Warm Starting CMA-ES for Hyperparameter Optimization

Masahiro Nomura[*†1,2], **Shuhei Watanabe**[*‡3], **Youhei Akimoto**[4,5], **Yoshihiko Ozaki**[2,6], **Masaki Onishi**[2]

[1]CyberAgent, Inc.
[2]Artificial Intelligence Research Center, AIST.
[3]University of Freiburg.
[4]University of Tsukuba.
[5]RIKEN Center for Advanced Intelligence Project.
[6]GREE, Inc.

Abstract

Hyperparameter optimization (HPO), formulated as black-box optimization (BBO), is recognized as essential for automation and high performance of machine learning approaches. The CMA-ES is a promising BBO approach with a high degree of parallelism, and has been applied to HPO tasks, often under parallel implementation, and shown superior performance to other approaches including Bayesian optimization (BO). However, if the budget of hyperparameter evaluations is severely limited, which is often the case for end users who do not deserve parallel computing, the CMA-ES exhausts the budget without improving the performance due to its long adaptation phase, resulting in being outperformed by BO approaches. To address this issue, we propose to transfer prior knowledge on similar HPO tasks through the initialization of the CMA-ES, leading to significantly shortening the adaptation time. The knowledge transfer is designed based on the novel definition of task similarity, with which the correlation of the performance of the proposed approach is confirmed on synthetic problems. The proposed warm starting CMA-ES, called WS-CMA-ES, is applied to different HPO tasks where some prior knowledge is available, showing its superior performance over the original CMA-ES as well as BO approaches with or without using the prior knowledge.

1 Introduction

Hyperparameter optimization (HPO) is an essential for achieving effective performance in a wide range of machine learning algorithms (Feurer and Hutter 2019). HPO is formulated as a black-box optimization (BBO) problem because the objective function of the task of interest (referred to as the target task) cannot be described using an algebraic representation in general. One way to accelerate the optimization for HPO on the target task is to exploit results from a related task (referred to as the source task). This *transfer learning* setting on HPO often appears in practical situations and is actively studied in HPO literature (Vanschoren 2019).

The covariance matrix adaptation evolution strategy (CMA-ES) (Hansen and Ostermeier 2001; Hansen 2016) is one of the most powerful methods for BBO and has a high degree of parallelism. The CMA-ES facilitates optimization by updating the parameters of the multivariate Gaussian distribution (MGD). Subsequently, the CMA-ES samples candidate solutions, which can be evaluated in parallel, from the MGD. It has been applied widely in practice, including in HPO often under parallel evaluation settings (Loshchilov and Hutter 2016; Friedrichs and Igel 2005; Watanabe and Le Roux 2014). The CMA-ES is particularly useful for solving *difficult* BBO problems such as non-separable, ill-conditioned, and rugged problems (Rios and Sahinidis 2013); furthermore, it has shown the best performance among more than 100 optimization methods for various BBO problems (Loshchilov, Schoenauer, and Sebag 2013) with moderate to large evaluation budgets ($> 100\times$ the number of variables).

However, the CMA-ES does not necessarily outperform Bayesian optimization (BO) (Frazier 2018) in the context of HPO, in which the evaluation budget is severely limited (Loshchilov and Hutter 2016). This is because the CMA-ES requires a relatively long adaptation phase to sample solutions into promising regions, especially at the beginning of optimization. Thus, the CMA-ES has received much less attention in the context of HPO, despite the excellent performance verified empirically in BBO.

In this work, to address the inefficiency of the CMA-ES when the evaluation budget is severely limited, we introduce a simple and effective warm starting method WS-CMA-ES. This warm starting strategy can shorten the CMA-ES adaptation phase significantly by utilizing the relationship between source and target tasks.

We first define a promising distribution in the search space and task similarity. The proposed method is designed to perform successful warm starting when the defined task similarity between a source task and a target task is high. To warm-start the optimization, we estimate a promising distribution of the source task. The

[*]These authors contributed equally in this work.
[†]Corresponding author: nomura_masahiro@cyberagent.co.jp
[‡]The work was done at Artificial Intelligence Research Center, AIST.

mean vector and the covariance matrix that are the parameters of the MGD in the CMA-ES are initialized by minimizing the Kullback–Leibler (KL) divergence between the MGD and the promising distribution. In this study, we performed experiments on synthetic and HPO problems for several warm starting settings. In particular, we have compared the proposed method with the original CMA-ES, Gaussian process expected improvement (GP-EI) (Snoek, Larochelle, and Adams 2012), tree-structured Parzen estimator (TPE) (Bergstra et al. 2011), multi-task Bayesian optimization (MTBO) (Swersky, Snoek, and Adams 2013), and multi-task BOHAMIANN (MT-BOHAMIANN) (Springenberg et al. 2016).

In summary, the contributions of this work are as follows:

- We formally defined a promising distribution and task similarity to give us the insight required to design a warm starting strategy for the CMA-ES.
- We proposed a warm starting method called the WS-CMA-ES that speeds up HPO by reducing the adaptation phase of the CMA-ES.
- We demonstrated that the performance of WS-CMA-ES is correlated with the defined task similarity through numerical experiments.
- We verified by synthetic problems that the WS-CMA-ES is more effective than naive warm starting methods even when a source task and a target task are not very similar.
- We demonstrated that the WS-CMA-ES converges faster than the existing methods for HPO problems.

2 Background

In this study, we considered the following BBO problem: minimizing a black-box function $f : \mathcal{X} \to \mathbb{R}$ over a compact measurable subset $\mathcal{X} \subseteq \mathbb{R}^d$, where d is the number of variables. Let Λ_{Leb} be the Lebesgue measure on $\mathcal{X}$. In HPO, a solution $x \in \mathcal{X}$ corresponds to one hyperparameter setting, and $f(x)$ is generally a validation error of the trained model.

2.1 CMA-ES

The CMA-ES is a BBO method that uses an MGD $\mathcal{N}(m, \Sigma)$, wherein $m \in \mathbb{R}^d$, $\Sigma \in \mathbb{R}^{d\times d}$ is a positive-definite symmetric matrix. This algorithm generates λ solutions following the MGD and evaluates each solution in every iteration, which is defined as a generation. The mean vector m and the covariance matrix Σ are updated according to the ranking of the solutions in the latest generation, and the CMA-ES learns to sample solutions from the promising region. [1] The update of the CMA-ES is strongly related to the natural gradient descent (Akimoto et al. 2010; Ollivier et al. 2017); m and Σ in the CMA-ES are updated to decrease the expected evaluation value. For more details, see the CMA-ES tutorial (Hansen 2016).[2]

[1] Note that we followed the standard formulation of the CMA-ES. Therefore, Σ was decomposed into $\Sigma = \sigma^2 C$ where $\sigma > 0$ and $C \in \mathbb{R}^{d\times d}$.

[2] Among several versions of the CMA-ES, we use the recent standard version described in (Hansen 2016).

The CMA-ES is invariant with order-preserving transformations of the objective function because the CMA-ES uses only a ranking of solutions, not the evaluation value itself. In addition, the CMA-ES has the affine invariance to the search space. These invariances allow us to generalize the successful empirical results of the CMA-ES to a more wide range of problems (Hansen and Auger 2014).

2.2 CMA-ES for Hyperparameter Optimization

The invariances mentioned above make the CMA-ES suitable for HPO. For example, when transferring HPO to different *dataset* or different *objectives*, the scale of each objective may significantly vary. However, the CMA-ES is robust to such a heterogeneous scale owing to the use of rank, not the evaluation value itself. Further, hyperparameters are often dependent on each other, such as the batch size and the learning rate in deep neural networks (Keskar et al. 2017; Smith et al. 2018). The CMA-ES can address this dependency by learning the covariance matrix appropriately. Indeed, Loshchilov and Hutter (Loshchilov and Hutter 2016) reported that the CMA-ES outperformed BO in HPO when a moderate evaluation budget was available. However, the evaluation budget is often severely limited and is insufficient for the CMA-ES to adapt the covariance matrix, particularly for the end users whose computational resources are limited. In such cases, the CMA-ES does not yield better solutions than other approaches such as BO approaches (Loshchilov and Hutter 2016).

The possible reason for the lower performance of the CMA-ES is a long adaptation phase of the covariance matrix; we explain the reason below. The CMA-ES attempts to adapt the covariance matrix to approximate the shape of the level set of the objective function by that of MGD. In the case of a convex quadratic objective function, the covariance matrix approximates the inverse Hessian of the function. Once the covariance matrix is well-adapted, the CMA-ES exhibits a linear convergence, where the convergence speed is as high as the one for the spherical function, $f(x) = \|x\|^2$. However, as the degree of freedom of the covariance matrix is $\Theta(d^2)$, the learning rate for the covariance matrix update is set to $\Theta(1/d^2)$ by default for stability. Therefore, $\mathcal{O}(d^2)$ iterations are required to adapt the covariance matrix.

Two approaches can be considered to mitigate this problem. One is increasing λ, which is the number of solutions per iteration, and evaluating them in parallel. The number of iterations for the adaptation decreases as λ increases (Akimoto and Hansen 2020). However, this approach is useful only for those users who can afford parallel computational environments. The other approach is employing variants of the CMA-ES with a restricted covariance matrix model, such as the diagonal model (Ros and Hansen 2008). Because the covariance matrix model has few degrees of freedom, the learning rate can be set higher, thereby accelerating the adaptation, while compromising rotational invariance. Hence, we also propose another method which uses a restricted matrix model, in addition to the proposed method with the full covariance matrix.

3 Warm Starting CMA-ES

We consider a transfer HPO setting, where we have pairs (*hyperparameter, performance*) on a source task. This setting often appears in the practical use of HPO (Vanschoren 2019). The original CMA-ES and other variants aiming to mitigate the problem of the long adaptation phase, which are described in Section 2.2, do not have any mechanism to exploit such observational data on the source task.

In this work, we propose a simple and effective warm starting CMA-ES (WS-CMA-ES). First, we construct the definitions of a promising distribution and a task similarity. Next, the details of WS-CMA-ES are given.

3.1 Definition of Task Similarity

To define task similarity, it is necessary to identify which parts of the objective function characterize the task. Because the goal of optimization is to identify the best solution, one possible definition of a task feature is a promising distribution, which represents the regions wherein promising solutions exist with a higher probability. Herein, we define the γ-*promising distribution* as follows:

Definition 3.1 (γ-Promising Distribution)**.** *Suppose that $f : \mathcal{X} \to \mathbb{R}$ is a measurable function defined over the compact measurable subset $\mathcal{X} \subseteq \mathbb{R}^d$. For $\gamma \in (0, 1]$, let $\mathcal{F}^\gamma = \{x \in \mathcal{X} \mid f(x) \leq f^\gamma\}$, where f^γ is defined such that $\gamma = \Lambda_{\mathrm{Leb}}(\mathcal{F}^\gamma)/\Lambda_{\mathrm{Leb}}(\mathcal{X})$. We define γ-promising distribution P^γ, whose probability density function p^γ is defined as*

$$p^\gamma(x) = \frac{1}{Z} \int_{x' \in \mathcal{F}^\gamma} \exp\Big(-\frac{\|x - x'\|^2}{2\alpha^2}\Big) dx', \qquad (1)$$

where $Z = \int_{x \in \mathcal{X}} \int_{x' \in \mathcal{F}^\gamma} \exp\Big(-\frac{\|x-x'\|^2}{2\alpha^2}\Big) dx' dx$ and $\alpha \in \mathbb{R}^{>0}$ is a prior parameter.

Our definition of the γ-promising distribution is based on two HPO problem assumptions: (1) the continuity between hyperparameters and objective function and (2) the local convexity of a promising region.

The first assumption is the continuity of the objective function. When a hyperparameter varies slightly, its performance also changes to a small extent. More precisely, the continuity of the objective function leads to the smoothness of the promising distribution. Another possible definition for the promising distribution is a uniform distribution $\mathbf{1}\{x \in \mathcal{F}^\gamma\}/\Lambda_{\mathrm{Leb}}(\mathcal{F}^\gamma)$, where $\mathbf{1}$ is the indicator function. An advantage of the uniform distribution is its simplicity. However, the measure of the regions not within $\mathcal{F}^\gamma$ becomes 0 when the promising distribution is based on the uniform distribution. In other words, this distribution considers the regions in $\mathcal{F}^\gamma$ as promising to the same extent and considers the other regions totally unpromising. The main problem with the uniform distribution is that it ignores the importance of the proximity of the boundaries around $\mathcal{F}^\gamma$. In fact, the magnitudes of importance for the boundary regions should not fluctuate greatly depending on whether the regions are inside or beyond $\mathcal{F}^\gamma$. The promising distribution should measure such slight variations of importance over the entire search space. This condition requires the promising distribution to be smooth. Therefore, the uniform distribution, which does not satisfy the smoothness, is not suitable for defining the promising distribution.

The second assumption is related to the local convexity of the promising region. From the first assumption, the inside of the level set is likely to be more promising, and it naturally leads to the local convexity. The γ-promising distribution can represent these assumptions more appropriately than the uniform distribution $\mathbf{1}\{x \in \mathcal{F}^\gamma\}/\Lambda_{\mathrm{Leb}}(\mathcal{F}^\gamma)$.

Next, γ-*similarity*, which measures task similarity, is formulated using the γ-promising distribution as follows:

Definition 3.2 (γ-Similarity)**.** *Suppose that $f_1,\ f_2 : \mathcal{X} \to \mathbb{R}$ are measurable functions defined over the compact measurable subset $\mathcal{X} \subseteq \mathbb{R}^d$. Let $\gamma_1, \gamma_2 \in (0, 1]$ and let $P_i^{\gamma_i}$ be γ_i-promising distribution of f_i for $i = 1, 2$ defined in Definition 3.1. We define γ-similarity from f_1 to f_2 as*

$$s(\gamma_1, \gamma_2) := D_{\mathrm{KL}}(P_* || P_2^{\gamma_2}) - D_{\mathrm{KL}}(P_1^{\gamma_1} || P_2^{\gamma_2}), \qquad (2)$$

where $D_{\mathrm{KL}}(P||Q)$ is the KL divergence between P and Q, and P_* is a non-informative prior distribution.

A non-informative prior distribution is used when no information is available on the objective function. In the CMA-ES, for the search space $\mathcal{X} = [0, 1]^d$, $\mathcal{N}(0.5, 0.2^2)$ is given as an initial distribution for each variable (Loshchilov and Hutter 2016). BO uses a uniform distribution as a non-informative prior distribution in many cases.

Intuitively, if $s(\gamma_1, \gamma_2) > 0$, i.e., $D_{\mathrm{KL}}(P_1^{\gamma_1} || P_2^{\gamma_2}) < D_{\mathrm{KL}}(P_* || P_2^{\gamma_2})$, the promising distribution $P_1^{\gamma_1}$ for task 1 is closer to the promising distribution $P_2^{\gamma_2}$ for task 2 than a non-informative prior distribution P_*.

3.2 Algorithm Construction

We assume a source task (task 1) is similar to a target task (task 2). Hence, the γ-similarity holds $s(\gamma_1, \gamma_2) > 0$, i.e., $D_{\mathrm{KL}}(P_1^{\gamma_1} || P_2^{\gamma_2}) < D_{\mathrm{KL}}(P_* || P_2^{\gamma_2})$. Note that the non-informative prior distribution P_* is used as an initial distribution for the CMA-ES if there is no information on the source task. Because we assume knowledge transfer from the source task, we obtain $D_{\mathrm{KL}}(P_1^{\gamma_1} || P_2^{\gamma_2}) < D_{\mathrm{KL}}(P_* || P_2^{\gamma_2})$. Therefore, the CMA-ES can begin optimization from the location close to the promising region by exploiting the information for the promising region of the source task from $P_1^{\gamma_1}$ instead of P_*. In this study, the initial parameters of the MGD were estimated by minimizing the KL divergence between the MGD and the empirical version of $P_1^{\gamma_1}$. The empirical version of $P_1^{\gamma_1}$ uses Gaussian mixture models (GMM) as shown in Eq. (3).

This method transfers prior knowledge as follows. First, the top $\gamma \times 100\%$ solutions are selected from a set of solutions on a source task. Second, a GMM, i.e., a promising distribution of the source task, is built using the solutions selected above. Finally, the parameters of the MGD are initialized via the approximation of the GMM. Further details of each operation are described in the next section.

3.3 Details of Each Operation

Estimation of a Promising Distribution of a Source Task
Let N be the number of observations in a source task. Based

on the definition of γ-promising distribution, a probability density function $p(x)$ that represents a promising distribution of the source task is estimated using the top $\gamma \times 100$ % solutions as follows:

$$p(x) = \frac{1}{N_\gamma} \sum_{i=1}^{N_\gamma} \mathcal{N}(x \mid x_i, \alpha^2 I), \tag{3}$$

where $\alpha \in \mathbb{R}^{>0}$ is a prior parameter, $I \in \mathbb{R}^{d \times d}$ is the identity matrix, $N_\gamma = \lfloor \gamma \cdot N \rfloor$, and x_i, which is an observation in the source task, is sorted so that $f(x_1) \leq f(x_2) \leq \cdots \leq f(x_{N_\gamma}) \leq \cdots \leq f(x_N)$. The robustness of these two parameters, γ and α, is shown in Appendix.

Transferring Prior Knowledge to the CMA-ES Based on the aforementioned promising distribution definition, we introduced the estimation method for the initial parameters of the MGD for the CMA-ES. The initial parameters were determined by minimizing the KL divergence between the promising distribution $p(x)$ and the MGD $q(x) = \mathcal{N}(x \mid m, \Sigma)$.

Based on Theorem 3.2 and Eqs. (2)–(4) of (Runnalls 2007), we can easily identify the parameters that minimize the KL divergence as follows:

$$m^* = \frac{1}{N_\gamma} \sum_{i=1}^{N_\gamma} x_i, \tag{4}$$

$$\Sigma^* = \alpha^2 I + \frac{1}{N_\gamma} \sum_{i=1}^{N_\gamma} (x_i - m^*)(x_i - m^*)^\top. \tag{5}$$

We can observe that Eq. (5) agrees with the formula for the maximum a posteriori estimation (Bishop 2006), which implies that the first term in Eq. (5) has the effect of the regularization. We can also derive a variant that restricts the covariance matrix to a diagonal: $\Sigma^* = \mathrm{diag}(l_1, \cdots, l_d)$. For $j \in \{1, \cdots, d\}$, it can be easily calculated as follows, considering the independence of the variables:

$$l_j = \alpha^2 + \frac{1}{N_\gamma} \sum_{i=1}^{N_\gamma} ([x_i]_j - [m^*]_j)^2, \tag{6}$$

where $[x]_j$ denotes the j-th element of the vector x. We call this restricted variant WS-sep-CMA-ES; its performance is validated in Section 4.2.

4 Experiments on Synthetic Problems

4.1 Performance Depending on Task Similarity

As defined in the previous section, WS-CMA-ES is expected to achieve faster convergence on problems with higher γ-similarity (i.e. $s(\gamma_1, \gamma_2) > 0$). To confirm this correlation, we measured the γ-similarity and the performance of WS-CMA-ES using two synthetic problems:

- **Sphere Function**: $f(x) = (x_1 - b)^2 + (x_2 - b)^2$
- **Rotated Ellipsoid Function**: $f(x) = f_{\mathrm{ell}}(Rx)$

where $f_{\mathrm{ell}}(x) = (x_1 - b)^2 + 5^2(x_2 - b)^2$, $R \in \mathbb{R}^{2 \times 2}$ is a rotation matrix rotating $\pi/6$ around the origin, and b is the coefficient for each problem. The sphere function is a simple problem. On the other hand, the characteristics of the rotated ellipsoid function are non-separable and ill-conditioned. Non-separable characteristic is related to the dependencies between the variables, and the ill-conditioned characteristic is that the contribution to the objective function varies widely depending on each variable.

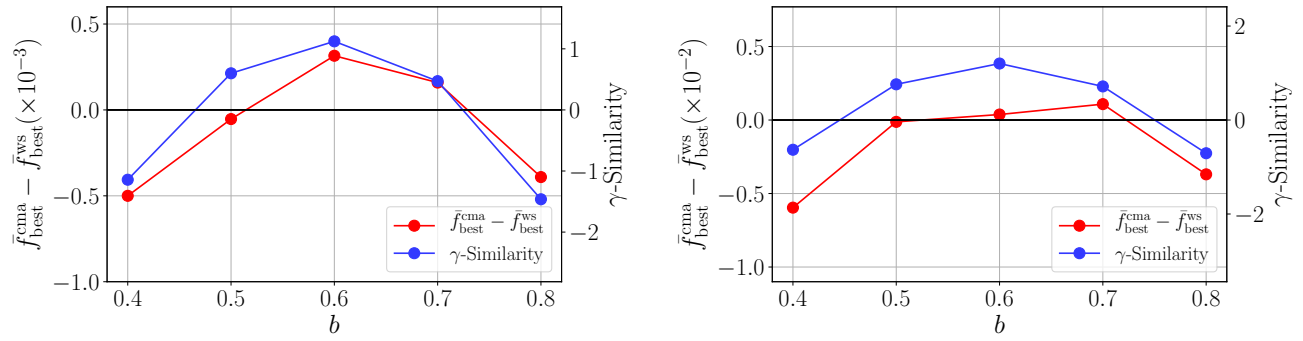
We optimized each synthetic problem using two methods: the WS-CMA-ES and the original CMA-ES. The target task for each problem is the function with a coefficient $b = 0.6$. As prior knowledge for both settings, we evaluated each function with a coefficient from $b = 0.4, \cdots, 0.8$ in increments of 0.1. In other words, we optimized the synthetic problems by the WS-CMA-ES using each prior knowledge for each case (for $b = 0.4, 0.5, \cdots, 0.8$). Each optimization was performed 20 times. We employed $\mathcal{N}(0.5, 0.2^2)$, which is a non-informative distribution used in Definition 3.2, as an initial distribution for each variable in the CMA-ES. In all the experiments, α and γ in WS-CMA-ES were set to 0.1 for each. For more details, see Appendices A and B.

The results are presented in Figure 1. To visualize the correlation between the performance improvement and the γ-similarity, we measured the γ-similarity between the cases of $b = 0.4, \cdots, 0.8$ and $b = 0.6$ and plotted it along with the results. In both settings, the variation of the γ-similarity almost corresponds to that of the improvement achieved by WS-CMA-ES compared with the CMA-ES with respect to the value b. In brief, WS-CMA-ES successfully transferred prior knowledge when a source task resembled the target task in terms of γ-similarity; in contrast, when the task similarity was low, the WS-CMA-ES did not perform well.

4.2 When Naïve Transfer Fails

If we know in advance that the source and target task is similar enough, transferring the knowledge of the source task is relatively easy. For example, one intuitive and naïve method, in this case, is to sample a solution near the solution with good performance in the source task. Alternatively, if the CMA-ES is performed for the source task, we can reuse the final MGD obtained on the source task as the initial MGD for the target task. The assumption that the tasks are similar is reasonable in practical cases (Vanschoren 2019). However, it is difficult to guarantee it before performing optimization. Therefore, it is desirable to alleviate dramatic performance degradation even when these tasks are not very similar. To confirm the robustness of our proposed warm starting method in such situations, we compare the behavior of the proposed method with the following naïve transferring methods:

- *ReuseGMM* : This method samples solutions from the GMM which represents a promising distribution estimated on the source task; that is, the solutions are sampled from the distribution defined in Eq. (3) throughout the optimization.
- *ReuseNormal* : This method uses the final mean and covariance matrix obtained on the source task as the initial MGD on the target task. This method is the same as the (WS-)CMA-ES except for the initialization of MGD.



(a) Sphere Function

(b) Rotated Ellipsoid Function

Figure 1: Results of the experiments to confirm the correlation between γ-similarity and performance. The horizontal axis represents the prior parameter b used for each warm starting setting where the prior knowledge was the result with $b = 0.6$. The vertical axis for red and blue lines denote the subtraction $\bar{f}^{\mathrm{cma}}_{\mathrm{best}} - \bar{f}^{\mathrm{ws}}_{\mathrm{best}}$ of the mean of the best evaluation value in the CMA-ES and the WS-CMA-ES (20 runs for each) and γ-similarity in Definition 3.2, respectively. $\bar{f}^{\mathrm{cma}}_{\mathrm{best}} - \bar{f}^{\mathrm{ws}}_{\mathrm{best}} > 0$ implies that the result of the WS-CMA-ES is better than that of the CMA-ES.

Random search is used as the optimization of a source task for all methods except for ReuseNormal; in ReuseNormal, the result of the CMA-ES is used as the source task. We consider the sphere function and the rotated ellipsoid function defined in Section 4.1; the experimental settings remain the same.

In addition to these transferring methods, we experiment with the CMA-ES and the sep-CMA-ES, which are not transferring methods, as references. When the offset changes largely between the source and target tasks, these non-transferring methods become advantageous, as shown in Section 4.

Figure 2 presents the results of the experiments over 20 runs. As expected, ReuseNormal shows the best performance on offset $b = 0.6$ where the source and target tasks are the same. However, the performance of ReuseNormal deteriorates drastically when the offset is changed. This is because ReuseNormal converges more than necessary near the optimal solution of the source task even when the optimal solution of the target task is largely different. In this case, it takes significant time to move from the promising region estimated by the source task, which impairs the performance of ReuseNormal. In contrast, the proposed methods, WS-CMA-ES and WS-sep-CMA-ES, are less dependent on how long the optimization is performed on the source task, which leads to relatively less performance degradation even in such cases. Similar to the case of ReuseNormal, ReuseGMM, which does not adapt during optimization, is strongly affected by the dissimilarity of the tasks. This demonstrates the necessity of the adaptation toward the optimal solution direction of the target task by the CMA-ES (or sep-CMA-ES). In conclusion, compared with the naïve transferring methods, which strongly assume that the tasks are similar, the proposed method is more robust and efficient to the difference between the source and target tasks.

5 Experiments for Hyperparameter Optimization

We applied WS-CMA-ES to several HPO problems to verify its effectiveness on HPO. The experiments comprise the following two practical scenarios:

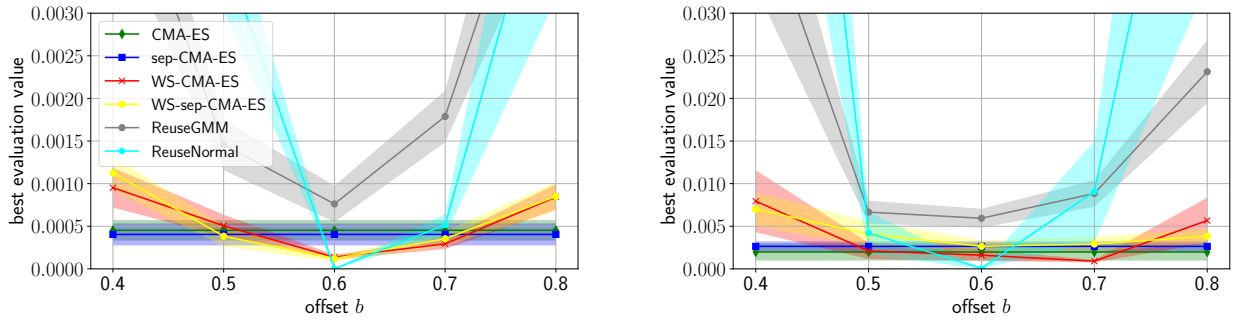
- Warm starting using a result of HPO for a subset of a dataset (Section 5.1), and
- Warm starting using a result of HPO for another dataset (Section 5.2).

As the baseline methods, we select the (1) CMA-ES, (2) random search (RS) (Bergstra and Bengio 2012), (3) random sampling from the initial MGD used in WS-CMA-ES (WS-only), (4) GP-EI (Snoek, Larochelle, and Adams 2012), (5) TPE (Bergstra et al. 2011), (6) MTBO (Swersky, Snoek, and Adams 2013), and (7) MT-BOHAMIANN (Springenberg et al. 2016). MTBO, which is an extension of GP-EI, and MT-BOHAMIANN are warm starting methods for BO. TPE is known to provide strong performance in HPO. Note that we do not use WS-sep-CMA-ES because the performance is similar to WS-CMA-ES in the severely limited budget setting, which is confirmed in Section 4.2. We evaluated 100 hyperparameter settings by RS as prior knowledge in all the experiments to allow every method to transfer the same data fairly. Each optimization was run 12 times. Details of the experimental settings are shown in Appendix.

5.1 Warm Starting using a Result of a Subset

We evaluated hyperparameter settings of each machine learning algorithm trained on 10% of a full dataset. This result was considered as the source task and was used by the warm starting methods.

LightGBM on Multilabel Classification LightGBM (Ke et al. 2017) is used as an ML model. Six hyperparameters shown in Appendix were optimized in the experiments. We



(a) Sphere Function

(b) Rotated Ellipsoid Function

Figure 2: Comparing the proposed methods with naive transferring methods. The mean (line) and the standard error (shadow) over 20 runs are shown. The results of the CMA-ES and the sep-CMA-ES, which are not transferring methods, are included as references.

used the Toxic Comment Classification Challenge data[3] as a dataset. As a metric in the experiments, the mean column-wise area under the receiving operating characteristic curve (ROC AUC) was used. Note that this measurement is better when the value is higher, so we used 1 – AUC as the objective function.

MLP on MNIST and Fashion-MNIST The proposed method was applied to the HPO of multilayer perceptrons (MLPs). We used the MNIST handwritten digits dataset (LeCun et al. 1998) and the Fashion-MNIST clothing articles dataset (Xiao, Rasul, and Vollgraf 2017). We optimized eight hyperparameters as shown in Appendix.

CNN on CIFAR-100 We further applied the proposed method to more sophisticated 8-layer convolutional neural networks (CNNs). The CNNs were trained on the CIFAR-100 dataset (Krizhevsky 2009) and have ten types of hyperparameters as described in Appendix.

Results and Discussion on Knowledge Transfer of a Subset Figure 3 shows the experiment results. In each experiment, the proposed method and the WS-only identified better objective metrics much faster than the CMA-ES did. Further, we found that MTBO yielded better solutions quickly than GP-EI. Clearly, there was high task similarity between the given tasks that could be exploited by warm starting methods. WS-CMA-ES and WS-only found better hyperparameter settings in the earlier stage of the optimizations than the others. In the later stage of the optimization, WS-CMA-ES adapted successfully and converged to better solutions than that of WS-only. Figure 3 (a) shows that WS-CMA-ES and WS-only behave similarly. This is because the evaluation budget is quite limited, and the update of MGD only happens a few times.

To observe the correlation between the performance of the WS-CMA-ES and a subset ratio, we applied WS-CMA-ES using prior knowledge of MNIST and Fashion-MNIST of different subset ratios $2\%, 10\%,$ and 50%. Figure 4 shows the results of the experiments. We observe that a higher subset ratio tends to result in faster convergence. The results imply that the sets of relatively good hyperparameters in the source tasks are spatially closer to those in the target tasks, while the best values may not be close, as is claimed in (Swersky, Snoek, and Adams 2013).

[3]https://www.kaggle.com/c/jigsaw-toxic-comment-classification-challenge

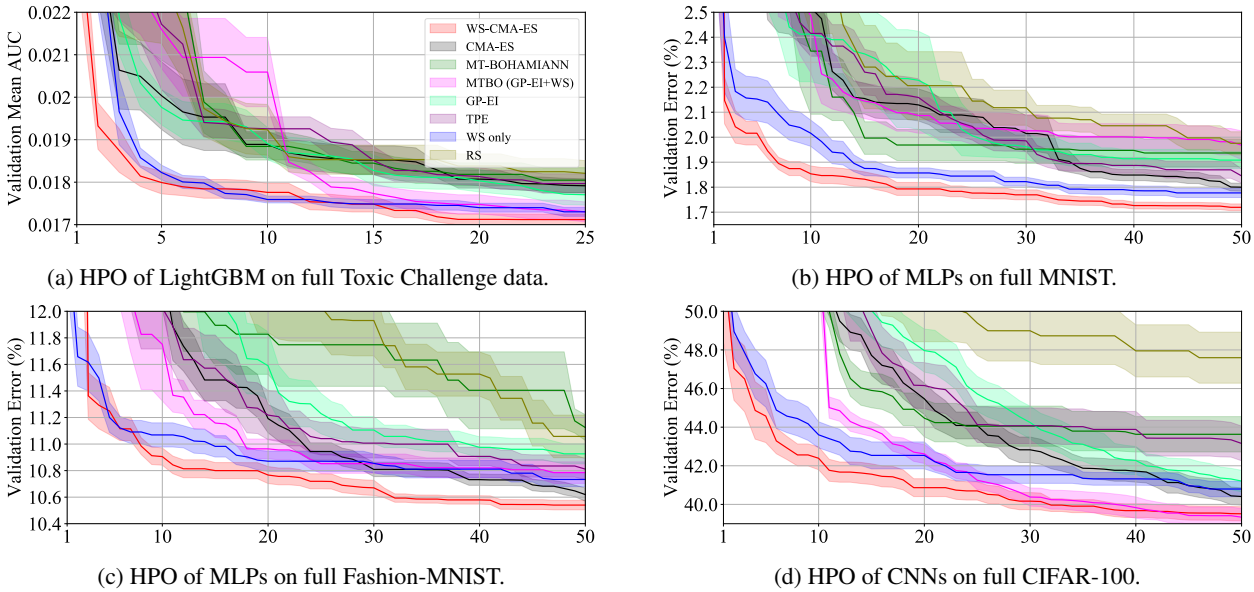
5.2 Warm Starting using a Result of Another Dataset

This section examines what happens when prior knowledge of a different dataset is utilized by warm starting methods. We carried out experiments to demonstrate the effectiveness of the proposed method in such a practical situation.

Using the Knowledge of MLP on MNIST for MLP on Fashion-MNIST We first trained MLPs on MNIST and then transferred the result to the HPO of Fashion-MNIST. The architecture of the MLPs and their hyperparameters are the same as those described in Section 5.1.

Using the Knowledge of CNN on SVHN for CNN on CIFAR-10 We optimized the 8-layer CNNs. Hyperparameters for this model are the same as those of the model optimized earlier (see Section 5.1). CNNs initially learned the Street View House Numbers (SVHN) dataset (Netzer et al. 2011). Next, the warm starting methods employed the knowledge to obtain the optimal hyperparameter settings for CNNs trained on CIFAR-10.

Results and Discussion on the Knowledge Transfer of Another dataset Figure 5 shows the results of the experiments. The proposed method exhibited outstanding convergence speed in the experiments and found better hyperparameter settings far more quickly than the CMA-ES. Although MTBO also successfully found better solutions than GP-EI, the performance of MTBO was not considerably better than that of RS. In fact, MTBO required approximately 25 evaluations to find better hyperparameter settings than GP-EI in the results described in Figure 3 (b), (d). According to Figure 5, however, it required approximately 40 evaluations in these experiments. Contrarily, the WS-CMA-ES identified better hyperparameter settings than the CMA-ES



(a) HPO of LightGBM on full Toxic Challenge data.

(b) HPO of MLPs on full MNIST.

(c) HPO of MLPs on full Fashion-MNIST.

(d) HPO of CNNs on full CIFAR-100.

Figure 3: Experiments with warm starting optimization using a result of the HPO for a subset of each dataset. Warm starting methods used a result of the HPO on $1/10^{\text{th}}$ of each dataset as prior knowledge. The horizontal axis represents the number of evaluations. We plotted the mean and the standard error of the best evaluation value over 12 runs.

in approximately 25 and 30 evaluations in the experiments using a small dataset and experiments using another dataset, respectively. This is probably because knowledge transfer from other datasets is more difficult than knowledge transfer from a subset of a dataset. MTBO obtains promising solutions using the approximation of the entire search space, but the WS-CMA-ES obtains promising solutions using that of only the promising region. The former approximation requires more observations to yield promising solutions compared with the latter. This may be the reason for the effectiveness of WS-CMA-ES in knowledge transfer from another dataset. This behavior can also be confirmed with the transfer HPO experiments with other datasets, which are provided in Appendix.

6 Related Work and Discussion

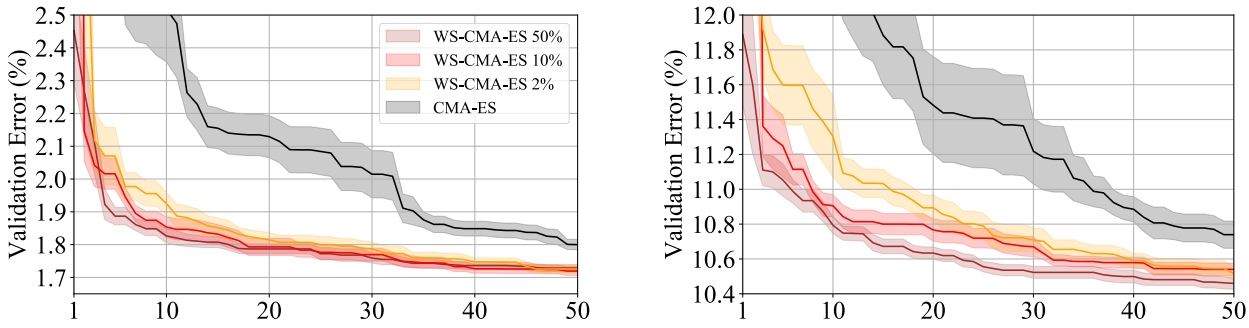
Various types of warm starting methods for BO have been actively studied in the HPO context. These methods model the relationship between tasks using a variety of ways, such as a Gaussian process (Swersky, Snoek, and Adams 2013; Poloczek, Wang, and Frazier 2017; Feurer, Letham, and Bakshy 2018), deep neural networks (Springenberg et al. 2016; Kim, Kim, and Choi 2017), and Bayesian linear regression (Perrone et al. 2018). However, the CMA-ES, which shows outstanding performance in BBO, has not been thoroughly considered in HPO.

One difference between our method and the warm starting methods for BO is in the usage of the source tasks' result. Most warm starting methods for BO repeatedly construct a probabilistic model using prior knowledge in each iteration. In contrast, WS-CMA-ES uses prior knowledge only at the inception of optimization. Therefore, the computational complexity of WS-CMA-ES does not depend on the number of observations. This enables users to implement the method even when numerous results are available. Although the meta-feature based warm starting (Feurer, Springenberg, and Hutter 2015) can alleviate this computational problem, it is not always possible to prepare such meta-feature for the dataset. The method of initializing the search space using the result of the source task does not incur extra computational complexity and can be used without such a meta feature (Wistuba, Schilling, and Schmidt-Thieme 2015; Perrone et al. 2019).

Another difference is that it is usually challenging for most BO approaches to handle the scale variation of objective functions across tasks. This situation often appears when exploiting prior knowledge in transfer HPO; for example, the validation error may significantly change across different datasets. This situation also appears when transferring between different objectives, such as transferring between the result of misclassification error and that of cross entropy. Salinas et al. introduced a sophisticated semi-parametric approach to deal with such a heterogeneous scale (Salinas, Shen, and Perrone 2020).

7 Conclusion and Future Work

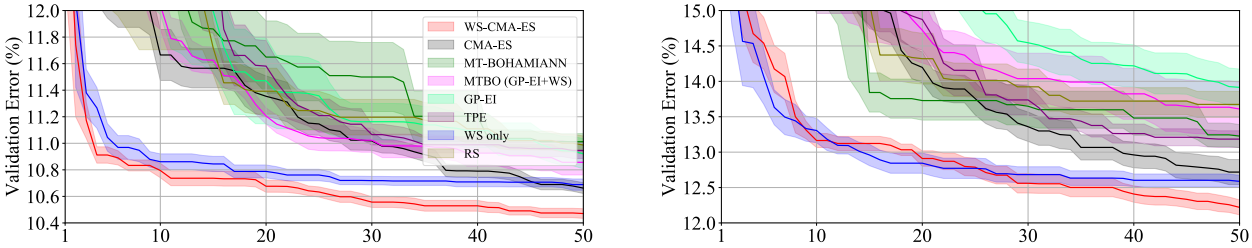
We proposed the WS-CMA-ES, a simple and effective warm starting strategy for the CMA-ES. The proposed method was designed based on the theoretical definitions of a promising distribution and task similarity. It initializes MGD in the CMA-ES by approximating the promising distribution on a source task. This knowledge transfer performs well especially when a target task is similar to a source task in terms of the defined task similarity, which is confirmed by our ex-



(a) MLPs trained on full MNIST

(b) MLPs trained on full Fashion-MNIST

Figure 4: Relationship between task similarity and the performance of WS-CMA-ES over 12 times. As the size of the dataset approaches that of the complete dataset, the WS-CMA-ES attains faster convergence.



(a) MLPs trained on Fashion-MNIST. As prior knowledge, the result of HPO of MLPs trained on MMIST was used.

(b) CNNs trained on CIFAR-10. As prior knowledge, the result of HPO of CNNs trained on SVHN was used.

Figure 5: Experiments with warm starting optimizations using the result of HPO on another dataset.

periments. Experiments with synthetic and HPO problems confirm that WS-CMA-ES is effective, even with low budgets or when the source and target tasks are not very similar.

The main limitation of this study is the assumption of task similarity. From our experiments and the desirable results of warm starting methods that assume task similarity (e.g., (Bardenet et al. 2013; Yogatama and Mann 2014)), we hypothesize that HPO tasks are often similar as long as so are they intuitively. However, WS-CMA-ES can be worse than the CMA-ES when the similarity between the source and target tasks is low, as shown in Figure 1. Automatic detection of task dissimilarity and switching back to the original CMA-ES is essential for this method to be more convincing and reliable.

Acknowledgements

The authors thank Shota Yasui, Yuki Tanigaki, Yoshiaki Bando for valuable feedback and suggestion. This paper is based on the results obtained from a project commissioned by the New Energy and Industrial Technology Development Organization (NEDO). Computational resource of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST) was used.

References

Akimoto, Y.; and Hansen, N. 2020. Diagonal Acceleration for Covariance Matrix Adaptation Evolution Strategies. *Evolutionary computation* 28(3): 405–435.

Akimoto, Y.; Nagata, Y.; Ono, I.; and Kobayashi, S. 2010. Bidirectional Relation between CMA Evolution Strategies and Natural Evolution Strategies. In *International Conference on Parallel Problem Solving from Nature*, 154–163.

Bardenet, R.; Brendel, M.; Kégl, B.; and Sebag, M. 2013. Collaborative hyperparameter tuning. In *International conference on machine learning*, 199–207.

Bergstra, J.; and Bengio, Y. 2012. Random Search for Hyper-Parameter Optimization. *Journal of Machine Learning Research* 13(Feb): 281–305.

Bergstra, J. S.; Bardenet, R.; Bengio, Y.; and Kégl, B. 2011. Algorithms for Hyper-Parameter Optimization. In *Advances in neural information processing systems*, 2546–2554.

Bishop, C. M. 2006. *Pattern recognition and machine learning*. springer.

Feurer, M.; and Hutter, F. 2019. Hyperparameter Optimization. In *Automated Machine Learning*, 3–33.

Feurer, M.; Letham, B.; and Bakshy, E. 2018. Scalable Meta-Learning for Bayesian Optimization using Ranking-Weighted Gaussian Process Ensembles. In *AutoML Workshop at ICML*.

Feurer, M.; Springenberg, J. T.; and Hutter, F. 2015. Initializing Bayesian Hyperparameter Optimization via Meta-learning. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 1128–1135.

Frazier, P. I. 2018. A Tutorial on Bayesian Optimization. *arXiv preprint arXiv:1807.02811* .

Friedrichs, F.; and Igel, C. 2005. Evolutionary Tuning of Multiple SVM Parameters. *Neurocomputing* 64: 107–117.

Hansen, N. 2016. The CMA Evolution Strategy: A Tutorial. *arXiv preprint arXiv:1604.00772* .

Hansen, N.; and Auger, A. 2014. Principled Design of Continuous Stochastic Search: From Theory to Practice. In *Theory and principled methods for the design of metaheuristics*, 145–180.

Hansen, N.; and Ostermeier, A. 2001. Completely Derandomized Self-Adaptation in Evolution Strategies. *Evolutionary computation* 9(2): 159–195.

Ke, G.; Meng, Q.; Finley, T.; Wang, T.; Chen, W.; Ma, W.; Ye, Q.; and Liu, T.-Y. 2017. LightGBM: A Highly Efficient Gradient Boosting Decision Tree. In *Advances in neural information processing systems*, 3146–3154.

Keskar, N. S.; Mudigere, D.; Nocedal, J.; Smelyanskiy, M.; and Tang, P. T. P. 2017. On Large-Batch Training for Deep Learning: Generalization Gap and Sharp Minima. *International Conference on Learning Representations* .

Kim, J.; Kim, S.; and Choi, S. 2017. Learning to Warm-Start Bayesian Hyperparameter Optimization. *arXiv preprint arXiv:1710.06219* .

Krizhevsky, A. 2009. Learning Multiple Layers of Features from Tiny Images. *Master's thesis, University of Tront* .

LeCun, Y.; Bottou, L.; Bengio, Y.; Haffner, P.; et al. 1998. Gradient-Based Learning Applied to Document Recognition. *Proceedings of the IEEE* 86(11): 2278–2324.

Loshchilov, I.; and Hutter, F. 2016. CMA-ES for Hyperparameter Optimization of Deep Neural Networks. In *ICLR Workshop*.

Loshchilov, I.; Schoenauer, M.; and Sebag, M. 2013. Bi-population CMA-ES Algorithms with Surrogate Models and Line Searches. In *Proceedings of the 15th annual conference companion on Genetic and evolutionary computation*, 1177–1184.

Netzer, Y.; Wang, T.; Coates, A.; Bissacco, A.; Wu, B.; and Ng, A. Y. 2011. Reading Digits in Natural Images with Unsupervised Feature Learning. In *NIPS Workshop on Deep Learning and Unsupervised Feature Learning 2011*.

Ollivier, Y.; Arnold, L.; Auger, A.; and Hansen, N. 2017. Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles. *The Journal of Machine Learning Research* 18(1): 564–628.

Perrone, V.; Jenatton, R.; Seeger, M. W.; and Archambeau, C. 2018. Scalable Hyperparameter Transfer Learning. In *Advances in Neural Information Processing Systems*, 6845–6855.

Perrone, V.; Shen, H.; Seeger, M. W.; Archambeau, C.; and Jenatton, R. 2019. Learning search spaces for Bayesian optimization: Another view of hyperparameter transfer learning. In *Advances in Neural Information Processing Systems*, 12751–12761.

Poloczek, M.; Wang, J.; and Frazier, P. 2017. Multi-Information Source Optimization. In *Advances in Neural Information Processing Systems*, 4288–4298.

Rios, L. M.; and Sahinidis, N. V. 2013. Derivative-free optimization: A review of algorithms and comparison of software implementations. *Journal of Global Optimization* 56(3): 1247–1293.

Ros, R.; and Hansen, N. 2008. A Simple Modification in CMA-ES Achieving Linear Time and Space Complexity. In *International Conference on Parallel Problem Solving from Nature*, 296–305.

Runnalls, A. R. 2007. A Kullback-Leibler Approach to Gaussian Mixture Reduction. *IEEE Transactions on Aerospace and Electronic Systems* 43(3): 989–999.

Salinas, D.; Shen, H.; and Perrone, V. 2020. A Quantile-based Approach for Hyperparameter Transfer Learning. In *International conference on machine learning*, 7706–7716.

Smith, S. L.; Kindermans, P.-J.; Ying, C.; and Le, Q. V. 2018. Don't Decay the Learning Rate, Increase the Batch Size. In *International Conference on Learning Representations*.

Snoek, J.; Larochelle, H.; and Adams, R. P. 2012. Practical Bayesian Optimization of Machine Learning Algorithms. In *Advances in neural information processing systems*, 2951–2959.

Springenberg, J. T.; Klein, A.; Falkner, S.; and Hutter, F. 2016. Bayesian Optimization with Robust Bayesian Neural Networks. In *Advances in Neural Information Processing Systems*, 4134–4142.

Swersky, K.; Snoek, J.; and Adams, R. P. 2013. Multi-Task Bayesian Optimization. In *Advances in neural information processing systems*, 2004–2012.

Vanschoren, J. 2019. Meta-Learning. In *Automated Machine Learning*, 35–61.

Watanabe, S.; and Le Roux, J. 2014. Black box optimization for automatic speech recognition. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 3256–3260.

Wistuba, M.; Schilling, N.; and Schmidt-Thieme, L. 2015. Hyperparameter Search Space Pruning–A New Component for Sequential Model-Based Hyperparameter Optimization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, 104–119.

Xiao, H.; Rasul, K.; and Vollgraf, R. 2017. Fashion-MNIST: a Novel Image Dataset for Benchmarking Machine Learning Algorithms. *arXiv preprint arXiv:1708.07747* .

Yogatama, D.; and Mann, G. 2014. Efficient Transfer Learning Method for Automatic Hyperparameter Tuning. In *Artificial intelligence and statistics*, 1077–1085.