Is the statistical significance between stochastic optimization algorithms' performances also significant in practice?

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

Peter Korošec
Computer Systems Department,
Jožef Stefan Institute
Ljubljana, Slovenia
peter.korosec@ijs.si

ABSTRACT

To transfer the learned knowledge that is coming from benchmarking studies, which involve stochastic optimization algorithms, we should find a way to decide if the statistical significance between their performance is also important for real-world applications. For this reason, we have recently proposed a practical Deep Statistical Comparison (pDSC). It takes into account practical significance when making a statistical comparison of meta-heuristic stochastic optimization algorithms for single-objective optimization problems. Experimental results showed that our recently proposed approach provided very promising results.

CCS CONCEPTS

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

KEYWORDS

stochastic optimization algorithms, practical significance

ACM Reference Format:
Tome Eftimov and Peter Korošec. 2020. Is the statistical significance between stochastic optimization algorithms' performances also significant in practice?. In *Genetic and Evolutionary Computation Conference Companion (GECCO '20 Companion), July 8–12, 2020, Cancún, Mexico.* ACM, New York, NY, USA, 2 pages. https://doi.org/10.1145/3377929.3397485

1 INTRODUCTION

Over the past years, one mandatory task that should be done in order to publish a newly developed stochastic optimization algorithms is to show that its performance is statistically significantly better than the performances of state-of-the-art-algorithms. For this reason, different statistical approaches are used and applied on the collected experimental data. However, when we are interested to transfer the learned knowledge to a real-world problem, or to select the algorithm that will give us a satisfying solution, the question that arises is "Are the differences among samples big enough to have also significant meaning in practice?". So instead of focusing only on a statistical significance, there is also a need to determine practical significance. Practical significance is defined as the relationships among the quality of solutions of real-world applications. For example, if we have two algorithms developed to find a global optimum for a given problem, and one of them solves the problem with an approximation error of 10^{-10} and the other with an error of 10^{-16}, statistical significance can be found, but this significance can be insignificant in a practical sense with respect to the application of the problem.

There are a lot of applications where the practical significance is relevant. Some of them are:

- Industrial tasks:
  - Production scheduling where simulations are based on predefined production norms;
  - Production where various products are developed with some tolerances that do not have big impact on their performance, which means that products within these tolerances are equal.
- Benchmarking comparisons made in the literature:
  - The influence of the computer accuracy due to the IEEE 754 standard;
  - Type of variables that are used (e.g., 4-byte float, 8-byte gloat, 10-byte float);
  - An error threshold that is used in competitions as a stopping criteria for the algorithms.

To address the problem of testing stochastic optimization algorithms for practical significance, we propose an extended version of our previously published approach for comparing with regard to statistical significance (i.e. Deep Statistical Comparison [2].) The new approach that deals with the practical significance is known as practical Deep Statistical Comparison [1]. It consists of two steps: i) the obtained results must first be preprocessed at some practical level ϵ, which is user-specified depending of the problem that is solved, and ii) the preprocessed data is used as an input data for a statistical test in order to see if there is a practical significance between the performances of the compared algorithms.

2 PRACTICAL DEEP STATISTICAL COMPARISON

The practical Deep Statistical Comparison (pDSC) is an extension of the Deep Statistical Comparison (DSC). Two variants of the pDSC ranking scheme are introduced, based on how the data is preprocessed with regard to the practical threshold, which is specified by the user and depends on the problem that is being solved.

In the first variant, called sequential pDSC, the preprocessing is made in a sequential order, in which the g-th run from one algorithm is compared with the g-th run of the other algorithm. In the case

GECCO '20 Companion, July 8–12, 2020, Cancún, Mexico

ACM ISBN 978-1-4503-7127-8/20/07.
https://doi.org/10.1145/3377929.3397485

when the absolute difference between their values (i.e. obtained solutions in the g-th run) is smaller or equal than the practical threshold that is specified, then both of the algorithms replace the g-th run value with an average of their values obtained in the g-th run. Otherwise, their values remain the same. This kind of preprocessing should be done using the multiple runs obtained for a given problem for every pairwise comparison. Next, the preprocessed data for each pairwise comparison is used to compare their distributions in order to define the p-value, which is stored in a matrix M that consists of the p-values for all pairwise comparisons.

The second variant of pDSC is known as Monte-Carlo pDSC. We introduced this variant because the sequential preprocessing of the independent runs can affect the practical significance, since the algorithms are stochastic in nature and there is no guarantee that the same order will be produced if the algorithms are run again. To avoid the dependence of the practical significance from the order of the independent runs and to provide a more robust comparison, each pairwise comparison is made using permutations of the independent runs of both algorithms. This simulates N runs (i.e. combination) of the algorithms on the same problem, where the final solutions are in a different order. Next, the data for every combination is preprocessed in the same way as in the sequential pDSC and their distributions are compared. Comparing N different combinations for each pairwise comparison results in N different p-values. To select a representative p-value for every pairwise comparison, a new random variable should be defined as the number of combinations where the null hypothesis is rejected, together with a prior level of significance α_p that gives us an estimation if the compared distributions are the same or not. If the distributions of the algorithms involved in the pairwise comparison are the same, then the p-value for this pairwise comparison can be randomly selected from a subset of N p-values which are greater than α. If the distributions are different, a kernel density estimation is used to estimate the probability density function of a subset of N p-values that are lower than α. The mode of the probability density function is used as an appropriate p-value, which will be used in the M matrix. In this case, the kernel density estimation and the mode are used, because if p-value is chosen at random, it can be further affected when it needs to be corrected to control the family-wise error rate. This is the probability of making one or more false discoveries, or type I errors, among all hypotheses when performing multiple hypotheses testing.

After calculating the matrix M, either using sequential or Monte-Carlo pDSC, which consists of the p-values for all pairwise comparisons, the next step is to check its transitivity.

If the transitivity is satisfied, we have a relation of equivalence, so we can split the algorithms into groups of equivalence, which means that there is no practical significance between the algorithms from the same group and they should obtain the same ranking.

If the transitivity is not satisfied, the algorithms are ranked by using some performance metric specified by the experimental design. The performance metric can be average, median, a combination of average and standard deviation, etc. In our case, we used an average, since it is an unbiased estimator.

The pDSC ranking scheme works on a single problem level. If we are performing a benchmarking scenario that involves multiple problems, it should be applied separately on the data obtained for every problem. After obtaining the pDSC rankings for each problem involved in the benchmark data set, they are further used as an input data for an appropriate omnibus statistical test in order to compare them with regard to practical significance.

We should also mention that in the cases when the practical significance threshold is set to zero, the pDSC approach becomes the DSC approach.

3 DISCUSSION AND CONCLUSION

To the best of our knowledge, there is one published approach known as a Chess Rating System for Evolutionary Algorithms (i.e. CRS4EAS) [3], which simulates a chess tournament where the optimization algorithms are considered as chess players and a competition between the results of two optimization algorithms as the outcome of a single game. It requires a draw limit (i.e. ϵ_d), where the result of the game is assumed as a draw. Each algorithm is treated as a player and is described by a rating R, rating deviation RD, and a rating confidence interval RI calculated with regard to the Glicko-2 rating system. If the RIs of two algorithms do not overlap, then the performances of the two algorithms differ significantly, in the opposite way we can not give any conclusion, since the RIs may overlap, but there still can be a practical significance between their performances. For calculating RIs, authors proposed using $RD = 50$ as an appropriate value. However, this is questionable and means that it can be tuned until we get the results that are in our interest. There exist studies where 95% RIs are calculated with different values for RD, without providing a relevant argument how to choose the RD value.

An evaluation of both practical Deep Statistical Comparison (pDSC) variants was made using the experimental results from the Black-Box Benchmarking 2015 competition, which uses single-objective problems for benchmarking, and comparing them to the Chess Rating System for Evolutionary Algorithms (CRS4EAs). Preprocessing for practical significance is carried out in a similar way, but there are cases when the results for practical significance differ. This happens because pDSC is inherently more robust against outliers. The pDSC variants are easily accessible through web services at https://ws.ijs.si:8443/dsc-1.4/documentation.pdf.

ACKNOWLEDGMENTS

This work was supported by the financial support from the Slovenian Research Agency (research core funding No. P2-0098 and project No. Z2-1867).

REFERENCES

[1] Tome Eftimov and Peter Korošec. 2019. Identifying practical significance through statistical comparison of meta-heuristic stochastic optimization algorithms. *Applied Soft Computing* (2019), 105862.

[2] 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.

[3] Niki Veček, Marjan Mernik, and Matej Črepinšek. 2014. A chess rating system for evolutionary algorithms: a new method for the comparison and ranking of evolutionary algorithms. *Information Sciences* 277 (2014), 656–679.