2 - \frac{1}{6}x_m^2 + \frac{1}{3}x_1 x_2 + \frac{1}{2}\sum_{i=2}^{m-1} x_i\left(x_{i+1} - x_i\right)\right) ,$$

which leads to the following remark:

Remark 1. *If $m \to \infty$ and $x_i > 0$, then $s_w^2 - s^2 \geq 0$. Therefore, a diversity (measured by a variance) of $X \sim w(x)$ is not lesser than a diversity of $X \sim d(x)$, if only a size of a sample is large enough, and all $x_i > 0$.*

3.2.2 Generation procedure

Now, to generate a fuzzy number $\tilde{b}_j \in \mathcal{B}$, if $j = 1, \ldots, n$, instead of the discrete distribution $d(x)$, the previously introduced density $w(x)$ is used (see also Algorithm 3). However, an overall procedure of a construction of $\tilde{b}_j$ is similar to the previous case, which is described in Section 3.1.

During the first step, a value of a core $b_j(1) = C$ is drawn using the distribution $w(x)$ based on the elements from the set $\mathcal{C}(1)$, so $C \sim w(x)$, where $x \in \mathcal{C}(1)$. Next, consecutive α-cuts of the given $\tilde{b}_j$ are calculated, starting from the value α_{k-1} and ending at $\alpha_0 = 0$. For each α_i, a value of a left end of $\tilde{b}_j(\alpha_i)$ is equal to (6), where $S^L(\alpha_i)$ is an independently drawn random value from the set $\mathcal{S}^L(\alpha_i)$, using the distribution $w(x)$ for the set $\mathcal{S}^L(\alpha_i)$. In the same way, a right end of $\tilde{b}_j(\alpha_i)$ is given by (7), where $S^R(\alpha_i)$ is independently drawn from the set $\mathcal{S}^R(\alpha_i)$, using the respective distribution $w(x)$ for this set.

Let us continue our example using the method of generation described in this subsection. We will show the construction of only one element of this sample. The remaining elements are constructed in the same way.

Example 1 (Continued). Let us start from the generation of the core of a new fuzzy number. According to the density function $w(x)$ defined by (9) there are four possibilities for choosing this value: take 1 with probability 1/6, take a randomly chosen (using the uniform distribution) number from the interval $[1, 2.5]$ with probability 1/3, take a randomly chosen number from the interval $[2.5, 3.5]$ with probability 1/3, take 3.5 with probability 1/6, (see also Figure 1). Suppose that the second option has been chosen, and a new core has been set to $b_1(1) = 1.75$. Now, consider the left and the right spreads for $\alpha = 0.5$. For choosing the value of the left incremental spread there are also 4 possibilities: take 0.5 with probability 1/6, take a randomly chosen number from the interval $[0.5, 0.75]$ with probability 1/3, take a randomly chosen number from the interval $[0.75, 1.25]$ with probability 1/3, take 1.25 with probability 1/6. Suppose that the first option has been chosen, and a new left incremental spread has been set to $S_1^L(0.5) = 0.5$. Similarly, for choosing the value of the right incremental spread there are 4 possibilities: take 0.75 with probability 1/6, take a randomly chosen number from the interval $[0.75, 1]$ with probability 1/3, take a randomly chosen number from the interval $[1, 1.25]$ with probability 1/3, take 1.25 with probability 1/6. Suppose that the third option has been chosen, and a new right incremental spread has been set to $S_1^R(0.5) = 0.8$. Now, consider the left and the right spreads for $\alpha = 0$. For choosing the value of the left incremental spread there are 4 similar possibilities: take 0.5 with probability 1/6, take a randomly chosen number from the interval $[0.5, 0.75]$ with probability 1/3, take a randomly chosen number from the interval $[0.75, 1.25]$ with probability $1/3$, take 1.25 with probability $1/6$. Suppose that the second option has been chosen, and a new left incremental spread has been set to $S_1^L(0) = 0.6$. Similarly, for choosing the value of the right incremental spread there are 4 possibilities: take 0.75 with probability $1/6$, take a randomly chosen number from the interval $[0.75, 1]$ with probability $1/3$, take a randomly chosen number from the interval $[1, 1.25]$ with probability $1/3$, take 1.25 with probability $1/6$. Suppose that the fourth option has been chosen, and a new right incremental spread has been set to $S_1^R(0) = 1.25$. Finally, the new generated element of the secondary sample is the fuzzy number defined by its core $b_1(1) = 1.75$, and two α-cuts defined by their respective limits: $b_1^L(0.5) = 1.25, b_1^R(0.5) = 2.55$, and $b_1^L(0) = 0.65, b_1^R(0) = 3.8$.

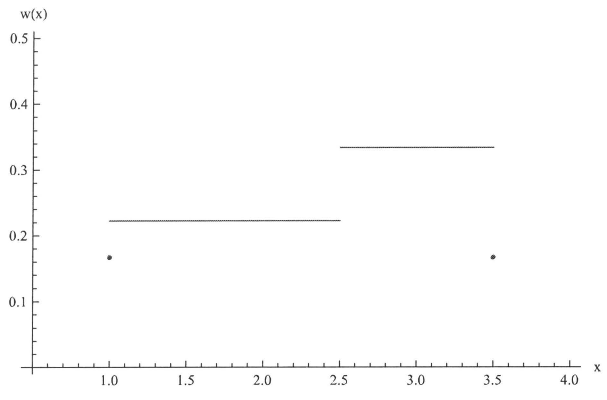

Figure 1: A plot of the density $w(x)$ for the set $\mathcal{C}(1)$ in Example 1

4 Properties of bootstrap-like secondary samples

After the introduction of the d-method and the w-method, we can numerically compare secondary samples, which are generated using these approaches. Moreover, we also apply a classical bootstrap in order to verify, if there are any significant differences among this widely used simulation method (see, e.g., [8, 9, 16, 21] for a more detailed discussion) and the algorithms proposed in this paper.

Let us start from a certain population $\mathcal{P}_{n_0}$, which consists of n_0 LRFNs. From this population, we randomly draw m elements. Let these elements constitute a primary sample $\mathcal{A}_m$. Afterwards, using the fuzzy numbers from this primary sample, three methods (i.e., the classical bootstrap, the d-method, and the w-method) are used to generate a secondary sample $\mathcal{B}_n$, which consists of n elements.

In our numerical experiments different experiment setting are used: a moderate population $\mathcal{P}_{100}$ (for which $n_0 = 100$) together with a small primary sample $\mathcal{A}_5$ (where $m = 5$) and a moderate secondary sample $\mathcal{B}_{100}$ (where $n = 100$), and a rather big population $\mathcal{P}_{200}$ with a moderate primary sample $\mathcal{A}_{100}$, and a rather big secondary sample $\mathcal{B}_{200}$. It allows us to compare outcomes for the classical bootstrap, the d-method, and the w-method, if preliminary information about a model (which is available only via a primary sample) is very sparse (in the case of $\mathcal{A}_5$) or relatively abundant (for $\mathcal{A}_{100}$).

For simplicity, only triangular fuzzy numbers will be considered, i.e., only two α-cuts (where $\alpha_0 = 0$ and $\alpha_1 = 1$) are used to construct the whole

LRFN. Of course, both the d-method and the w-method can be easily used to generate the second sample, even if more α-cuts are considered. However, the simplest types of LRFNs (like the triangular or the trapezoidal numbers) seem to be used very often in practical situations.

In the following numerical experiments, two types of triangular numbers are considered as a model for the population $\mathcal{P}_{n_0}$. The first one (which is further abbreviated as the "type 1 number") is a fuzzy number with an expected symmetrical spread, where the centre is random and has the standard normal distribution $N(0,1)$, and the semiwidths of the support are given as independent chi-square variables with 1 degree of freedom. A similar LRFN is discussed in a detailed way in [2]. The second kind (the "type 2 number") of a fuzzy number has a strictly non-symmetrical shape. In this case, the center points are described by the gamma distribution with a shape parameter 1 and a scale parameter 2, and the semiwidths of the support are drawn from independent exponential distributions with parameter 1 (for the left spread) or 2 (for the right spread).

We are interested in an analysis of mutual relations between a primary and a secondary sample for the different generation procedures and the mentioned types of LRFNs. Therefore, properties of both the primary sample and the generated, secondary set, are statistically summarized using a sample mean versus a population mean, which are calculated for the support and the center of a fuzzy number. From a statistical point of view, also diversity of the simulated fuzzy numbers is very important. Thus, a standard deviation is also found for the support and the center of LRFN in the case of the second (i.e., generated) sample.

Moreover, the simulated fuzzy numbers should give us some additional "insight" into the model, which is (generally) completely unknown and "hidden" in data from the primary sample. In an ideal situation, LRFNs from the second sample should be (in some way) "similar" to the numbers from the primary sample, but, simultaneously, not exactly "the same" as the elements from $\mathcal{A}_m$, and also "very close" to the population. Therefore, values of some classical measures (see Section 2) are evaluated for each possible pair of fuzzy numbers, which consists of one "old" LRFN (i.e., from $\mathcal{A}$), and one "new", generated number (i.e., an element from $\mathcal{B}$). The obtained measures are also summarized using common tools, like minimum, maximum, mean, and standard deviation. Afterwards, we can point out, if some generation method produces fuzzy numbers which are "the same as", "similar" or only "close" (and to what extent) to the LRFNs from the primary sample.

4.1 Small primary sample, type 1 fuzzy number

Based on the small sample $\mathcal{A}_5$ of type 1 of triangular fuzzy numbers, three moderate secondary samples $\mathcal{B}_{100}$ were generated, using the classical bootstrap, the d-method, and the w-method. Then, means for the core $\bar{X}_C^*$ (see Figure 2), the left end of the support $\bar{X}_L^*$ (see Figure 3) and the right end $\bar{X}_R^*$ (see Figure 4) for each of the simulated samples were calculated. From now on, obtained results for the bootstrap are marked with circles in the graphs, for the d-method – with diamonds, and for the w-method – with squares. Horizontal bold lines depict means of the primary sample $\mathcal{A}$ for the core $\bar{X}_C^{\mathcal{A}}$, for the left end of the support $\bar{X}_L^{\mathcal{A}}$ and for the right end $\bar{X}_R^{\mathcal{A}}$, and axes of respective graphs start exactly in the means of the population (for the core $\bar{X}_C$, for the left end of the support $\bar{X}_L$ and for the right end $\bar{X}_R$).

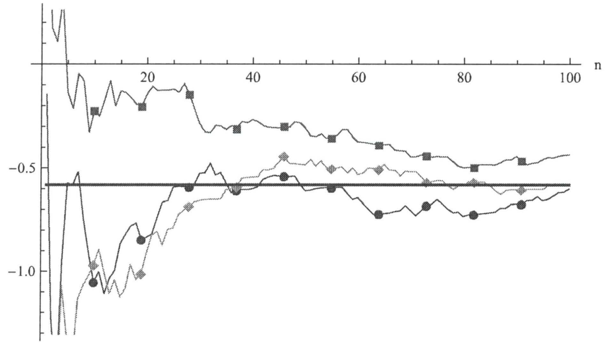

Figure 2: Small primary sample, type 1 fuzzy number: means of the core as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

As it is seen, each of the simulation method behaves generally well. In each case, after generation of 30 – 40 fuzzy numbers, a mean of the secondary sample $\bar{X}^*$ approaches a respective mean of the primary set $\bar{X}^{\mathcal{A}}$. Moreover, an application of the d-method or the w-method seem to have some advantages, if they are compared to the classical bootstrap. For example, the means for these two approaches are, in general, closer to $\bar{X}^{\mathcal{A}}$ (i.e., a mean of $\mathcal{A}_5$), than in the case of the bootstrap. The respective graphs are also much smoother. Surprisingly, in the case of the w-method, the respective mean is also closer to "the real" result – a mean of our unknown model, i.e., the population $\mathcal{P}_{100}$.

Apart from the comparison of the means, a diversity of the generated LRFNs should be also considered. Hence, standard deviations for the core (see Figure 5), for the left end of the support (see Figure 6) and for its right

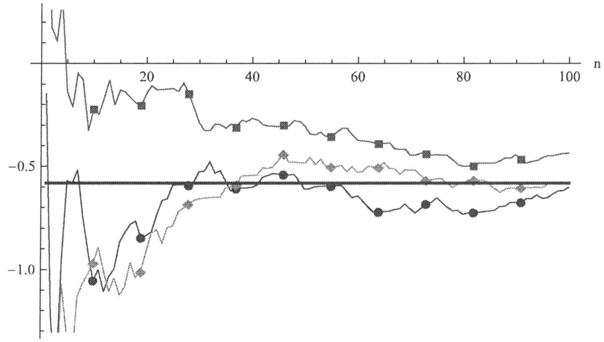

Figure 3: Small primary sample, type 1 fuzzy number: means of the left end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

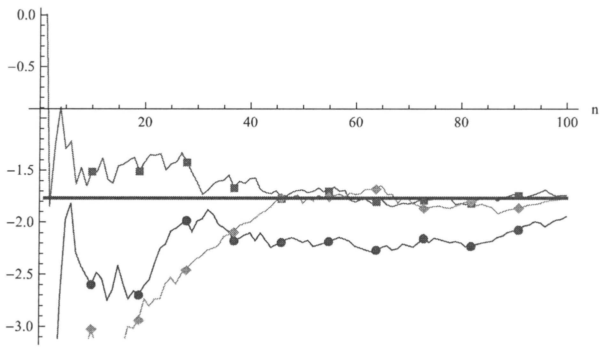

Figure 4: Small primary sample, type 1 fuzzy number: means of the right end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

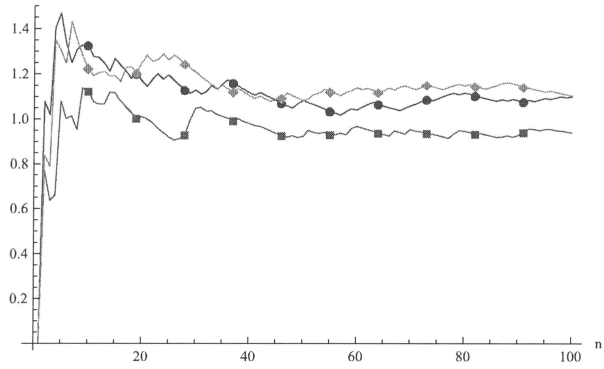

Figure 5: Small primary sample, type 1 fuzzy number: standard deviations of the core as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

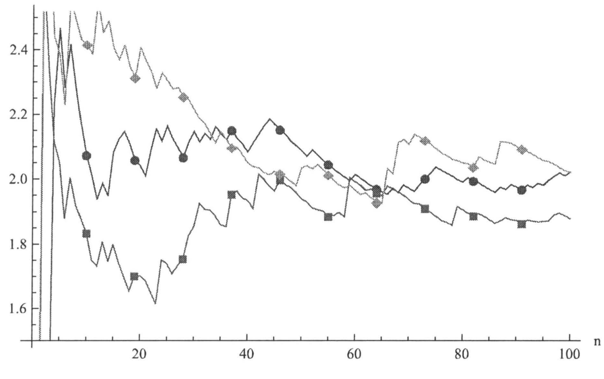

Figure 6: Small primary sample, type 1 fuzzy number: standard deviations of the left end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

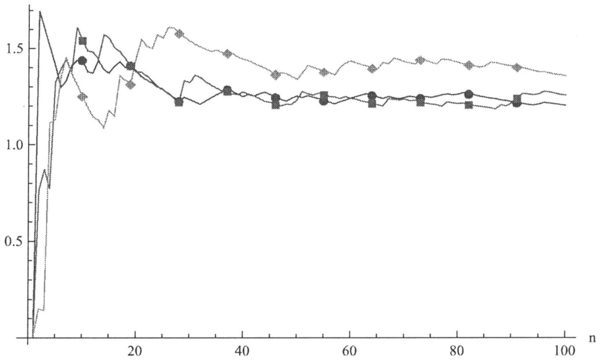

Figure 7: Small primary sample, type 1 fuzzy number: standard deviations of the right end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

end (see Figure 7) are plotted. These graphs are marked in the same way as previous ones. In the each of these cases, a standard deviation of the secondary sample is the lowest, if the w-method is used.

Now, we compare the three secondary samples, which are generated using the considered simulation procedures, but with a help of the measures, which were recalled in Section 2. Let us assume that $l(\tilde{a}_i, \tilde{b}_j)$ is a value of some measure $l(.,.)$ of similarity between LRFNs $\tilde{a}_i \in \mathcal{A}$ and $\tilde{b}_j \in \mathcal{B}$. Then, the following notation is used

$$\mathrm{MinMin} = \min_j\{\min_i l(\tilde{a}_i, \tilde{b}_j)\}, \mathrm{MinMax} = \min_j\{\max_i l(\tilde{a}_i, \tilde{b}_j)\},$$

$$\mathrm{MaxMin} = \max_j\{\min_i l(\tilde{a}_i, \tilde{b}_j)\}, \mathrm{MaxMax} = \max_j\{\max_i l(\tilde{a}_i, \tilde{b}_j)\},$$

$$\mathrm{MeanMin} = \frac{1}{n}\sum_{j=1}^{n} \min_i l(\tilde{a}_i, \tilde{b}_j), \mathrm{MeanMax} = \frac{1}{n}\sum_{j=1}^{n} \max_i l(\tilde{a}_i, \tilde{b}_j),$$

$$\mathrm{StDevMin} = \frac{1}{n}\sum_{j=1}^{n} \left(\min_i l(\tilde{a}_i, \tilde{b}_j) - \mathrm{MeanMin}\right)^2,$$

$$\mathrm{StDevMax} = \frac{1}{n}\sum_{j=1}^{n} \left(\min_i l(\tilde{a}_i, \tilde{b}_j) - \mathrm{MeanMax}\right)^2.$$

The respective measures of similarity are summarized in Table 1 (if $\mathcal{B}_{100}$ is simulated using the bootstrap approach), Table 2 (in a case of the d-method) and Table 3 (for the w-method). Of course, the bootstrap only repeats fuzzy numbers, which are already present in the primary sample. Therefore, MinMin and MinMax values for the measures m_{l_1}, m_∞ and m_H are strictly equal to zero. But in a case of the d-method, even the values of these measures are more diversified, so we have StDevMin > 0. The same applies for the w-method. Therefore, these two approaches produce LRFNs, which are more diversified ("not exactly the same" in some way) than the numbers from $\mathcal{A}_5$. However, the generated LRFNs are also "similar" (in a sense of the applied measures) to the fuzzy numbers from the primary sample, because the obtained MinMin and MeanMin values are very close to zero. It seems that using the w-method is more fruitful than the d-method, because MinMax, MaxMax and MeanMax values are generally lesser for this first approach, and MeanMin values are very similar. Hence, even LRFNs, which are "maximally" distant from the fuzzy numbers from the primary sample, are "closer" in the case of the w-method than for the d-method.

Let us analyze the noticed similarity in another way. In order to do this, an additional independent sample $\mathcal{T}_{200}$ which consists of 200 fuzzy numbers of type 1 was generated. Then, three secondary sets $\mathcal{B}_{200}$ are sampled based on $\mathcal{A}_5$, using the bootstrap, the d-method, and the w-method. We find a LRFN from each of $\mathcal{B}_{200}$, which is the nearest to some fuzzy number from $\mathcal{T}_{200}$ in a sense of one of the measures $m_{l_1}, m_{\infty}, m_{TD}, m_H$, i.e. a value

$$\text{MinMin} = \min_j \{\min_i l(\tilde{t}_i, \tilde{b}_j)\} ,$$

where $\tilde{t}_i \in \mathcal{T}_{200}$ and $\tilde{b}_j \in \mathcal{B}_{200}$, is calculated. The obtained minimal values of these measures for respective pairs of LRFNs are given in Table 4, and for each of the measure its minimum is emboldened. As it is seen, if the w-method is used, then a fuzzy number is generated, which is the most similar to some element from $\mathcal{T}_{200}$. In some way, this new independent sample $\mathcal{T}_{200}$ gives an additional insight into the "true model", because it is a supplementary sample from the unknown source, which models our LRFNs. Therefore, the w-method produces fuzzy numbers, which are the nearest to this model in the considered case. Note, that because the bootstrap only repeats elements from the primary sample, then for this method the obtained values of the measures are even $6-7$ times greater than for the best match.

4.2 Small primary sample, type 2 fuzzy number

Now we analyze the three considered simulation procedures, if the small primary sample $\mathcal{A}_5$ consists of the strictly non-symmetrical triangular fuzzy numbers (i.e., the previously mentioned LRFNs of "type 2"). Graphs of the means (for the core – see Figure 8, for the left end of the support – see Figure 9, for the right end of the support – see Figure 10) are very similar to the case, which was described in Section 4.1. Once again, these means for the d-method and the w-method are, in general, closer to the respective means of the primary sample, than in the case of the bootstrap. Their graphs are also very smooth. Moreover, the plots of the standard deviations behave reasonably well (for the core – see Figure 11, for the left end of the support – see Figure 12, for the right end of the support – see Figure 13). The obtained values are the lowest, if the w-method is used.

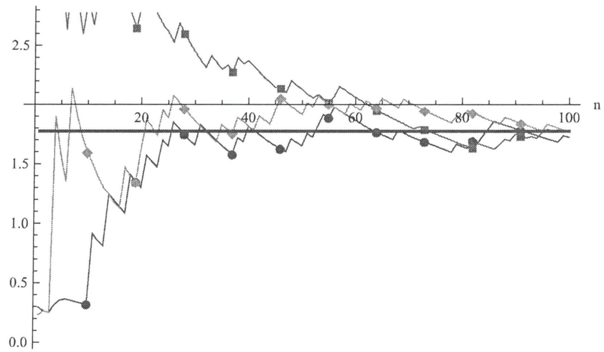

Figure 8: Small primary sample, type 2 fuzzy number: means of the core as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

Also the characteristics of the similarity measures, which were introduced in Section 4.1, can be found in this case (see Table 5, Table 6 and Table 7 for the respective summaries for the different simulation approaches). In general, the conclusions are similar as in the case of the type 1 fuzzy numbers, i.e., the bootstrap only repeats LRFNs from the primary sample, and the d-method and the w-method produce more diversified output, which is still similar (in the sense of the considered measures) to the values from $\mathcal{A}_5$. But a decision,

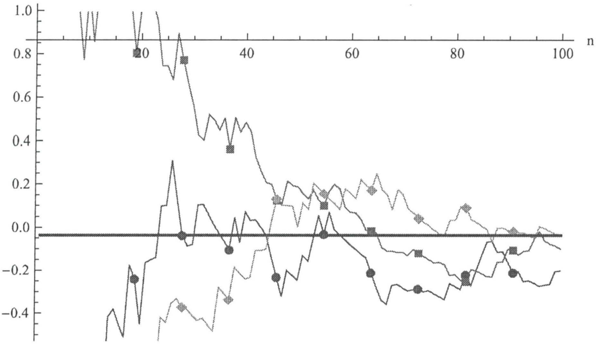

Figure 9: Small primary sample, type 2 fuzzy number: means of the left end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

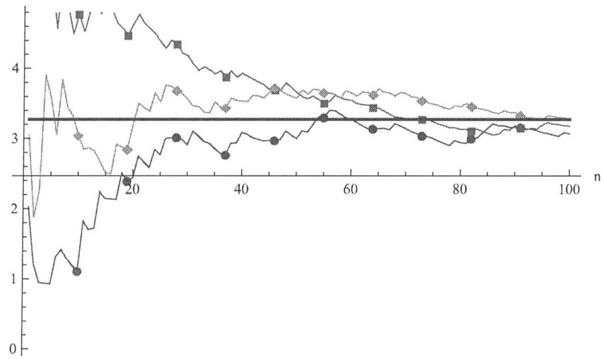

Figure 10: Small primary sample, type 2 fuzzy number: means of the right end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

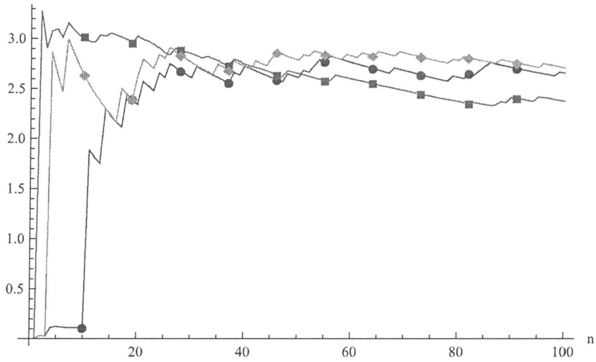

Figure 11: Small primary sample, type 2 fuzzy number: standard deviations of the core as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

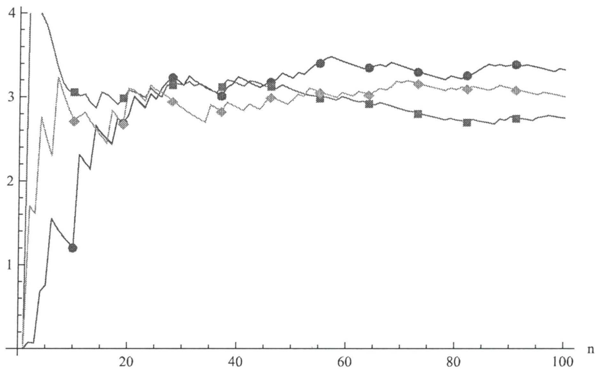

Figure 12: Small primary sample, type 2 fuzzy number: standard deviations of the left end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

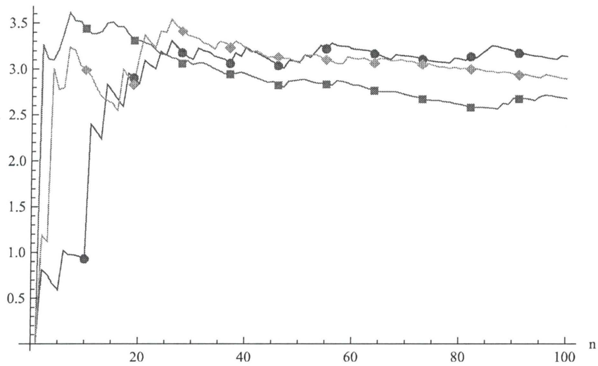

Figure 13: Small primary sample, type 2 fuzzy number: standard deviations of the right end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

if the d-method or the w-method is better suited, when a "maximum" distant criterion is taken into account, is not so straightforward now. As it is seen, MinMax values are lower for the d-method, but MaxMax and MeanMax are lower in the case of the w-method.

And once again, we analyze the supplementary, independent sample $\mathcal{T}_{200}$ of LRFNs of type 2. The fuzzy numbers from this set are compared with three samples $\mathcal{B}_{200}$ which were generated using the classical bootstrap and the two methods introduced in this paper. As in Section 4.1, LRFNs from $\mathcal{T}_{200}$ and each of $\mathcal{B}_{200}$ are compared in order to find pairs of fuzzy numbers, which are the most similar. The obtained minimal values of measures can be found in Table 8. Also in this case, the w-method generates fuzzy numbers which are the nearest to some element from the set $\mathcal{T}_{200}$, apart from the measure m_∞, for which the d-method gives the best result. The classical bootstrap gives values, which are even 2 – 3 times greater than the best matches.

4.3 Moderate primary sample

In practical situations, apart from small statistical samples, which consist of only a few values, larger samples are also used. Therefore, we also analyze the behavior of a moderate primary sample, for which $m = 100$ (i.e. $\mathcal{A}_{100}$), and a respectively simulated secondary sample $\mathcal{B}_{200}$, which is rather a big one, especially comparing to the previous examples (now we have $n = 200$). As it turns out, general conclusions for both type 1 and type 2 of LRFNs are very similar to the outcomes for the small sample, which were summarized in Section 4.1 and Section 4.2. Hence, we may omit a more detailed discussion, in order to present other, but in some way, supplementary approach.

Up till now, we have discussed a convergence speed of the mean of the secondary sample $\bar{X}^*$ to the "true" (but, in general, unknown) mean of the population $\bar{X}$. And, in our reasoning, three "focal points" (a core, a left and a right end of a support) have been taken into account. In [2] the authors consider an application of LIL (the law of iterated logarithm) as a tool for a convergence diagnosis for the simulated fuzzy numbers. Therefore, we will also analyze the behavior of a distance between $\sqrt{n}/\sqrt{2n \log\log n}\bar{X}^*$ and $\bar{X}$ as a function of the secondary sample size n. To keep consistency with our previous analysis, the three mentioned "focal points" will be still in the centre of our attention. Hence, a distance for a core

$$\frac{\sqrt{n}}{\sqrt{2n \log\log n}} |\bar{X}^*_C - \bar{X}| \; , \tag{11}$$

and similarly defined measures for a left and a right end of a support, will be further used, instead of the supremum distance, which is considered in [2].

Because the secondary sample $\mathcal{B}_{200}$ is rather big, the convergence speed for (11) and rest of the similar measures, as functions of n, is now more visible. We restrict our analysis only to type 1 of fuzzy numbers, but the obtained conclusions are also similar for the type 2. The calculated distances as functions of the secondary sample size are plotted in Figure 14 (the core), in Figure 15 (the left end of the support) and in Figure 15 (the right end of the support). As it is seen, the bootstrap approach is the worst one, especially for larger values of n, because the obtained distances are, in general, more distant from zero for this simulation method. Both the d-method and the w-method produce the relatively well behaving output.

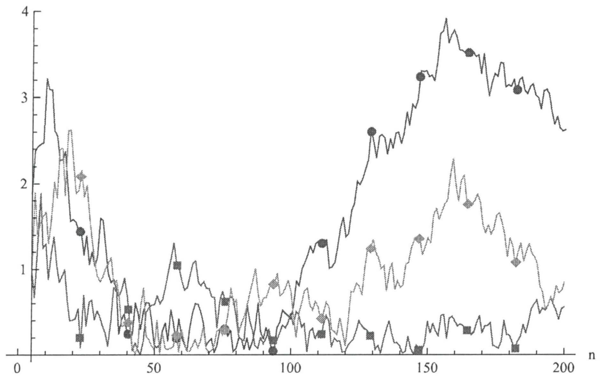

Figure 14: Moderate primary sample, type 1 fuzzy number: LIL distances for the core as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

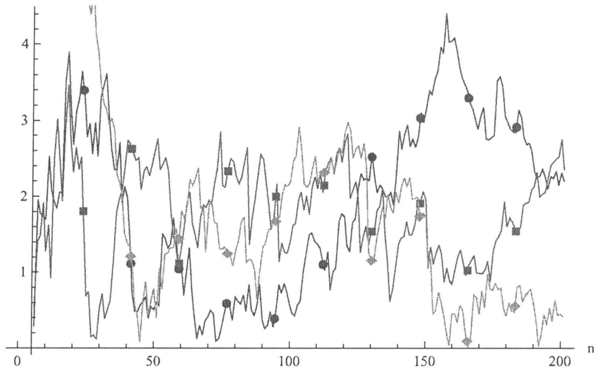

Figure 15: Moderate primary sample, type 1 fuzzy number: LIL distances for the left end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

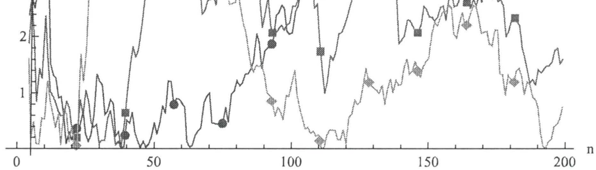

Figure 16: Moderate primary sample, type 1 fuzzy number: LIL distances for the right end of the support as functions of the secondary sample size n (the bootstrap – circles, the d-method – diamonds, the w-method – squares)

5 New bootstrap-like sample as a tool in statistical tests

Apart from the statistical properties of the simulated LRFNs, a possibility of an application of the d-method and the w-method in practical statistical cases is also investigated. Two types of tests for the expected value of the fuzzy numbers are then considered (see Section 2.3 for additional details and respective notation) as an appropriate example.

The first one is a bootstrapped version of the test proposed in [12] (see Corollary 1). From now on, it will be called as the K-test (from its author's name) for the expected value. The second test is the procedure developed in [8, 16] (see Corollary 2). It will be called as the GRMCG-test for the expected value (also based on the authors' names). In this case, we apply a standard uniform density as the weight normalized measure φ in $D_W^{\varphi}(\tilde{a}, \tilde{b})$ metric (2) (see [1, 16] for additional details and other approaches).

As an initial sample in each of these tests, three types of triangular fuzzy numbers are simulated. Type 1 and type 2 are described in Section 4. Type 3, which was considered in [12], is a fuzzy number, where the random center has the standard normal distribution $N(0, 1)$, and the spreads of the support are independently drawn from the standard uniform distribution $U([0, 1])$.

For each of these types of the fuzzy numbers, three different simulation procedures (the classical bootstrap, the d-method and the w-method) are used to generate an input random sample for the test. A number of the elements in such a sample n is varying, so both small and medium sample sizes are considered, i.e., we set $n = 5, 10, 30, 100$. Also a few values of the number of bootstrap replications r (namely $r = 100, 200, 1000$) are used to generate a respective bootstrapped distribution of the test statistics, in order to investigate a possible influence of this parameter. In each of these experiments, the whole resampling procedure is iterated 100000 times (see, e.g., [6, 8, 16, 21] for additional details of such an approach).

Based on the respective statistics in each of the tests of the expected value, an empirical percentage of rejections $\hat{p}$ at the nominal significance level $p = 0.05$ for the true null hypothesis is then computed. This estimated value is widely used as a benchmarking tool for the bootstrapped version of the statistical tests (see, e.g., [6, 8, 16, 20, 21]). Then, three considered simulations procedures can be directly compared.

In general, the simulated values of $\hat{p}$ for all of the approaches are very close to one another, and the overall properties are very similar. Especially, the empirical percentages of rejections converge to one another for larger values of n and r (like $n = 100$ and $r = 1000$). However, there are also some significant differences. In order to emphasize them, in each of the experiment a value of $\hat{p}$, which is nearest to the true value of significance level of p, is emboldened.

Let us start from the K-test of the expected value. As it is seen for the fuzzy numbers of type 1 (see Table 9), type 2 (see Table 10) and type 3 (see Table 11), a comparison of the simulation approaches seems to be quite simple. In each of these cases, the d-method leads to $\hat{p}$, which is nearest to the assumed significance level p, apart from a few exceptions. For all of them the classical bootstrap approach gives the most "true" answer. But even for these exceptions, differences between the empirical percentages of rejections for the d-method and the classical bootstrap are not very significant (about 0.001–0.002). And these differences favor the d-method especially for smaller values of n and r. Moreover, in overall, the classical bootstrap occupies a second place in measuring a proximity between $\hat{p}$ and p.

For the GRMCG-test, an analysis of the differences between $\hat{p}$ and p is not so straightforward. In the case of type 1 of the fuzzy numbers (see Table 12), $\hat{p}$ seems to be nearest to the true significance level for the w-method (when a smaller number of the elements in the initial sample is taken into account, i.e. $n = 5, 10$) or for the d-method (for larger values $n = 30, 100$). Especially for the small samples, the classical bootstrap approach gives the worst answers and differences in the estimation between the bootstrap and one of the other approaches are quite important (about 0.008–0.01).

However, when type 2 of the fuzzy numbers is analyzed (see Table 13), the picture is not so clear. Firstly, for $n = 5, 10$, the estimated percentages of rejections favor the classical bootstrap approach, and the other approaches give larger values of $\hat{p}$. In these cases, the differences between the application of the classical bootstrap and other simulation methods are quite clear (even equal to 0.012–0.015). Secondly, for $n = 30, 100$, the outputs are more accurate, if the d-method or the w-method are used. Then, the differences among various simulated $\hat{p}$ are quite small (about 0.001–0.002).

In the case of type 3 of the fuzzy numbers (see Table 14), it seems that, in overall, the d-method or the w-method produce the most accurate estimators of $\hat{p}$. It can be seen especially for the smaller samples ($n = 5, 10$), for which the classical bootstrap approach gives an estimator of the rejection rate is about 0.004 smaller than for the other methods. For the largest sample

($n = 100$), the d-method is favored, but once again, the differences among various simulated values of $\hat{p}$ are quite small.

Taking into account the whole analysis, it is not possible to point out the undoubtedly best simulation procedure, which gives the most accurate values of $\hat{p}$. However, the application of the d-method and the w-method looks promising, especially for smaller initial samples.

6 Conclusions

In this paper we propose two simulation algorithms for the generation of left-right fuzzy numbers, i.e., the d-method and the w-method. Both of these approaches, based on the resampling paradigm, utilize the primary sample of fuzzy numbers in order to randomly generate the secondary bootstrap-like sample. This generation is based on α-cuts of LRFNs, and a strictly non-parametric approach, without necessity of taking additional assumptions about a source (or a model) of the primary sample. Then, we have compared numerically the outputs generated using the classical bootstrap with the outputs generated by our two (the d-method, and the w-method) proposed methods. In the examples of such comparisons, which are presented in this paper, both the small and the moderate samples of the two types of LRFNs have been analyzed. Moreover, the similar simulated samples for three types of LRFNs are compared in the bootstrapped versions of two tests of the expected value. We have shown that the properties of the introduced methods are very promising. Both our new methods, the d-method and the w-method, produce LRFNs, which are "similar" to the elements of the original population and the primary sample, but also "more variable" in comparison to the case of the classical bootstrap. Moreover, in some cases of the considered tests of the expected value, our benchmarking tool (a difference between the nominal significance level of the test and the empirical percentage of rejections of the true null hypothesis) indicates the supremacy of the d-method and the w-method over the classical bootstrap. The proposed methods have, in comparison to the classical bootstrap, one disadvantage, when considered fuzzy numbers have their "natural" limits (e.g., when their supports must contain only non-negative numbers). In such a case it may happen that some generated elements of a secondary (bootstrap-like) sample may not fulfil such requirements. One can introduce certain modifications (e.g., a simple curtailment) in order to make them reasonable, but the consequences of such modifications require consideration in the future research.

Compliance with Ethical Standards

All authors of this paper declare, that they have no conflict of interest. This article does not contain any studies with human participants or animals performed by any of the authors.

References

[1] Bertoluzza C., Corral N., Salas A. On a new class of distances between fuzzy numbers. Mathware & Soft Computing. 1995; 2: 71–84.

[2] Colubi A., Fernández-García C., Gil M. A. Simulation of Random Fuzzy Variables: An Empirical Approach to Statistical/Probabilistic Studies

With Fuzzy Experimental Data. IEEE Transactions on Fuzzy Systems. 2002; 10(3): 384–390.

[3] Couso I., Dubois D., Sánchez L. Random sets and random fuzzy sets as ill-perceived random variables. Springer briefs in applied sciences and technology. Springer, Cham, 2014

[4] Gil M.A., Hryniewicz O. Statistics with Imprecise Data. In: Meyers, R.E. (ed.) Encyclopedia of Complexity and Systems Science. Springer, Heidelberg, 2009: 8679–8690.

[5] Gil M.A., López-Díaz M., Ralescu D. A. Overview on the Development of Fuzzy Random Variables. Fuzzy Sets and Systems. 2006; 157(19): 2546–2557.

[6] Gil M. A., Montenegro M., González-Rodríguez G., Colubi A., Casals M. R. Bootstrap approach to the multi-sample test of means with imprecise data. Computational statistics & data analysis. 2006; 51: 148–162.

[7] González-Rodríguez G., Colubi A., Trutschnig W. Simulation of fuzzy random variables. Information Sciences. 2009; 179: 642–653.

[8] González-Rodríguez G., Montenegro M., Colubi A., Gil M. A. Bootstrap techniques and fuzzy random variables: Synergy in hypothesis testing with fuzzy data. Fuzzy sets and systems. 2006; 157: 2608–2613.

[9] Hung W.-L. Weighted bootstrap method for fuzzy data. Soft Computing. 2006; 10: 140–143.

[10] Hryniewicz O. Comparison of Fuzzy and Crisp Random Variables by Monte Carlo simulations. In: Grzegorzewski P, Gągolewski M,

Hryniewicz O, Gil M A. (eds.). Strengthening Links Between Data Analysis and Soft Computing. Springer, 2015.

[11] Hryniewicz O., Kaczmarek K., Nowak P. Bayes statistical decisions with random fuzzy data – An application for the Weibull distribution. Eksploatacja i Niezawodnosc – Maintenance and Reliability. 2015; 17(4): 610–616.

[12] Körner R. An asymptotic α-test for the expectation of random fuzzy variables. Journal of statistical planning and inference. 2000; 83: 331–346.

[13] Körner R., Näther W. On the variance of random fuzzy variables. In: Bertoluzza C., Gil M. A., Ralescu D. A. (eds.) Statistical Modeling, Analysis and Management of Fuzzy Data. Physica-Verlag, 2001; 22–39.

[14] Kwakernaak H. Fuzzy Random Variables, Part I: Definitions and Theorems. Information Sciences, 1978; 15:1–29.

[15] Kwakernaak H. Fuzzy Random Variables, Part II: Algorithms and Examples for the Discrete Case, Information Sciences. 1979; 17:253–278.

[16] Montenegro M., Colubi A., Casals M. R., Gil M. A. Asymptotic and Bootstrap techniques for testing the expected value of a fuzzy random variable. Metrika. 2004; 59: 31–49.

[17] Nowak P., Romaniuk M. A fuzzy approach to option pricing in a Levy process setting. International Journal of Applied Mathematics and Computer Science. 2013; 23(3): 613–622.

[18] Nowak P., Romaniuk M. Catastrophe bond pricing for the two–factor Vasicek interest rate model with automatized fuzzy decision making. Soft Computing. 2017; 21(10): 2575–2597.

[19] Puri M.L., Ralescu D.A., Fuzzy Random Variables. Journ. of Math. Anal. and Appl. 1986; 114:409-422.

[20] Ramos-Guajardo A. B., Colubi A., González-Rodríguez G., Gil M. A. One-sample tests for a generalized Fréchet variance of a fuzzy random variable. Metrika. 2010; 71: 185–202.

[21] Ramos-Guajardo A. B., Lubiano M. A. K-sample tests for equality of variances of random fuzzy sets. Computational Statistics and Data Analysis. 2012; 56: 956–966.

[22] Ralescu A. L., Ralescu D. A.. Probability and fuzziness. In for. Sci. 1984; 34: 85–92.

[23] Robert Ch. P., Casella G. Monte Carlo Statistical Methods (2nd ed.). Springer, New York, 2004.

[24] Romaniuk M. On simulation of maintenance costs for water distribution system with fuzzy parameters. Eksploatacja i Niezawodnosc – Maintenance and Reliability. 2016; 18(4): 514–527.

[25] Tran L., Duckstein L. Comparison of fuzzy numbers using a fuzzy distance measure. Fuzzy Sets and Systems. 2002; 130: 331–341.

[26] Zwick R., Carlstein E., Budescu D. V. Measures of similarity among fuzzy concepts: a comparative analysis. International Journal of Approximate Reasoning. 1987; 1: 221–242.

Algorithm 2: Secondary sample generation for the $d(x)$-method

Input: Sets of the cores and the incremental spreads for $\mathcal{A}$, a number of LRFNs in a secondary sample n, a number of possible α-cuts $k+1$.

Output: A secondary sample $\mathcal{B}$ generated using the discrete distribution $d(x)$.

for $j \leftarrow 1$ **to** n **do**

Randomly draw a value of a core $b_j(1)$ from the set $\mathcal{C}(1)$, using a discrete uniform distribution for m elements;

for $i \leftarrow k-1$ **to** 0 **do**

Randomly draw a value of a left incremental spread $S^L(\alpha_i)$ from the set $\mathcal{S}^L(\alpha_i)$, using a discrete uniform distribution for m elements;

Find a left end of the α_i-cut $b_j^L(\alpha_i) = b_j^L(\alpha_{i+1}) - S^L(\alpha_i)$;

Randomly draw a value of a right incremental spread $S^R(\alpha_i)$ from the set $\mathcal{S}^R(\alpha_i)$, using a discrete uniform distribution for m elements;

Find a right end of the α_i-cut $b_j^R(\alpha_i) = b_j^R(\alpha_{i+1}) + S^R(\alpha_i)$;

end

Construct $\tilde{b}_j$ from the obtained α-cuts and append it to $\mathcal{B}$;

end

Algorithm 3: Secondary sample generation using the $w(x)$-method

Input: Sets of the cores and the incremental spreads for $\mathcal{A}$, a number of LRFNs in a secondary sample n, a number of possible α-cuts $k+1$.

Output: A secondary sample $\mathcal{B}$ generated using the distribution $w(x)$.

for $j \leftarrow 1$ **to** n **do**

Randomly draw a value of a core $b_j(1)$ *from the set* $\mathcal{C}(1)$*, using the density* (9) *for* m *elements;*

for $i \leftarrow k-1$ **to** 0 **do**

Randomly draw a value of a left incremental spread $S^L(\alpha_i)$ *from the set* $\mathcal{S}^L(\alpha_i)$*, using the density* (9) *for* m *elements;*

Find a left end of the α_i*-cut* $b_j^L(\alpha_i) = b_j^L(\alpha_{i+1}) - S^L(\alpha_i)$;

Randomly draw a value of a right incremental spread $S^R(\alpha_i)$ *from the set* $\mathcal{S}^R(\alpha_i)$*, using the density* (9) *for* m *elements;*

Find a right end of the α_i*-cut* $b_j^R(\alpha_i) = b_j^R(\alpha_{i+1}) + S^R(\alpha_i)$;

end

Construct $\tilde{b}_j$ *from the obtained* α*-cuts and append it to* $\mathcal{B}$;

end

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0	0	0.000736245	0
MaxMin	0	0	1.53948	0
MeanMin	0	0	0.019812	0
StDevMin	0	0	0.154899	0
MinMax	3.5179	1	4.99107	3.58809
MaxMax	3.5179	1	11.012	3.73185
MeanMax	3.5179	1	10.9518	3.73042
StDevMax	0	0	0.599078	0.0143039

Table 1: Small primary sample, type 1 fuzzy number: measures for the bootstrap

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0	0	0.000736245	0
MaxMin	2.62947	0.801501	2.13371	2.0113
MeanMin	0.0980749	0.210644	0.0417063	0.0697916
StDevMin	0.271452	0.219908	0.211372	0.204192
MinMax	4.12642	1	17.5908	5.74315
MaxMax	4.41593	1	21.6322	5.74315
MeanMax	4.22721	1	19.0016	5.74315
StDevMax	0.137404	0	1.9228	0

Table 2: Small primary sample, type 1 fuzzy number: measures for the d-method

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0.0100468	0.211763	0.000652598	0.0100467
MaxMin	1.02656	0.955989	0.415887	0.97555
MeanMin	0.0880436	0.223415	0.022968	0.0770365
StDevMin	0.166473	0.0737455	0.0723465	0.143194
MinMax	4.07298	1	14.4999	3.85217
MaxMax	4.07298	1	19.3633	5.74315
MeanMax	4.07298	1	16.7118	4.84823
StDevMax	0	0	2.26648	0.804912

Table 3: Small primary sample, type 1 fuzzy number: measures for the w-method

	m_{l_1}	m_∞	m_{TD}	m_H
Bootstrap	0.0415538	0.310524	0.00157894	0.0262274
d-method	0.0186548	0.0918212	0.000502853	0.0156908
w-method	**0.00643799**	**0.0626196**	**0.000242158**	**0.00595363**

Table 4: Small primary sample, type 1 fuzzy number: minimal measures for the comparisons with the independent sample $\mathcal{T}_{200}$

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0	0	0.0324768	0
MaxMin	0	0	0.417745	0
MeanMin	0	0	0.0391855	0
StDevMin	0	0	0.0429783	0
MinMax	3.40761	1	46.4003	6.82781
MaxMax	5.52281	1	53.2759	8.27309
MeanMax	5.41894	1	53.1464	8.22917
StDevMax	0.452897	0	0.740394	0.201653

Table 5: Small primary sample, type 2 fuzzy number: measures for the bootstrap

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0	0	0.0324768	0
MaxMin	0.364996	0.0654479	1.8041	0.184694
MeanMin	0.0217277	0.00916271	0.122545	0.017279
StDevMin	0.0616384	0.0227096	0.182615	0.0449755
MinMax	5.52281	1	55.9515	8.45778
MaxMax	7.54289	1	65.602	8.7359
MeanMax	7.41111	1	61.854	8.68305
StDevMax	0.367233	0	3.03121	0.109105

Table 6: Small primary sample, type 2 fuzzy number: measures for the d-method

	m_{l_1}	m_∞	m_{TD}	m_H
MinMin	0.0481819	0.0333	0.0625318	0.0275902
MaxMin	0.725874	0.494869	0.804754	0.40568
MeanMin	0.0652436	0.0415227	0.124169	0.0388207
StDevMin	0.0866392	0.0499653	0.143116	0.0592306
MinMax	5.60831	1	58.8443	8.59327
MaxMax	7.46832	1	63.1713	8.7359
MeanMax	7.35736	1	61.6798	8.69881
StDevMax	0.439196	0	2.00388	0.0625592

Table 7: Small primary sample, type 2 fuzzy number: measures for the w-method

	m_{l_1}	m_∞	m_{TD}	m_H
Bootstrap	0.0649436	0.202817	0.0260107	0.0371561
d-method	0.0587924	**0.0782876**	0.0109019	0.0305752
w-method	**0.051629**	0.098061	**0.00938974**	**0.0302706**

Table 8: Small primary sample, type 2 fuzzy number: minimal measures for the comparison with the independent sample $\mathcal{T}_{200}$

n	5	10	30	100
r	100			
Bootstrap	0.16024	0.10113	0.07006	0.06263
d-method	**0.14617**	**0.09448**	**0.067571**	**0.06127**
w-method	0.1668	0.10129	0.07117	0.06283
r	200			
Bootstrap	0.15438	0.09583	0.06558	0.05714
d-method	**0.1378**	**0.08804**	**0.06363**	**0.05636**
w-method	0.16153	0.09723	0.06464	0.0571
r	1000			
Bootstrap	0.14834	0.08961	0.06109	0.05449
d-method	**0.13435**	**0.08354**	**0.05954**	**0.05195**
w-method	0.15875	0.09133	0.06169	0.0541

Table 9: Simulated values of $\hat{p}$ for the K-test, type 1 of LRFNs

n	5	10	30	100
r	100			
Bootstrap	0.22788	0.15132	0.09559	0.07145
d-method	**0.22117**	**0.15006**	**0.09534**	**0.07049**
w-method	0.23952	0.1579	0.09693	0.07106
r	200			
Bootstrap	0.22278	0.14885	0.0923	0.06783
d-method	**0.2184**	**0.14639**	**0.09096**	**0.06748**
w-method	0.23506	0.15312	0.09303	0.0686
r	1000			
Bootstrap	0.21723	0.14492	**0.08758**	0.06296
d-method	**0.2153**	**0.14143**	0.08865	**0.06188**
w-method	0.23495	0.14697	0.08841	0.06429

Table 10: Simulated values of $\hat{p}$ for the K-test, type 2 of LRFNs

n	5	10	30	100
r	100			
Bootstrap	0.17526	**0.10734**	0.0729	0.0643
d-method	**0.16805**	0.10808	**0.07281**	**0.06385**
w-method	0.18535	0.11309	0.07356	0.06466
r	200			
Bootstrap	0.16682	**0.09992**	0.06944	0.05768
d-method	**0.16205**	0.10241	**0.06901**	**0.05714**
w-method	0.18389	0.10714	0.06958	0.05945
r	1000			
Bootstrap	0.1615	0.09987	**0.06424**	0.055
d-method	**0.15917**	**0.09716**	0.06578	**0.05404**
w-method	0.17689	0.10472	0.06509	0.05627

Table 11: Simulated values of $\hat{p}$ for the K-test, type 3 of LRFNs

n	5	10	30	100
r	100			
Bootstrap	0.03375	0.04906	0.0562	0.05892
d-method	0.04047	0.05229	**0.05618**	**0.05827**
w-method	**0.0422**	**0.04994**	0.05747	0.05981
r	200			
Bootstrap	0.02988	0.04449	0.05184	0.05395
d-method	0.03659	0.04611	0.05224	**0.05331**
w-method	**0.03869**	**0.04621**	**0.05132**	0.05405
r	1000			
Bootstrap	0.02748	0.03862	0.04817	0.05064
d-method	0.03412	0.04158	**0.04862**	**0.04952**
w-method	**0.03524**	**0.04234**	0.04844	0.0514

Table 12: Simulated values of $\hat{p}$ for the GRMCG-test, type 1 of LRFNs

n	5	10	30	100
r	100			
Bootstrap	**0.08495**	**0.08385**	0.06892	0.06107
d-method	0.09687	0.0857	**0.06792**	0.06094
w-method	0.0986	0.08808	0.06904	**0.06037**
r	200			
Bootstrap	**0.07903**	0.08167	0.0635	0.05769
d-method	0.093	**0.08148**	**0.06348**	**0.05743**
w-method	0.09391	0.08245	0.06545	0.05754
r	1000			
Bootstrap	**0.07574**	**0.07701**	0.06077	0.05299
d-method	0.08984	0.07804	0.06129	**0.05176**
w-method	0.09241	0.07787	**0.06038**	0.05383

Table 13: Simulated values of $\hat{p}$ for the GRMCG-test, type 2 of LRFNs

n	5	10	30	100
r	100			
Bootstrap	0.03593	**0.05029**	0.05763	0.05949
d-method	0.03923	0.05274	0.05812	**0.05941**
w-method	**0.03936**	0.05256	**0.05744**	0.06028
r	200			
Bootstrap	0.03141	0.04413	0.05423	0.05346
d-method	0.03491	0.04529	0.05355	**0.05321**
w-method	**0.0356**	**0.04599**	**0.05282**	0.05517
r	1000			
Bootstrap	0.02805	0.04163	0.0489	0.04992
d-method	**0.03221**	0.04089	**0.05046**	**0.05001**
w-method	0.03155	**0.04306**	0.04887	0.052

Table 14: Simulated values of $\hat{p}$ for the GRMCG-test, type 3 of LRFNs