The Impact of Statistics for Benchmarking in Evolutionary Computation Research

Tome Eftimov
Computer Systems Department,
Jožef Stefan Institute
Ljubljana, Slovenia
tome.eftimov@ijs.si

Peter Korošec
Computer Systems Department,
Jožef Stefan Institute
Ljubljana, Slovenia
Faculty of Mathematics, Natural Sciences and Information Technologies
Koper, Slovenia
peter.korosec@ijs.si

ABSTRACT

Benchmarking theory in evolutionary computation research is a crucial task that should be properly applied in order to evaluate the performance of a newly introduced evolutionary algorithm with performance of state-of-the-art algorithms. Benchmarking theory is related to three main questions: which problems to choose, how to setup experiments, and how to evaluate performance. In this paper, we evaluate the impact of different already established statistical ranking schemes that can be used for evaluation of performance in benchmarking practice for evolutionary computation. Experimental results obtained on Black-Box Benchmarking 2015 showed that different statistical ranking schemes, used on the same benchmarking data, can lead to different benchmarking results. For this reason, we examined the merits and issues of each of them regarding benchmarking practices.

CCS CONCEPTS

• **Mathematics of computing** → **Hypothesis testing and confidence interval computation**; **Evolutionary algorithms**;

KEYWORDS

Statistical analysis, benchmarking, evolutionary algorithms

ACM Reference Format:
Tome Eftimov and Peter Korošec. 2018. The Impact of Statistics for Benchmarking in Evolutionary Computation Research. In *Proceedings of the Genetic and Evolutionary Computation Conference 2018 (GECCO '18 Companion),* Jennifer B. Sartor, Theo D'Hondt, and Wolfgang De Meuter (Eds.). ACM, New York, NY, USA, Article 4, 8 pages. https://doi.org/10.1145/3205651.3208232

1 INTRODUCTION

To determine the strengths and weaknesses of a newly introduced evolutionary algorithm, its performance should be compared with performances of state-of-the-art algorithms. Over last years, several competitions for optimization algorithms at evolutionary computation conferences (e.g., GECCO, CEC) are organized, in which the proposed algorithms are compared using a set of benchmark functions. The idea behind those comparisons is that by using the results obtained on different functions, the "best" algorithm (i.e. algorithm that perform best in average over all benchmark functions) can be found, or to use the benchmarking results to transfer the knowledge onto a real-world problem. To find the "best" algorithm, benchmarking is a key scientific technique and having a good one is a difficult task [16]. The first step in benchmarking theory is that the problem domain should be defined and it is a crucial task since it restricts the domain to which any conclusions made can be generalized. Finding good test functions is a challenging task because they should be uniformly distributed in the space of all possible functions from the problem domain. Currently, a lot of papers presented in the domain of evolutionary algorithms used the black-box optimization algorithm benchmarking (BBOB) [13]. After the set of benchmark functions is selected, benchmarking results depend on the performance metrics and statistical ranking techniques. Statistical analyses that are performed are crucial and need to be made with a great care because they provide the information from where the conclusions are made, so an appropriate statistical analysis should be performed. Further, using the benchmarking results, the obtained knowledge can be transferred onto a real-world problem. For this, Exploratory Landscape Analysis (ELA) can be used [15]; the idea is to describe the problem by high-level empirical properties, to find out which algorithms perform especially well on certain property combinations, and to develop ways to automatically extract problem properties from a concrete problem instance. The selection of the benchmark problems, the experiments setups, and ELA are not a subject of this paper.

In this paper, we evaluate the impact of different already established statistical ranking schemes that can be used in benchmarking practice for evolutionary algorithms. For this reason, in Section II, most commonly used statistical approaches together with a recently proposed approach used for benchmarking in evolutionary computation are explained. Section III provides information about the selected benchmarking, the compared algorithms, and the results obtained from the comparisons. In Section IV, discussion about the impact of statistics for benchmarking in evolutionary computation is provided. Section V presents the conclusions of the study.

GECCO '18 Companion, July 15–19, 2018, Kyoto, Japan

ACM ISBN 978-1-4503-5764-7/18/07.
https://doi.org/10.1145/3205651.3208232

2 STATISTICAL ANALYSIS IN BENCHMARKING PRACTICES

The statistical analysis of the performance of a new algorithm with regard to state-of-the-art algorithms is in most cases made using statistical comparisons that follow the idea of hypothesis testing [5]. Statistical comparisons can be conducted in two scenarios: single-problem analysis and multiple problem analysis. Single-problem analysis involves analyzing data from multiple runs of evolutionary algorithms on one problem (i.e. test function). This happens because a single run on a single problem instance is not enough to make conclusions, since the algorithms are stochastic in nature, meaning we do not have any guarantee that the result will be the same for every run; even the path leading to the final solution is often different. Multiple-problem analysis is a scenario when the algorithms are compared on a set of benchmark functions, and it involves a benchmarking theory.

Nowadays, many researchers have problems making a statistical comparison because statistical tools are relatively complex and there are many to choose from. The problem is in selecting the right statistic to apply on a selected performance measure. For example, researchers often report either average or median without being aware that averaging is sensitive to outliers and both, the average and median, are sensitive to statistically insignificant differences in the data. Even reporting the standard deviation of the average needs to be made with care since large variances result from the presence of outliers. Furthermore, these statistics only describe the data and do not provide any additional information about the relations that exist between the data. For this, a statistical test needs to be applied. Additionally, the selection of a statistic can influence the outcome of a statistical test. This means that applying the appropriate statistical test requires knowledge of the necessary conditions about the data that must be met in order to apply it. This step is often omitted and researchers simply apply a statistical test, in most cases borrowed from a similar published study, which is inappropriate for their data set. This kind of misunderstanding is all too common in the research community and can be observed in many high-ranking journal papers. Even if the statistical test is the correct one, if the experimental design is flawed (e.g., comparison of results of tuned and non-tuned algorithms) their conclusions will be wrong. This is sometimes done on purpose to mislead the reader in believing that the author's results are better than they actually are.

In machine learning research, Demšar [6] has examined several nonparametric statistical tests that can be used for comparing classifiers on a benchmark of data sets. He has also provided an experimental setup to find the power of the selected statistical test. Following his idea, Garcia et al. [7] has presented a study on the use of nonparametric test for analyzing the behavior of evolutionary algorithms for optimization problems. This approach is one of the most commonly used approach for making a statistical comparison of evolutionary algorithms, so we call it a "common approach" (CA). Recently, a new approach for making a statistical comparison in multiple problem analysis (i.e. Deep Statistical Comparison, DSC) was presented [8], which gives more robust statistical results when outliers (i.e. poor runs) or statistically insignificant differences between data values exist. In this paper, we evaluate ranking schemes used by these approaches and their impact to the benchmarking results. For this reason, we are going to explain them in more detail. However, before we continue, we would like explicitly to point that we are talking only for statistical significance and not practical significance. To explain the difference between them, let us assume that two algorithms are used for optimization of a given problem. Let, the first algorithm solves the problem within 10^{-10} of error of the global optimum and the second one to within 10^{-16}. Between them, statistical significance could be found; however this difference most probably is not significant in a practical sense.

2.1 Common approach

Working with evolutionary algorithms in multiple-problem analysis requires finding a unique representative value from multiple runs for each algorithm on each problem. There are a lot of papers in which authors reported best/worst (i.e. the sample min/max) values out of multiple runs for each algorithm and each problem and further they are analyzed using an appropriate statistical test. However this kind of comparison is wrong because the sample min/max is a biased estimator of the expected value. For this reason, Garcia et al. [10] suggest using an average of multiple runs as a representative value for each algorithm on each problem. Average is an unbiased estimator of the expected value; however it can be affected by outliers (i.e. poor runs of evolutionary algorithms) and instead median can be used as a representative value. So by using the common approach, either average or median from the multiple runs obtained on a single problem can be used as a representative value involved in the multiple-problem scenario for specific algorithm on specific problem. Further the data obtained for multiple-problem analysis should be analyzed using an appropriate omnibus statistical test [10] .

2.2 Deep statistical comparison approach

Using the common approach, we need to be aware that averages are known to be sensitive to outliers. In general, outliers can be disregarded using some techniques, but they need to be used with great care. For multiple-problem analysis, removing outliers is questionable because only the results for certain problems would be changed. In stochastic optimization it can happen that in a set of independent runs the average result of one problem for a given algorithm is better than another algorithm, but in the next set of independent runs the average result for the same problem and the same algorithm could be worse than the other algorithm, and this happen because in any new set of independent runs different poor runs exist. The common approach is also used with medians because they are less sensitive to outliers. However, in both cases the results can still be affected by the ranking scheme of some statistical tests. This happens when differences between the averages or medians are in some ε-neighborhood (e.g., 10^{-9},10^{-10}, etc.), so algorithms consequently obtain different rankings because there are no ties presented. It can happen that the distribution of the data is the same, the medians are in some ε-neighborhood and the algorithms will be ranked differently, but they need to obtain the same ranking; even more the distribution can be different, the medians can be the same and the algorithms will be ranked as the same, but they need to obtain different rankings. All this lead to a need for new robust analyses that can be used to compose a sample for each algorithm over multiple problems, which

can be used for further analysis using a standard omnibus statistical test.

For these reasons, Deep Statistical Comparison (*DSC*) for comparing meta-heuristic stochastic optimization algorithms over multiple single-objective problems was recently proposed [8]. Its main contribution is its ranking scheme, which is based on the whole distribution, instead of using only one statistic to describe the distribution, such as average or median. The approach consists of two steps. The first step uses a newly proposed ranking scheme to obtain data in order to make a statistical comparison. The ranking scheme is based on comparing distributions using a statistical test, such as, the two-sample *Kolmogorov-Smirnov (KS) test* or the two-sample *Anderson-Darling (AD) test* [9]. All pairwise comparisons between the compared algorithms must be made, and the obtained p-values are organized in a matrix. Further, because multiple pairwise comparisons are made, these p-values are corrected using the *Bonferroni correction* [10] in order to control the family-wise error, FWER [14]. The FWER is the probability of making one or more false discoveries, or type I errors, among all hypotheses when performing multiple hypotheses tests. The matrix is then checked for transitivity, and on this basis the algorithms obtain their rankings. The second step is a standard omnibus statistical test, which uses data obtained by the *DSC* ranking scheme as the input data.

3 EXPERIMENTS

To see the impact of different statistical ranking schemes for benchmarking in evolutionary computation, experiments were performed using results of algorithms presented at the sixth Black-Box Optimization Benchmarking 2015 (BBOB 2015) workshop that was a part of GECCO 2015 [3]. Two experiments are presented. In the first, we randomly selected combinations of three algorithms, and we reported only three combinations that have different results. We used three algorithms because the impact of statistics is more emphasized when the number of compared algorithms is lower. In the second, we presented a most commonly used scenario in benchmarking, which is multiple comparisons with a control algorithm, for which we used 15 algorithms presented at GECCO 2015.

In both experiments, we used the common approach and deep statistical comparison approach to transform raw data for multiple-problem analysis. For the common approach, we used average and median from algorithm's multiple runs on a single problem, as a representative values. For the deep statistical comparison approach, we used the DSC ranking scheme to obtain a sample for multiple-problem analysis. Further, the transformed data for multiple-problem analysis was analyzed using an appropriate omnibus statistical test. The appropriate omnibus statistical test was selected after checking the required conditions for the safe use of the parametric test (i.e. normality of the data, homscedasticity of the variances, and independence).

The comparisons were performed in the *R programming language*. The *Kolmogorov-Sminrov test for normality*, which is a part of the "*stat*" package [18], is used to check the normality condition. The *Levene's test* from the "*lawstat*" package [11] is used to check the homoscedasticity of the variances (i.e. if data sets have the same variance). For the DSC ranking scheme, we used the *two-sample Kolmogorov-Smirnov (KS) test* from the *stat* package [18] and *two-sample Anderson-Darling test* from the *kSamples* package [19].

There are also already established websites that work as e-Learning tools, one for using the common approach (http://ws.ijs.si/statTool/), and the other for using the DSC approach (http://ws.ijs.si/dsc/).

3.1 Black-Box Optimization Benchmarking 2015

The BBOB 2015 was a competition that provided single-objective functions for benchmarking. The test functions, F, are from five groups, that represent

- separable functions,
- functions with low or moderate conditioning,
- functions with high conditioning and unimodal,
- multi-modal functions with adequate global structure, and
- multi-modal functions with week global structure.

More details about them can be found in [12].

3.2 Algorithms

From the competition, 15 algorithms were used: BSif [17], BSifeg [17], BSqi [17], BSrr [17], CMA-CSA [1], CMA-MSR [1], CMA-TPA [1], GP1-CMAES [2], GP5-CMAES [2], RAND-2xDefault [4], RF1-CMAES [2], RF5-CMAES [2], Sif [17], Sifeg [17], and Srr [17]. For the experiments, a statistical comparison was performed by comparing them on 22 different noiseless functions with the dimension fixed at 10.

3.3 Comparison of three algorithms

In this experiment, the results of statistical comparison of three combinations of algorithms are presented. We selected them from a set of randomly generated combinations in order to show different scenarios that can happen. Each combination is a statistical comparison of three randomly selected algorithms. Data samples for multiple-problem analysis were generated using the common approach with averages and median, respectively, and the DSC ranking scheme using two different criteria for comparing distributions, the *two-sample KS test* and *two-sample AD test*, respectively. For each data sample the required conditions for the safe use of the parametric tests were checked, and for each one the *Friedman test* is an appropriate omnibus statistical test. Table 1 presents the p-values of the comparisons. For the first combination, results obtained by the common approach differ from the results obtain by the DSC ranking scheme. Using the common approach the null hypothesis is rejected so there is a difference between the performance of the compared algorithms, however using the DSC approach the null hypothesis is not rejected. Additionally the p-values for the common approach differ for using averages and medians, and same happen also for the DSC ranking scheme with different criteria for comparing distributions. So, it follows that different statistics that are used in the benchmarking affect the rankings, which are further used in calculation of the test statistic of the omnibus statistical test, and they lead to different p-values. For the second and the third combination the result is statistically the same. In the second combination, the null hypothesis is not rejected using both approaches with different statistical criteria, while in the third combination, the null hypothesis is rejected for both approaches. Because only in the first combination the results for both approaches differ, it is presented in more detail

Table 1: Statistical comparisons of three algorithms

	Algorithms	Common approach (with averages)	Common approach (with medians)	DSC approach (KS)	DSC approach (AD)
		p_{valueF}	p_{valueF}	p_{valueF}	p_{valueF}
1	*GP5-CMAES*, *Sifeg*, *BSif*	*(.02)	*(.04)	(.42)	(.44)
2	*BSifeg*, *RF1-CMAES*, *BSrr*	(.16)	(.23)	(.28)	(.48)
3	*BSrr*, *RAND-2xDefault*, *Srr*	*(.00)	*(.00)	*(.00)	*(.00)

* indicates that the null hypothesis is rejected, using $\alpha = 0.05$
p_{valueF} corresponds to the p-value obtained by the *Friedman test*

analysis in order to see the impact of different statistical criteria. Table 2 presents the rankings obtained for each compared algorithm on each single problem involved in the benchmarking using both approaches for different statistical criteria.

From Table 2, we can see that there are problems for which the compared algorithms obtain different rankings on the same problem using both approaches with different statistical criteria. For example, on f_{13}, the rankings are 2.00, 3.00, 1.00, using the common approach with averages, 1.00, 2.00, 3.00, using the common approach with medians, and 1.50, 1.50, 3.00, using the DSC approach with the *two-sample KS test* and *two-sample AD test*. From the obtained rankings on f_{13}, it follows that the common approach with different statistical criteria (i.e. average or median) provides different results that lead to different conclusions. Both results also differ from the results that are obtained using the DSC approach, where instead of average or median, distributions of the multiple runs of each algorithm on that problem are compared. In Figure 1, purple and green lines present averages and medians of the multiple runs of each algorithm obtained on f_{13}, respectively, while red (step) lines present the empirical cumulative distributions of the multiple runs of each algorithm obtained on f_{13}. In the case of the common approach lower value is better because we are dealing with minimization. From Figure 1, it is clear that the rankings with regard to averages are 2.00, 1.00, and 3.00, however using medians it is not obvious because the medians for the first and the second algorithm (i.e. GP5-CMAES and Sifeg) are overlapping. For this reason, Table 3 presents averages and medians of multiple runs for each algorithm on f_{13}.

Using medians, the rankings are 1.00, 2.00, and 3.00. In the case of averages, GP5-CMAES has an average 3.07 and Sifeg has 1.77, while the medians are 1.53 and 1.57, respectively, so they swap their rankings using different statistical criteria in the same approach. This happens because in the case of averages, an average can be affected by outliers (i.e. poor runs) and this actually happens for the GP5-CMAES. Using its empirical cumulative distribution, we can see that there are some poor runs around the value 15 that affect the average of multiple runs. Because averaging is sensitive to poor runs, which can happen with evolutionary algorithms, medians are also used. In this example, medians for GP5-CMAES and Sifeg are in some small ε-neighborhood, however they obtain different rankings since they are different. All these problems can be omitted with the DSC ranking scheme because it works with comparing whole distributions and not only one statistic that describe the distribution.

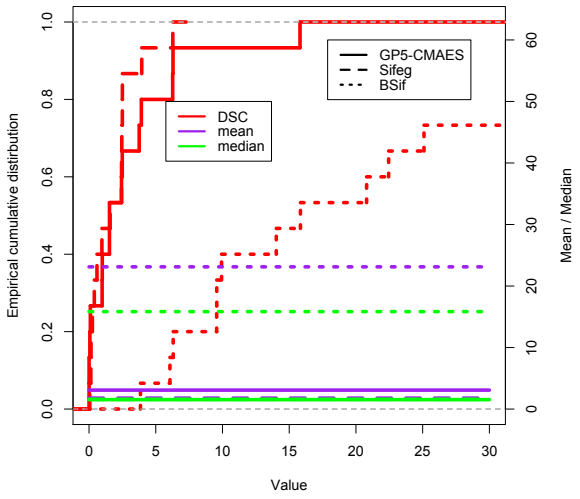

Figure 1: Empirical cumulative distributions (step functions) and means and medians (horizontal lines) for f_{13}

In Figure 2, averages, medians, and empirical cumulative distributions of the multiple runs obtained on f_{18} are presented. Using the common approach with either averages or medians provides the same rankings, while the DSC approach with different criteria for comparing distributions provides different rankings. The DSC ranking scheme involves all pairwise comparisons between the compared algorithms. In the case when the *two-sample KS test* is used the p-values are 0.00 (GP5-CMAES, Sifeg), 0.00 (GP5-CMAES, BSif), and 0.03 (Sifeg, BSif). Because multiple pairwise comparisons are made, these p-values are further corrected using the *Bonferroni correction*. The transitivity of these comparisons is satisfied and the algorithms are spit into disjoint sets {GP5-CMAES} and {Sifeg, BSif}, according to which they obtain their rankings, 1.00, 2.50, and 2.50. In the case when the *two-sample AD test* is used the p-values are 0.00 (GP5-CMAES, Sifeg), 0.00 (GP5-CMAES, BSif), and 0.01 (Sifeg, BSif). These p-values are further also corrected using the *Bonferroni correction*, but the transitivity is not satisfied, so the algorithms obtained their rankings according to their averages. If this happens it is better to use the *two-sample AD test* because it is more powerful and it can better detect differences than the *two-sample KS*

Table 2: Rankings for the algorithms A_1=GP5-CMAES, A_2=Sifeg, and A_3=BSif

F	Common approach (with averages)			Common approach (with medians)			DSC approach (*KS test*)			DSC approach (*AD test*)		
	A_1	A_2	A_3	A_1	A_2	A_3	A_1	A_2	A_3	A_1	A_2	A_3
f_1	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00
f_2	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00
f_3	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00	3.00	2.00	1.00
f_4	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.00
f_5	3.00	1.50	1.50	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00	2.00
f_6	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.00	2.50	1.00	2.50
f_7	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.50	2.50	1.00	2.50	2.50
f_8	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.50	1.50	3.00	1.00	2.00
f_9	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.50	1.50	3.00	1.50	1.50
f_{10}	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.50	2.50	1.00	2.50	2.50
f_{11}	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.50	2.50	1.00	2.50	2.50
f_{12}	3.00	2.00	1.00	3.00	2.00	1.00	3.00	1.50	1.50	3.00	1.50	1.50
f_{13}	**2.00**	**1.00**	**3.00**	**1.00**	**2.00**	**3.00**	**1.50**	**1.50**	**3.00**	**1.50**	**1.50**	**3.00**
f_{14}	3.00	1.00	2.00	1.00	2.00	3.00	2.50	2.50	1.00	2.50	2.50	1.00
f_{15}	2.00	1.00	3.00	2.00	1.00	3.00	2.00	2.00	2.00	2.00	2.00	2.00
f_{16}	2.00	1.00	3.00	2.00	1.00	3.00	2.00	2.00	2.00	2.00	1.00	3.00
f_{17}	1.00	2.00	3.00	1.00	2.00	3.00	1.00	2.50	2.50	1.00	2.50	2.50
f_{18}	**1.00**	**2.00**	**3.00**	**1.00**	**2.00**	**3.00**	**1.00**	**2.50**	**2.50**	**1.00**	**2.00**	**3.00**
f_{19}	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.50	1.50	3.00	1.50	1.50
f_{20}	3.00	1.00	2.00	3.00	1.00	2.00	3.00	1.50	1.50	3.00	1.50	1.50
f_{21}	1.00	2.00	3.00	1.00	2.00	3.00	2.00	2.00	2.00	2.00	2.00	2.00
f_{22}	**1.00**	**2.00**	**3.00**	**2.00**	**1.00**	**3.00**	**2.00**	**2.00**	**2.00**	**2.00**	**2.00**	**2.00**

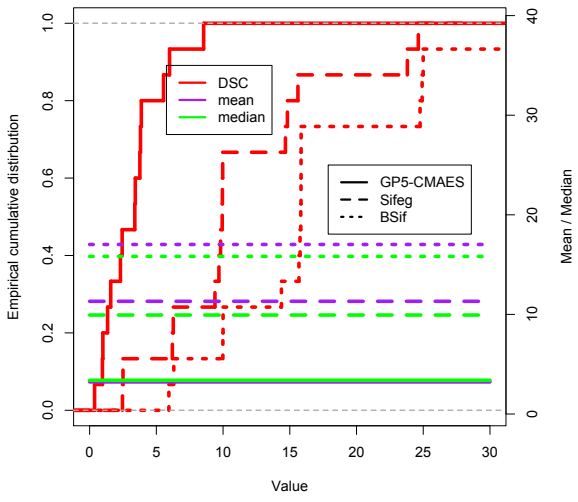

Figure 2: Empirical cumulative distributions (step functions) and means and medians (horizontal lines) for f_{18}

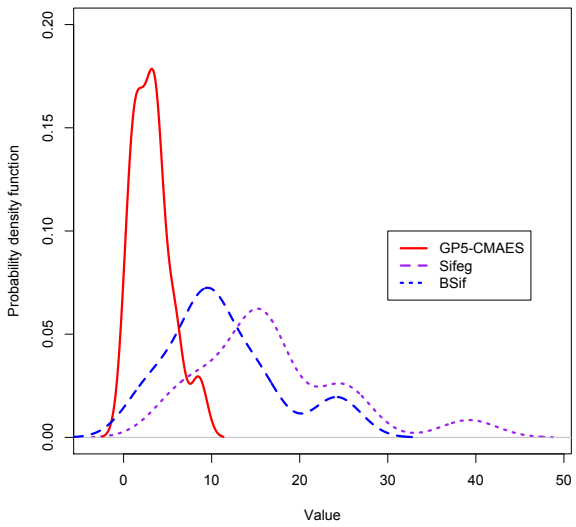

Figure 3: Probability density functions of multiple runs for each algorithm obtained on f_{18}

test when distributions vary in shift only, in symmetry only, or have the same average and standard deviation but differ on the tail ends only [9]. For this reason, in Figure 3, probability density functions of the multiple runs for each algorithm obtained on f_{18} are presented. Using this figure, the *two-sample KS test* is not able to detect the difference that exist in the shift and tail ends between distributions of Sifeg and BSif.

In Figure 4, averages, medians, and empirical cumulative distributions of the multiple runs obtained on f_{22} are presented. Table 4 presents averages and medians of the multiple runs for each algorithm obtained on f_{22}, which are used by the common approach. In the case of the DSC approach, both statistical criteria used for comparing distributions give the same rankings 2.00, 2.00, and 2.00.

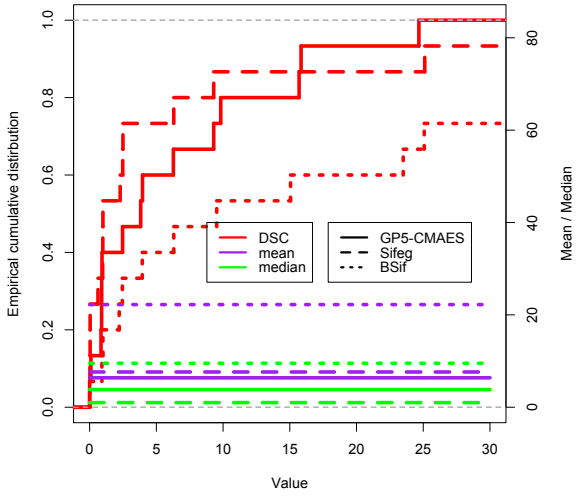

Figure 4: Empirical cumulative distributions (step functions) and means and medians (horizontal lines) for f_{22}

Table 3: Descriptive statistics obtained on f_{13}

Algorithm	Average	Median
GP5-CMAES	3.07	1.53
Sifeg	1.77	1.57
BSif	23.11	15.84

From the figure, it is not clear if there is a difference between distributions. For this reason, in Table 5 p-values obtained for multiple pairwise comparisons using both statistical tests for comparing distributions are presented. Further these p-values are corrected using the *Bonferroni correction*; the transitivity is satisfied and all algorithms belong to one set, so they obtain the same ranking.

Table 4: Descriptive statistics obtained on f_{22}

Algorithm	Average	Median
GP5-CMAES	6.37	3.82
Sifeg	7.63	0.99
BSif	22.22	9.52

We need to point that if the benchmarking result detects a difference between the performance of the compared algorithms (i.e. the null hypothesis is rejected), post-hoc statistical test should be performed in order to see from where this difference comes. However, we omitted this step because we would like to point only on the impact of statistics on benchmarking in evolutionary computation. The influence that appears in the omnibus statistical test is also transferred on the post-hoc level.

Table 5: P-values for all pairwise comparisons involved in DSC ranking scheme obtained on f_{22}

Pairs of algorithms	p-value (KS test)	p-value (AD test)
(GP5-CMAES, Sifeg)	0.67	0.51
(GP5-CMAES, BSif)	0.38	0.09
(Sifeg, BSif)	0.18	0.06

3.4 Multiple comparisons with a control algorithm

In this experiment, we present multiple comparisons with a control algorithm. We tested this scenario using multiple pairwise comparisons performed using the *Wilcoxon test*. We did this because DSC approach requires it, since otherwise it can happen that the DSC ranking scheme is affected by p-value correction when number of algorithms increases. As a control algorithm, CMA-CSA was selected. In table 6, p-values for each pairwise comparison using both statistical approaches with different statistical criteria are presented. From the p-values obtained, it is obvious that different statistical criteria lead to different test statistic values. For example, in the comparison of CMA-CSA with CMA-MSR, the DSC approach with both statistical criteria provides the same statistical result (i.e the null hypothesis is not rejected), the common approach with averages provides that the null hypothesis is rejected, while the common approach with medians also shows that the null hypothesis is not rejected.

From Table 6, if we focus on the DSC approach with the *two-sample KS test*, it follows that the performance of CMA-CSA is different from the performance of BSif, BSifeg, BSqi, BSrr, GP1-CMAES, GP5-CMAES, RAND-2xDefault, RF1-CMAES, RF5-CMAES, Sif, Sifeg, and Srr, while there is no difference with regard to CMA-MSR and CMA-TPA. However, identified differences are valid for independent comparisons because we have all vs. one scenario. If we would like to show that there is a difference between the performance of CMA-CSA and other 12 algorithms, then the true p-value must be calculated combining the p-values of the independent comparisons using the following equation

$$p_{value} = 1 - \prod_{i=1}^{k-1}[1 - p_{value_{H_i}}], \quad (1)$$

where $k-1$ is the number of independent pairwise comparisons that are combined, or in our case 12. Using this equation it follows that the p-value is 0.02, so the CMA-CSA has statistical different performance than the algorithms: BSif, BSifeg, BSqi, BSrr, GP1-CMAES, GP5-CMAES, RAND-2xDefault, RF1-CMAES, RF5-CMAES, Sif, Sifeg, and Srr.

4 DISCUSSION

Benchmarking in evolutionary computation is a task that should be performed to compare new algorithms with the existing ones. It is related to three main questions that should be treat equally with a great care: which problems to choose, how to setup experiments, and how to evaluate performance. We deal with the last question related to different statistical approaches that can be used for evaluation, when problems are chosen and experiments are set. This questions can be split into two others: which performance metric is selected and which ranking scheme is used in the evaluation. In our experiments,

Table 6: Multiple comparisons with a control algorithm (CMA-CSA) by using multiple *Wilcoxon* tests

j	CMA-CSA vs.	$p-value_{(DSC;KS)}$	$p-value_{(DSC;AD)}$	$p-value_{(CA;average)}$	$p-value_{(CA;median)}$
1	BSif	4.847534e-03	4.847534e-03	8.476892e-04	8.476892e-04
2	BSifeg	7.768118e-03	7.768118e-03	1.086096e-03	1.758873e-03
3	BSqi	3.081757e-03	7.768118e-03	1.227287e-03	2.223195e-03
4	BSrr	7.768118e-03	7.768118e-03	1.086096e-03	1.758873e-03
5	CMA-MSR	1.000000e+00	7.655945e-01	4.757041e-02	7.628835e-02
6	CMA-TPA	1.000000e+00	3.457786e-01	4.757041e-02	4.654448e-01
7	GP1-CMAES	1.451271e-05	8.553503e-06	4.768372e-07	6.411516e-05
8	GP5-CMAES	8.553503e-06	8.553503e-06	4.768372e-07	6.411516e-05
9	RAND-2xDefault	5.049088e-06	5.049088e-06	6.411516e-05	6.411516e-05
10	RF1-CMAES	5.049088e-06	5.049088e-06	4.768372e-07	6.411516e-05
11	RF5-CMAES	5.049088e-06	5.049088e-06	4.768372e-07	6.411516e-05
12	Sif	6.301490e-04	3.759531e-04	9.600603e-04	1.385265e-03
13	Sifeg	6.301490e-04	6.301490e-04	1.385265e-03	3.504330e-03
14	Srr	1.056542e-03	6.301490e-04	1.227287e-03	2.495261e-03

as performance metric the best solution of the algorithm is used. The evaluation was made to show how different statistical approaches can lead to different benchmarking result. For this reason, two already established approaches are evaluated, the common approach and the recently proposed DSC approach.

The common approach suggests using an average or a median of multiple runs for an algorithm on a given problem as a representative value that will be used in the benchmarking scenario (multiple-problem analysis). The DSC approach is based on a ranking scheme of comparing whole distributions of multiple runs instead of using one statistic, either average or median. Experiments showed that both approaches can lead to different benchmarking results, and this happens because different statistical criteria used for evaluation. Performing benchmarking using averages, we need to be aware that averages are sensitive to outliers, which are poor runs of evolutionary algorithms. Instead, medians can be used, but the problem is that they can be in some ε-neighborhood (i.e. insignificant statistical difference) and will be ranked as different, but should be ranked as the same. This can be solved by adding prior information for the ε-neighborhood having knowledge about the problem, but the question that appears is how to select the ε-neighborhood for a given problem dynamically because problems that are involved in the benchmarking scenario can have different ε-neighborhoods regarding different data ranges of solutions they provide. All these problems can be omitted using the DSC approach, which takes into account the whole distribution of the multiple runs for an algorithm obtained on a given problem, and it is based on comparing distributions. However, the statistical criteria that will be used for comparing distributions also needs to be selected appropriately. Experiments showed that different criteria for comparing distributions can lead to different benchmarking results. In a lot of papers, authors select the ranking method, which returns the desired outcome (e.g., average, median), but this lead to one of the main questions related to benchmarking theory: do we need to have a breakthrough in evolutionary computation research by in-depth understanding of algorithms performance and not only focusing on an applicative research loosing the information why some algorithm works well or not. Using distributions can provide more information about algorithm performance and maybe it is time to move from null-hypothesis testing from classical inferential statistics, which is most commonly used nowadays in benchmarking theory, to approaches from Bayesian statistics.

5 CONCLUSIONS

Working with evolutionary computation algorithms, benchmarking theory plays an important role for objectively comparing new algorithms in order to demonstrate their strengths and weaknesses, where they can be improved. Nowadays, researchers in the field of evolutionary computation have recognized that new standards for benchmarking in evolutionary computation research are needed. For this reason, in this paper we examined the merits and issues regarding different already established statistical criteria used in benchmarking practices, when problem domain is already chosen and experiments are set. Experimental results shown that different statistical criteria used on the same benchmark data can lead to different benchmarking result, due to the issues present in each statistical criteria of which we need to be aware when performing benchmarking.

ACKNOWLEDGMENTS

This work was supported by the project from the Slovenian Research Agency (research core funding No. P2-0098), from the European Union's Horizon 2020 research and innovation program under grant agreement No. 692286, and from COST Action CA15140, supported by COST.

REFERENCES

Asma Atamna. 2015. Benchmarking IPOP-CMA-ES-TPA and IPOP-CMA-ES-MSR on the BBOB Noiseless Testbed. In *Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference*. ACM, 1135–1142.

Lukáš Bajer, Zbyněk Pitra, and Martin Holeňa. 2015. Benchmarking gaussian processes and random forests surrogate models on the BBOB noiseless testbed. In *Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary*

Computation Conference. ACM, 1143–1150.

BBComp Black Box Optimization Competition. [n. d.]. Black-Box Benchmarking 2015. http://coco.gforge.inria.fr/doku.php?id=bbob-2015. ([n. d.]). Accessed: 2016-02-01.

Dimo Brockhoff, Bernd Bischl, and Tobias Wagner. 2015. The impact of initial designs on the performance of matsumoto on the noiseless BBOB-2015 testbed: A preliminary study. In *Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference*. ACM, 1159–1166.

Wayne W Daniel and Chad Lee Cross. 1995. Biostatistics: a foundation for analysis in the health sciences. (1995).

Janez Demšar. 2006. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research* 7 (2006), 1–30.

Joaquín Derrac, Salvador García, Daniel Molina, and Francisco Herrera. 2011. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation* 1, 1 (2011), 3–18.

Tome Eftimov, Peter Korošec, and Barbara Koroušić Seljak. 2017. A novel approach to statistical comparison of meta-heuristic stochastic optimization algorithms using deep statistics. *Information Sciences* 417 (2017), 186–215.

Sonja Engmann and Denis Cousineau. 2011. Comparing distributions: the two-sample Anderson-Darling test as an alternative to the Kolmogorov-Smirnoff test. *Journal of Applied Quantitative Methods* 6, 3 (2011), 1–17.

Salvador García, Daniel Molina, Manuel Lozano, and Francisco Herrera. 2009. A study on the use of non-parametric tests for analyzing the evolutionary algorithms behaviour: a case study on the CEC2005 special session on real parameter optimization. *Journal of Heuristics* 15, 6 (2009), 617–644.

Joseph L Gastwirth, Yulia R Gel, W L Wallace Hui, Vyacheslav Lyubchich, Weiwen Miao, and Kimihiro Noguchi. 2015. *lawstat: Tools for Biostatistics, Public Policy, and Law*. https://CRAN.R-project.org/package=lawstat R package version 3.0.

Nikolaus Hansen, Anne Auger, Steffen Finck, and Raymond Ros. 2010. Real-parameter black-box optimization benchmarking 2010: Experimental setup. (2010).

Nikolaus Hansen, Anne Auger, Olaf Mersmann, Tea Tusar, and Dimo Brockhoff. 2016. COCO: A platform for comparing continuous optimizers in a black-box setting. *arXiv preprint arXiv:1603.08785* (2016).

Mark J Laan van der, Sandrine Dudoit, and Katherine S Pollard. 2004. Multiple testing. Part II. Step-down procedures for control of the family-wise error rate. *Statistical applications in genetics and molecular biology* 3, 1 (2004), 1–33.

Olaf Mersmann, Bernd Bischl, Heike Trautmann, Mike Preuss, Claus Weihs, and Günter Rudolph. 2011. Exploratory landscape analysis. In *Proceedings of the 13th annual conference on Genetic and evolutionary computation*. ACM, 829–836.

Olaf Mersmann, Mike Preuss, and Heike Trautmann. 2010. Benchmarking evolutionary algorithms: Towards exploratory landscape analysis. In *International Conference on Parallel Problem Solving from Nature*. Springer, 73–82.

Petr Pošík and Petr Baudiš. 2015. Dimension selection in axis-parallel brent-step method for black-box optimization of separable continuous functions. In *Proceedings of the Companion Publication of the 2015 on Genetic and Evolutionary Computation Conference*. ACM, 1151–1158.

R Core Team. 2015. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. https://www.R-project.org/

Fritz Scholz and Angie Zhu. 2016. *kSamples: K-Sample Rank Tests and their Combinations*. https://CRAN.R-project.org/package=kSamples R package version 1.2-4.