Multi-Objective Multi-Fidelity Hyperparameter Optimization with Application to Fairness

Robin Schmucker[*]
Machine Learning Department
Carnegie Mellon University
rschmuck@cs.cmu.edu

Michele Donini
Amazon
Berlin, Germany
donini@amazon.com

Valerio Perrone
Amazon
Berlin, Germany
vperrone@amazon.com

Muhammad Bilal Zafar
Amazon
Berlin, Germany
zafamuh@amazon.com

Cédric Archambeau
Amazon
Berlin, Germany
cedrica@amazon.com

Abstract

In many real-world applications, the performance of machine learning models is evaluated not along a single objective, but across multiple, potentially competing ones. For instance, for a model deciding whether to grant or deny loans, it is critical to make sure decisions are fair and not only accurate. As it is often infeasible to find a single model performing best across *all* objectives, practitioners are forced to find a trade-off between the individual objectives. While several multi-objective optimization (MO) techniques have been proposed in the machine learning literature (and beyond), little effort has been put towards using MO for hyperparameter optimization (HPO) problems; a task that has gained immense relevance and adoption in recent years. In this paper, we evaluate the suitability of existing MO algorithms for HPO and propose a novel multi-fidelity method for this problem. We evaluate our approach on public datasets with a special emphasis on fairness-motivated applications, and report substantially lower wall-clock times when approximating Pareto frontiers compared to the state-of-the-art.

1 Introduction

Tuning complex machine learning (ML) models such as deep neural networks (DNNs) can be a time-consuming task even for expert practitioners. In recent years, automated hyperparameter optimization (HPO) techniques have become a popular and effective tool for finding models with maximal predictive accuracy in a sample-efficient manner. However, in several real-world domains, accuracy is not the only objective of interest, and the model must be simultaneously optimized w.r.t. one or more additional objectives such as fairness, interpretability, privacy, and number of FLOPs (e.g., for deployment on resource-restricted environments such as embedded devices).

As many of these objectives are often in direct contention with accuracy, it is unlikely to find a single model that maximizes accuracy while also providing optimal performance w.r.t. the other objectives. For instance, a plethora of recent studies have found that the models maximizing accuracy can also amplify historical biases in the data, leading to a high degree of unfairness in the

[*] Work done while interning at Amazon, Berlin, Germany.

34th Conference on Neural Information Processing Systems (NeurIPS 2020), Vancouver, Canada.

outcomes [2, 3, 8, 9, 10]. Similarly, training a DNN to be highly accurate may require increasing the number of parameters, which in turn leads to reduced interpretability.

In this paper, we cast the problem of simultaneously optimizing competing objectives as that of hyperparameter search with multi-objective optimization (MO). Given a list of the objectives to be optimized, our goal is to find the Pareto frontier to help the practitioners select a model suitable to their needs. While there has been some work on using MO, specifically, multi-objective Bayesian optimization (MBO) for hyperparameter search, these prior studies have been limited to a narrow range of tasks, that is, finding accurate DNNs that also takes a short time to make predictions [22]. Furthermore, experimental evaluations of MBO techniques are often limited to a small number of objectives (typically no more than three) and are focused on artificial functions.

In this paper, we build upon the Hyperband algorithm [35] to propose a novel, time-efficient multi-objective multi-fidelity HPO approach. Our algorithm leverages early stopping and parallel computing, and yields significant performance gains over existing MBO techniques.

To summarize our main contributions: (1) we revisit existing MBO techniques and perform the first large-scale evaluation of existing approaches on HPO problems, showing a surprisingly limited performance difference between existing methods; (2) we introduce the first multi-fidelity approach to handle MO problems, bringing significant speed-ups on a wide range of multi-objective HPO problems; (3) we systematically apply MO in the context of fairness-aware ML, showing how it can be used to mitigate unfairness in domains related to financial lending and criminal justice.

2 Related Work

Multi-Objective HPO Early efforts on HPO focused on designing evolutionary optimization techniques for neural networks and SVMs [53, 38, 24, 18]. Evolutionary techniques have also been applied to multi-objective HPO problems [26]. To mitigate the large computational cost of evaluating a single hyperparameter configuration (e.g., training a DNN), sample-efficient Bayesian optimization (BO) techniques have been introduced [23, 46, 7].

Multi-objective Bayesian optimization (MBO) techniques can be categorized into three classes. First, *scalarization*-based MBO methods map the vector of all objectives to a scalar [31, 58, 37, 40, 19] and then use conventional single-objective (SO) BO techniques. While these methods are comparatively easy to implement and scale gracefully with the number of objectives, they do not utilize information about the overall geometry of the current Pareto front approximation. A second class of MBO techniques builds on a performance measure of Pareto front approximations, namely the *dominated hypervolume* [59]. Both the expected hypervolume improvement (EHI) [14] and probability hyper-improvement (PHI) [29] operate by extending their single-objective BO counterparts—expected improvement [36, 27] and probability of improvement [33], respectively. Other methods include step-wise uncertainty reduction (SUR) [43], smsEGO [45, 51] and expected maximin improvement (EMMI) [48]. Finally, *information-theoretic* MBO approaches aim to select points that reduce uncertainty about the location of the Pareto front. Methods in this class tend to be more sample efficient and scale better with the number of objectives. PAL [60] iteratively reduces the size of a discrete uncertainty set. PESMO [22] adapts the predictive entropy search (PES) [21] criterion. Pareto-frontier entropy search (PFES) [47] is suitable when dealing with decoupled objectives. MESMO [4] builds on the max-value entropy-search criterion [52] and enjoys an asymptotic regret bound. Building on MESMO, two very recent works have proposed MF-OSEMO [6] and iMOCA [5], two multi-fidelity based information-theoretic MBO techniques which internally use multi-fidelity Gaussian processes.

Our approach differs from all these prior methods as it is the first to combine scalarization techniques with the bandit-inspired Hyperband [35] algorithm. The candidate generation is based on random search and its cost is negligible. Unlike GP based techniques [6, 5], our approach allows for *efficient* evaluation of multiple candidates *in parallel* and can leverage modern computing infrastructure. It also employs *early-stopping*, saving valuable resources on unpromising configurations.

Algorithmic Fairness Algorithmic fairness techniques aim to train accurate models subject to fairness criteria. These methods can be divided into three families: (1) *post-processing* to modify a pre-trained model to increase the fairness of its outcomes [16, 20, 44]; (2) *in-processing* to enforce fairness constraints during training [1, 13, 54, 55]; and, (3) *pre-processing* to modify the data representation and then apply standard machine learning algorithms [11, 56]. Unlike our method, most of these techniques provide solutions for only a single fairness metric. For example, [16] is limited to demographic parity definition of fairness. In contrast to in-processing methods, which are often dependent on the model class (e.g., [54, 57, 13]) and hence have limited extensibility, our proposal treats the model as a blackbox, and can be extended to arbitrary model classes, as it operates *only* on the hyperparameters.

The method most similar to ours is that of [42] where the authors use a standard constrained BO approach (CBO) to find hyperparameters that maximize accuracy subject to fairness constraints. Unlike our method, CBO requires knowing *a propri* the highest level of accepted unfairness; the constrained approach of CBO is not aimed at finding the whole Pareto front, which enables the user to make the trade-off *a posteriori*.

3 Formal Problem Setup

Multi-Objective Optimization Let $f : \mathcal{X} \to \mathbb{R}^n$ be a function over domain $\mathcal{X}$ that we aim to minimize. Given two points $\boldsymbol{x}_1, \boldsymbol{x}_2 \in \mathcal{X}$, we will write $\boldsymbol{x}_1 \succeq \boldsymbol{x}_2$ if $\boldsymbol{x}_2$ is *weakly-dominated* by $\boldsymbol{x}_1$, that is, iff $f(\boldsymbol{x}_1)_i \leq f(\boldsymbol{x}_2)_i, \forall i \in [n]$. We write $\boldsymbol{x}_1 \succ \boldsymbol{x}_2$ if $\boldsymbol{x}_2$ is *dominated* by $\boldsymbol{x}_1$, that is, iff $\boldsymbol{x}_1 \succeq \boldsymbol{x}_2$ and $\exists i \in [n]$ s.t. $f(\boldsymbol{x}_1)_i < f(\boldsymbol{x}_2)_i$. The Pareto front of f is defined by $\mathcal{P}_f = \{\boldsymbol{x} \in \mathcal{X} | \nexists \boldsymbol{x}' \in \mathcal{X} : \boldsymbol{x}' \succ \boldsymbol{x}\}$, that is, the set of all non-dominated points. As $\mathcal{P}_f$ is often an infinite object, MO algorithms aim to find an *approximation set* $A \subset \mathcal{X}$ of non-dominated objective vectors. A popular measure of approximation quality is the *dominated hyper-volume* [59]. Given an approximation set A and a *reference point* $\boldsymbol{r}$ the hyper-volume indicator $\mathcal{H}$ is given by:

$$\mathcal{H}(A) = \text{Vol}\left(\{\boldsymbol{x} \in \mathbb{R}^n | \exists \boldsymbol{z} \in A : \boldsymbol{z} \succeq \boldsymbol{x} \wedge \boldsymbol{x} \succeq \boldsymbol{r}\}\right).$$

Hyper-volume related quantities are usually computed by partitioning the space into hyper-cubes which are then summed. This operation scales exponentially with the number objective functions and can cause a bottleneck for related MO approaches.

Fairness Definitions A single, universal definition of fairness is intrinsically difficult to find as what is an appropriate definition varies across applications and use cases [17]. Moreover, many definitions of fairness might quantitatively conflict with each other where a solution perfectly satisfying all the definitions is not better than a random or majority class assignment [50, 30]. However, obtaining solutions that satisfy various definitions to some (albeit) imperfect extent and expose empirical trade-offs between various definitions can still be important from societal perspectives. To this end, our method is flexible in that it can seamlessly incorporate multiple definitions either independently or simultaneously.

We consider the following standard framework: Y is the true label (binary), S is the protected (or sensitive) attribute (binary), and $\hat{Y}$ is the predicted label. Then, we can introduce the following commonly used definitions for fairness:

Equal Opportunity (EO) requires equal True Positive Rates (TPR) across subgroups: $P(\hat{Y} = 1 | Y = 1, S = 0) = P(\hat{Y} = 1 | Y = 1, S = 1)$;

Equalized Odds (EOdd) requires equality of False Positive Rates (FPR) in addition to EO;

Statistical Parity (SP) requires positive predictions to be unaffected by the value of the protected attribute, regardless of the actual true label: $P(\hat{Y} = 1 | S = 0) = P(\hat{Y} = 1 | S = 1)$.

We use the violation of the fairness constraint as a measure of unfairness. Following [13], we consider the family of ϵ-fair models: $\hat{Y}$ is ϵ-fair if it violates the fairness definition by at most $\epsilon \geq 0$. In the case of EO, a model $\hat{Y}$ is ϵ-fair if the *difference in equal opportunity* (DEO) is at most ϵ:

$$|P(\hat{Y} = 1 | Y = 1, S = 0) - P(\hat{Y} = 1 | Y = 1, S = 1)| \leq \epsilon. \tag{1}$$

In the case of EOdd, we can consider two types of fairness constraints: the first one is equivalent to DEO, and the second one, denoted by DFP, is the difference in FPR. In the case of SP, we use the difference in statistical parity (DSP) to measure unfairness, which is defined as follows:

$$|P(\hat{Y} = 1 | S = 0) - P(\hat{Y} = 1 | S = 1)| \leq \epsilon. \tag{2}$$

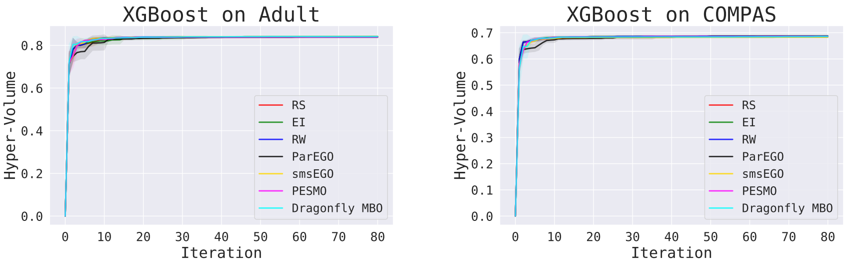

Figure 1: Dominated hyper-volume for XGBoost classifiers under error and DSP objectives on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. Among the different approaches, the quality of the generated approximations is very similar, with RS and RW being surprisingly competitive.

4 Preliminary Investigation

As an initial analysis, we evaluated the performance of five state-of-the-art MBO methods: ParEGO [31], smsEGO [45], EHI [14] PESMO [22] and a scalarization based method by Paria et al., [40] on four of the artificial benchmark functions used in [31] as well as on two fairness-related datasets. The motivation for this comparison is two-fold: (i) upon thorough review of the related literature, we found that there is a lack of comparisons against simple baselines, and (ii) only little attention has been put on evaluating MBO methods on prediction tasks commonly encountered in ML. The only ML-related benchmark in the MBO literature we are aware of and considered in [22] focuses on designing fast and accurate neural networks for MNIST [34]. For our investigation we used MBO implementations from the Spearmint [46] and Dragonfly [28] library.

We used three simple baselines: Random Search (RS), the SO criterion EI [36] which only optimizes for the first objective of the artificial functions and for classification error (1 - accuracy) in the ML tasks, and a simplified version of the ParEGO algorithm, which we call *Random Weights* (RW) and was not considered before. At every iteration t, RW reduces the MO to an SO problem in three steps: (i) A vector $\boldsymbol{w}_t$ is sampled uniformly from the unit simplex $\Delta_n = \{\boldsymbol{x} \in \mathbb{R}^n_{\geq 0} \mid \sum_{i=1}^n x = 1\}$. (ii) A set of scalar proxy objective values $\{s_1, \ldots, s_t\}$ is computed by taking the inner product between $\boldsymbol{w}_t$ and the previous objective vectors—i.e. $s_i = \langle \boldsymbol{w}_t, f(\boldsymbol{x}_i) \rangle, \forall i \in \{1, \ldots, t\}$. (iii) A surrogate model based on the scalar values is fitted and the standard EI criterion is applied to determine $\boldsymbol{x}_{t+1}$.

Appendix A, Figure 3 shows the average dominated hyper-volume over 100 iterations and 5 seeds on the four artificial functions. The advanced methods have an advantage over the simpler ones in most cases, although RS and RW are competitive on the artificial functions KNO1 and VLMOP2. Moreover, there is a clear difference in the per-iteration time over these competing approaches: RW are one order of magnitude faster than PESMO, and two orders of magnitude faster than smsEGO and EHI (see Appendix A, Table 2 for further information). Figure 1 illustrates the dominated hyper-volume over 80 iterations and 5 seeds for XGBoost on the Adult and COMPAS dataset (described in Section 6), with the objectives being the error and DSP and reference point for the hyper-volume computation being (error=1, DSP=1). While the per-iteration times of the different approaches still largely differ, the quality of the generated candidates is very similar, with RS and RW being surprisingly competitive.

5 Hyperband with Random Scalarizations

The experiments in Section 4 revealed the competitiveness of RS. Additionally, RS requires minimal computational overhead and is, unlike GP-based sequential techniques, easy to parallelize. Motivated by these attractive properties of RS, we explore its extension to the multi-fidelity setting by building upon the Hyperband algorithm [35]. Given a computational budget, Hyperband starts by providing a small initial resource allocation r_0 to each randomly sampled model configuration. If a configuration does not seem promising after its allocation is exhausted, Hyperband uses an early-stopping rule and reallocates additional larger resource allocations to a subset of most promising candidates. This process is repeated until the budget is exhausted. Unlike RS and standard GP based methods, Hyperband does not evaluate all candidate on their full budget and is able to allocate resources

Algorithm 1: Hyperband with Random Scalarizations

```
input          : $V$, $k$, $R$, $\eta$ (default $\eta = 3$)
initialization : $s_{max} = \lfloor \log_\eta(R) \rfloor$, $B = (s_{max} + 1)R$
for $s \in \{s_{max}, s_{max} - 1, \dots, 0\}$ do
    $n = \lceil \frac{B}{R} \frac{\eta^s}{(s+1)} \rceil, \quad r = R\eta^{-s}$
    $T = \{(\boldsymbol{x}_i, W_i = \{\boldsymbol{w}_{ij}\}_{j=1}^{k})\}_{i=1}^{n}$ where $\boldsymbol{x}_i \in \mathcal{X}$, $\boldsymbol{w}_{ij} \in \Delta_n$ are sampled uniformly
    for $i \in \{0, \dots, s\}$ do
        $n_i = \lfloor n\eta^{-i} \rfloor, \quad r_i = r\eta^i$
        $L = \{e_V(\boldsymbol{x}, W, r_i) \mid (\boldsymbol{x}, W) \in T\}$
        $T = \text{top}_m(T, L, \lfloor n_i/\eta \rfloor)$
return Pareto front approximation formed by evaluated configurations.
```

more efficiently. Hyperband extends the earlier Successive Halving algorithm [25] by introducing an additional parameter η which is used to trade of the number candidates with the per candidate budget.

For its early-stopping mechanism, Hyperband usually relies on the validation error of each configuration after partial training. This performance indicator allows us to select a subset of promising candidates, which then receives an increased resource allocation r_t. To adapt Hyperband to the MO domain, one needs to find an alternative performance measure for this ranking. The simplest way to approach this problem, which is investigated in this paper, is to employ scalarization techniques. For this each individual configuration $\boldsymbol{x} \in \mathcal{X}$ is equipped with a distinct set of k vectors $W = \{\boldsymbol{w}_1, \dots, \boldsymbol{w}_k\}$ sampled uniformly from Δ_n. With this the performance of configuration $\boldsymbol{x}$ after partial training is measured by

$$e_V(\boldsymbol{x}, W, r_t) = \min_{\boldsymbol{w} \in W} V(f(\boldsymbol{x}, r_t), \boldsymbol{w})$$

where $V : \mathbf{R}^n \times \mathbf{R}^n \to \mathbf{R}$ is a scalarization function of choice. An overview of our methods is given by Algorithm 1. As concrete choices of V we experimentally evaluate the scalarization functions employed by ParEGO and the simpler RW method.

6 Experiments

We evaluate our approach on two widely-used fairness-related binary classification datasets:

- Adult [32]: based on census data, contains binary gender as sensitive attribute with the class label being whether or not the income exceeds 50K USD;
- COMPAS [2]: data of criminal defendants, contains a binary sensitive attribute categorizing individuals into "white" and "other", with the target being the 2-year recidivism.

We perform a 70%/30% random split to form training and a validation sets. We search for efficient hyperparameters for gradient boosted tree ensemble (XGBoost [12]) and Multi-layer Perceptron (Sklearn MLP [41]) classifiers with respective 7- and 10-dimensional search spaces. The hyperparameter spaces are provided in Tables 3 and 4 in Appendix B. The maximum resource R per configuration is 200 epochs for MLPs and 256 boosting rounds for XGBoost. For our Hyperband-based method we choose $k = 100$ and use ParEGO and RW scalarization schemes. We perform comparisons with state-of-the-art MBO methods introduced in Section 4. The dominated hyper-volume is computed with respect to a worst case reference point (Error=1, DSP=1, DEO=1, DFP=1). For each dataset/model/objective-combination we perform 5 runs with different seeds and a reference Pareto front approximation $\mathcal{A}$ is formed by accumulating the evaluations from all runs. All experiments were performed using AWS m5.xlarge instances.

Comparison with existing MBO techniques Here we investigate the effectiveness of the proposed method to compute a good approximation set for the Pareto front of XGBoost and MLP models on the two datasets. We start with the two-objective scenario optimizing for error and DSP. The left side of Figure 2 visualizes the difference in average dominated hyper-volume of the MLP classifiers over wall-clock time w.r.t. reference approximation set $\mathcal{A}$. For both scalarization schemes our

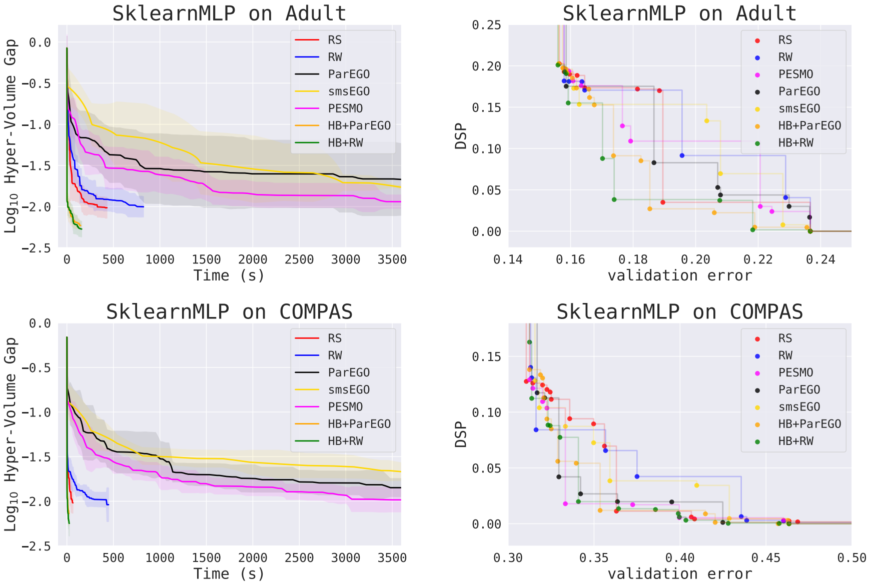

Figure 2: [Left] Dominated hyper-volume of the Pareto front approximations of MLP classifiers over time under error and DSP objective on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. [Right] Corresponding Pareto front approximations. For both scalarization schemes, our method obtains Pareto front approximations with larger hyper-volume in a significantly shorter wall-clock time.

method yields Pareto front approximations which (i) dominate a larger hyper-volume as visualized by the right side of Figure 2, (ii) are denser, hence allowing for a more granular trade-off during the model selection, and (iii) are obtained in a significantly shorter wall-clock time, as shown on the left hand side of Figure 2. The experiments with XGBoost models confirmed these observation (see Appendix B, Figure 5 for details). Experimental results for the 3- (Error, DSP, DEO) and 4- (Error, DSP, DEO, DFP) objective settings are shown in Figure 6 and 7 in Appendix B. Again, we observe that the Hyperband based method recovers Pareto front approximations covering a larger volume in significantly shorter wall-clock time. We ran into numerical difficulties when applying Spearmint's PESMO implementation to our 3- and 4-objective problems which prevented us from including it in the comparison.

Comparison with algorithmic fairness techniques We also perform comparisons with classic algorithmic fairness techniques which often require selecting a fixed definition of fairness and an *a priori* acceptance threshold. Following [42], we fix the DSP threshold at ≤ 0.1. Table 1 shows the most accurate fair model with DSP≤ 0.1 found by each of the baselines and our method.

The strongest of the baselines, FERM, produces a more accurate model on the COMPAS dataset, but is slightly less accurate compared to the XGB model identified by our method on Adult dataset. We note that all model-specific techniques tend to find solutions that are more fair than the required constraint. Our proposal is also the best model-agnostic method, outperforming both SMOTE and FERM preprocessing, and being comparable to CBO (both using MLP and XGB as base models). This shows that we can remove bias with a smaller impact on accuracy even without using specific-fairness constraints. We also highlight the flexibility of our model w.r.t. fairness threshold: if we decide to change the unfairness threshold to another value (e.g. DSP ≤ 0.05 from 0.1), our method has the benefit of being a multi-objective optimization algorithm and does not need any re-training, as it already found a set of models covering the whole Pareto front. Finally, it is important to note that, as our method only acts on the hyperparameters, it can be used on top of model-specific techniques, which come with their own hyperparameters. This hybrid strategy can help boost the performance of these schemes (as opposed to blindly tuning the hyperparameters).

Table 1: Validation error of the best fair models for model-specific (first three rows) and model-agnostic fairness methods. We use the fairness constraint, DSP ≤ 0.1.

Method	Adult	COMPAS
FERM	0.164 ± 0.010	0.285 ± 0.009
Zafar	0.187 ± 0.001	0.411 ± 0.063
Adversarial	0.237 ± 0.001	0.327 ± 0.002
FERM pre-processed	0.228 ± 0.013	0.343 ± 0.002
SMOTE	0.178 ± 0.005	0.321 ± 0.002
CBO MLP	0.167 ± 0.017	0.316 ± 0.004
CBO XGB	0.160 ± 0.003	0.313 ± 0.002
HB+RW MLP (ours)	0.168 ± 0.002	0.324 ± 0.003
HB+RW XGB (ours)	0.159 ± 0.001	0.310 ± 0.001

7 Conclusion and Future Work

We proposed a novel multi-fidelity multi-objective HPO method based on Hyperband that is computationally efficient and is easily parallelizable. Its use of scalarization techniques makes it amenable to a large number of objectives. Our experimental results show that our method is an order of magnitude faster for MO HPO problems compared to existing MBO techniques. It also returns denser Pareto front approximations allowing practitioners a more granular trade-off between the objectives. We compared our blackbox approach to specialized fairness techniques on two fairness related datasets showing competitive performance. Our method is applicable to other MO problems as well.

In future work we want to explore more specialized scalarization techniques as well as other ways to identify promising subsets of hyperparameters which can be used for efficient resource allocation. We also would like to compare the performance of our method to the ones proposed by Belakaria et al. [4, 6, 5], for which as for now there is no source code available. An implementation of our method is currently under review to be included in the open source project AutoGluon [15].

References

[1] A. Agarwal, A. Beygelzimer, M. Dudik, J. Langford, and H. Wallach. A reductions approach to fair classification. *ICML*, 2018.

[2] J. Angwin, J. Larson, S. Mattu, and L. Kirchner. Machine bias. *ProPublica*, 2016.

[3] S. Barocas, M. Hardt, and A. Narayanan. Fairness and machine learning. *URL: www.fairmlbook.org*, 2018.

[4] Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Max-value entropy search for multi-objective bayesian optimization. In *Advances in Neural Information Processing Systems*, pages 7823–7833, 2019.

[5] Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Information-theoretic multi-objective bayesian optimization with continuous approximations. *arXiv preprint arXiv:2009.05700*, 2020.

[6] Syrine Belakaria, Aryan Deshwal, and Janardhan Rao Doppa. Multi-fidelity multi-objective bayesian optimization: An output space entropy search approach. In *AAAI*, pages 10035–10043, 2020.

[7] James S Bergstra, Rémi Bardenet, Yoshua Bengio, and Balázs Kégl. Algorithms for hyper-parameter optimization. In *Advances in neural information processing systems*, pages 2546–2554, 2011.

[8] T. Bolukbasi, K. Chang, J. Y. Zou, V. Saligrama, and A. T. Kalai. Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *NeurIPS*, 2016.

[9] J. Buolamwini and T. Gebru. Gender shades: Intersectional accuracy disparities in commercial gender classification. *FAT**, 2018.

[10] A. Caliskan, J. J. Bryson, and A. Narayanan. Semantics derived automatically from language corpora contain human-like biases. *Science*, 2017.

[11] F. Calmon, D. Wei, B. Vinzamuri, K. N. Ramamurthy, and K. R. Varshney. Optimized pre-processing for discrimination prevention. *NeurIPS*, 2017.

[12] Tianqi Chen and Carlos Guestrin. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794, 2016.

[13] M. Donini, L. Oneto, S. Ben-David, J. S. Shawe-Taylor, and M. Pontil. Empirical risk minimization under fairness constraints. *NeurIPS*, 2018.

[14] Michael TM Emmerich, André H Deutz, and Jan Willem Klinkenberg. Hypervolume-based expected improvement: Monotonicity properties and exact computation. In *2011 IEEE Congress of Evolutionary Computation (CEC)*, pages 2147–2154. IEEE, 2011.

[15] Nick Erickson, Jonas Mueller, Alexander Shirkov, Hang Zhang, Pedro Larroy, Mu Li, and Alexander Smola. Autogluon-tabular: Robust and accurate automl for structured data. *arXiv preprint arXiv:2003.06505*, 2020.

[16] M. Feldman, S. A. Friedler, J. Moeller, C. Scheidegger, and S. Venkatasubramanian. Certifying and removing disparate impact. *ACM SIGKDD*, 2015.

[17] S. A. Friedler, C. Scheidegger, and S. Venkatasubramanian. On the (im)possibility of fairness. *arXiv preprint arXiv:1609.07236*, 2016.

[18] Frauke Friedrichs and Christian Igel. Evolutionary tuning of multiple svm parameters. *Neurocomputing*, 64:107–117, 2005.

[19] Daniel Golovin et al. Random hypervolume scalarizations for provable multi-objective black box optimization. *arXiv preprint arXiv:2006.04655*, 2020.

[20] M. Hardt, E. Price, and N. Srebro. Equality of opportunity in supervised learning. *NeurIPS*, 2016.

[21] José Miguel Henrández-Lobato, Matthew W. Hoffman, and Zoubin Ghahramani. Predictive entropy search for efficient global optimization of black-box functions. In *Proceedings of the 27th International Conference on Neural Information Processing Systems - Volume 1*, NIPS'14, page 918–926, Cambridge, MA, USA, 2014. MIT Press.

[22] Daniel Hernández-Lobato, Jose Hernandez-Lobato, Amar Shah, and Ryan Adams. Predictive entropy search for multi-objective bayesian optimization. In *International Conference on Machine Learning*, pages 1492–1501, 2016.

[23] Frank Hutter, Holger H Hoos, and Kevin Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *International conference on learning and intelligent optimization*, pages 507–523. Springer, 2011.

[24] Christian Igel, Stefan Wiegand, and Frauke Friedrichs. Evolutionary optimization of neural systems: The use of strategy adaptation. In *Trends and Applications in Constructive Approximation*, pages 103–123. Springer, 2005.

[25] Kevin Jamieson and Ameet Talwalkar. Non-stochastic best arm identification and hyperparameter optimization. In *Artificial Intelligence and Statistics*, pages 240–248, 2016.

[26] Yaochu Jin. *Multi-objective Machine Learning*, volume 16. Springer Science & Business Media, 2006.

[27] Donald R Jones, Matthias Schonlau, and William J Welch. Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, 13(4):455–492, 1998.

[28] Kirthevasan Kandasamy, Karun Raju Vysyaraju, Willie Neiswanger, Biswajit Paria, Christopher R Collins, Jeff Schneider, Barnabas Poczos, and Eric P Xing. Tuning hyperparameters without grad students: Scalable and robust bayesian optimisation with dragonfly. *Journal of Machine Learning Research*, 21(81):1–27, 2020.

[29] Andy J Keane. Statistical improvement criteria for use in multiobjective design optimization. *AIAA journal*, 44(4):879–891, 2006.

[30] J. Kleinberg. Inherent trade-offs in algorithmic fairness. *SIGMETRICS*, 2018.

[31] Joshua Knowles. Parego: a hybrid algorithm with on-line landscape approximation for expensive multiobjective optimization problems. *IEEE Transactions on Evolutionary Computation*, 10(1):50–66, 2006.

[32] Ron Kohavi. Scaling up the accuracy of naive-bayes classifiers: A decision-tree hybrid. In *Kdd*, volume 96, pages 202–207, 1996.

[33] H. J. Kushner. A New Method of Locating the Maximum Point of an Arbitrary Multipeak Curve in the Presence of Noise. *Journal of Basic Engineering*, 86(1):97–106, 03 1964.

[34] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[35] L. Li, K. Jamieson, G. DeSalvo, Afshin Rostamizadeh, and Ameet Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *J. Mach. Learn. Res.*, 18:185:1–185:52, 2017.

[36] Jonas Mockus, Vytautas Tiesis, and Antanas Zilinskas. The application of bayesian methods for seeking the extremum. *Towards global optimization*, 2(117-129):2, 1978.

[37] Hirotaka Nakayama, Yeboon Yun, and Min Yoon. *Sequential approximate multiobjective optimization using computational intelligence*. Springer Science & Business Media, 2009.

[38] Stefano Nolfi and Domenico Parisi. Evolution of artificial neural networks. In *In Handbook of brain theory and neural networks*, pages 418–421. MIT Press, 2002.

[39] Tatsuya Okabe, Yaochu Jin, Markus Olhofer, and Bernhard Sendhoff. On test functions for evolutionary multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 792–802. Springer, 2004.

[40] Biswajit Paria, Kirthevasan Kandasamy, and Barnabás Póczos. A flexible framework for multi-objective bayesian optimization using random scalarizations. In *UAI*, 2019.

[41] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[42] Valerio Perrone, Michele Donini, Krishnaram Kenthapadi, and Cédric Archambeau. Fair bayesian optimization. *AutoML Workshop - ICML*, 2020.

[43] Victor Picheny. Multiobjective optimization using gaussian process emulators via stepwise uncertainty reduction. *Statistics and Computing*, 25(6):1265–1280, 2015.

[44] G. Pleiss, M. Raghavan, F. Wu, J. Kleinberg, and K. Q. Weinberger. On fairness and calibration. *NeurIPS*, 2017.

[45] Wolfgang Ponweiser, Tobias Wagner, Dirk Biermann, and Markus Vincze. Multiobjective optimization on a limited budget of evaluations using model-assisted $\mathcal{S}$-metric selection. In *International Conference on Parallel Problem Solving from Nature*, pages 784–794. Springer, 2008.

[46] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

[47] Shinya Suzuki, Shion Takeno, Tomoyuki Tamura, Kazuki Shitara, and Masayuki Karasuyama. Multi-objective bayesian optimization using pareto-frontier entropy. *arXiv preprint arXiv:1906.00127*, 2019.

[48] Joshua D Svenson and Thomas J Santner. Multiobjective optimization of expensive black-box functions via expected maximin improvement. *The Ohio State University, Columbus, Ohio*, 32, 2010.

[49] David A Van Veldhuizen and Gary B Lamont. Multiobjective evolutionary algorithm test suites. In *Proceedings of the 1999 ACM symposium on Applied computing*, pages 351–357, 1999.

[50] S. Verma and J. Rubin. Fairness definitions explained. *FairWare*, 2018.

[51] Tobias Wagner, Michael Emmerich, André Deutz, and Wolfgang Ponweiser. On expected-improvement criteria for model-based multi-objective optimization. In *International Conference on Parallel Problem Solving from Nature*, pages 718–727. Springer, 2010.

[52] Zi Wang and Stefanie Jegelka. Max-value entropy search for efficient bayesian optimization. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 3627–3635. JMLR.org, 2017.

[53] Xin Yao. Evolving artificial neural networks. *Proceedings of the IEEE*, 87(9):1423–1447, 1999.

[54] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness constraints: Mechanisms for fair classification. *AISTATS*, 2017.

[55] M. B. Zafar, I. Valera, M. Gomez-Rodriguez, and K. P. Gummadi. Fairness constraints: A flexible approach for fair classification. *JMLR*, 2019.

[56] R. Zemel, Y. Wu, K. Swersky, T. Pitassi, and C. Dwork. Learning fair representations. *ICML*, 2013.

[57] B. H. Zhang, B. Lemoine, and M. Mitchell. Mitigating unwanted biases with adversarial learning. *AIES*, 2018.

[58] Qingfu Zhang, Wudong Liu, Edward Tsang, and Botond Virginas. Expensive multiobjective optimization by moea/d with gaussian process model. *IEEE Transactions on Evolutionary Computation*, 14(3):456–474, 2009.

[59] Eckart Zitzler and Lothar Thiele. Multiobjective optimization using evolutionary algorithms—a comparative case study. In *International conference on parallel problem solving from nature*, pages 292–301. Springer, 1998.

[60] Marcela Zuluaga, Guillaume Sergent, Andreas Krause, and Markus Püschel. Active learning for multi-objective optimization. In *International Conference on Machine Learning*, pages 462–470, 2013.

A Benchmark with Artificial Functions

We evaluate existing MBO techniques on some popular artificial functions which were also used in the original ParEGO paper by Knowles [31]. KNO1 [39], OKA1 [39] and VLMOP2 [49] are popular benchmark functions with two input and two output dimensions. VLMOP3 [49] is a function with two input and three output dimensions. For each function we let each algorithm iteratively evaluate 100 points and compute the dominated hyper-volume of the Pareto front at each step with respect to a reference point which is determined as described in [31].

Figure 3 visualizes the average dominated hyper-volume and variance for 5 random seeds. There are rather clear differences in performance between the individual methods. The more advanced MBO algorithms seem to outperform the simpler methods although RS and RW are competitive on the artificial functions KNO1 and VLMOP2. The average per iteration time between the individual methods varies largely and is provided in Table 2. EHI scales very poorly with the number of objectives and we aborted the runs on VLMOP3 after 24h. We note that the two-dimensional input of these artificial functions has a much smaller dimension compared to the 7- and 10-dimensional hyperparameter spaces used in our HPO experiments.

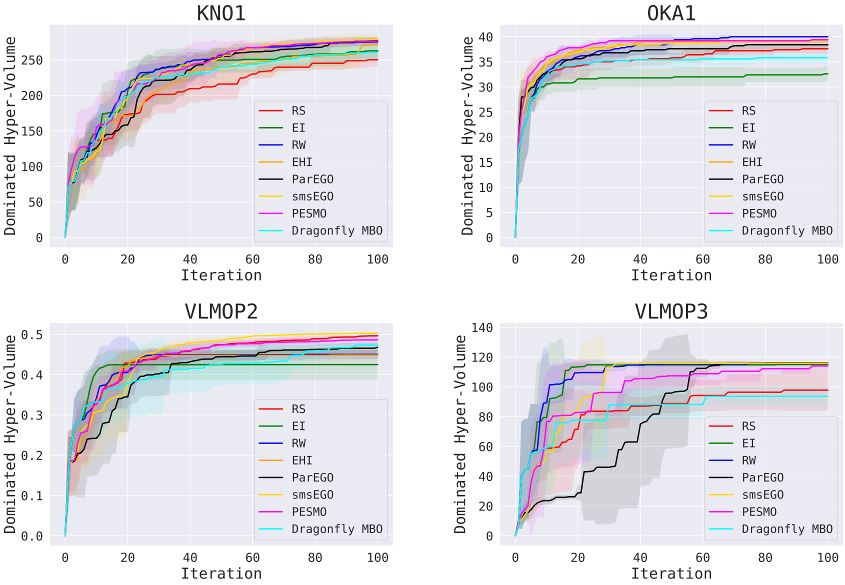

Figure 3: Dominated hyper-volume values for various multi-objective Bayesian optimization algorithms achieved on common artificial functions over 100 iterations. The average and standard deviation for 5 random seeds is shown. The more advanced methods have an advantage over the simpler ones. RS and RW are surprisingly competitive on KNO1 and VLMOP2.

Table 2: Average per iteration time over 100 steps in seconds. The time is solely dominated by the time it takes to determine the next evaluation candidate. The per iteration time varies largely between the individual methods. The EHI runs for 3-objective problem VLMOP3 were aborted after 24h.

Function	RS	EI	RW	ParEGO	PESMO	smsEGO	EHI
KNO1	0.01	2.50	2.86	18.52	31.46	276.36	327.79
OKA1	0.01	2.78	2.74	19.37	39.50	278.02	372.88
VLMOP2	0.01	1.97	2.48	23.17	42.94	305.19	802.11
VLMOP3	0.01	2.35	2.14	25.02	60.40	414.38	-

B Benchmark with FairML Tasks

Adding to Section 6, we provide more details about the experimental setup and visualize additional results. A detailed overview of the hyperparameter spaces used for the MLP and XGBoost classifiers is given by Table 3 and 4 respectively. Results for an experiment comparing various MBO methods on two fairness related datasets are visualized in Figure 4. Figure 5 visualizes dominated hyper-volume over time for XGB classifiers which are optimized for error and DSP. Figure 6 and 7 visualize experimental results for a 3 objective setting with error, DSP and DEO objective and a 4 objective setting with error, DSP, DEO and DFP objective respectively.

Table 3: Sklearn MLP search space

Parameter	Type	Domain	Scaling
n_layers	integer	$\{1, 2, 3, 4\}$	linear
layer_1	integer	$\{2, \ldots, 32\}$	logarithmic
layer_2	integer	$\{2, \ldots, 32\}$	logarithmic
layer_3	integer	$\{2, \ldots, 32\}$	logarithmic
layer_4	integer	$\{2, \ldots, 32\}$	logarithmic
alpha	real	$[10^{-6}, \ldots, 10^{-1}]$	logarithmic
learning_rate_init	real	$[10^{-6}, \ldots, 10^{-2}]$	logarithmic
beta_1	real	$[0.001, 0.99]$	logarithmic
beta_2	real	$[0.001, 0.99]$	logarithmic
tol	real	$[10^{-5}, 10^{-2}]$	logarithmic

Table 4: XGBoost search space

Parameter	Type	Domain	Scaling
n_estimators	integer	$\{1, 2, ..., 256\}$	logarithmic
learning_rate	real	$[0.01, 1.0]$	logarithmic
gamma	real	$[0.0, 0.1]$	linear
reg_alpha	real	$[10^{-3}, 10^{3}]$	logarithmic
reg_lambda	real	$[10^{-3}, 10^{3}]$	logarithmic
subsample	real	$[0.01, 1.0]$	linear
max_depth	integer	$\{1, 2, \ldots, 16\}$	linear

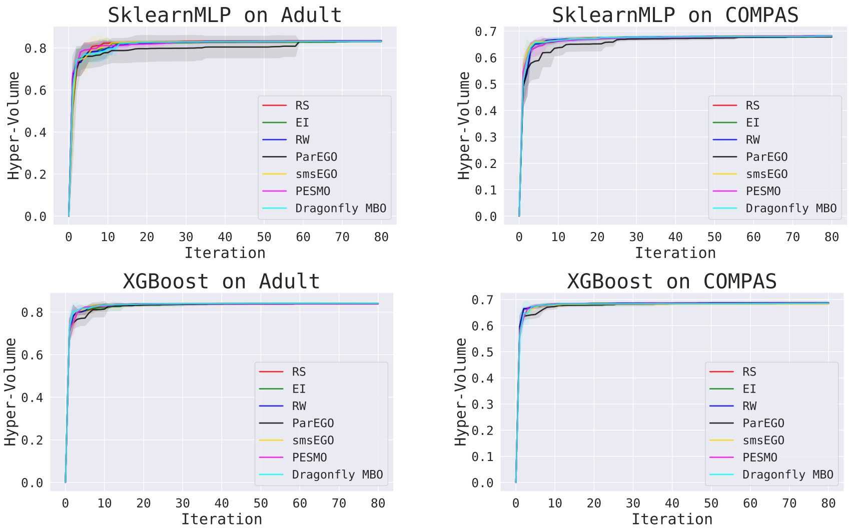

Figure 4: Dominated hyper-volume for MLP and XGBoost classifiers under error and DSP objectives on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. Among the different approaches, the quality of the generated approximations is very similar, with RS and RW being surprisingly competitive

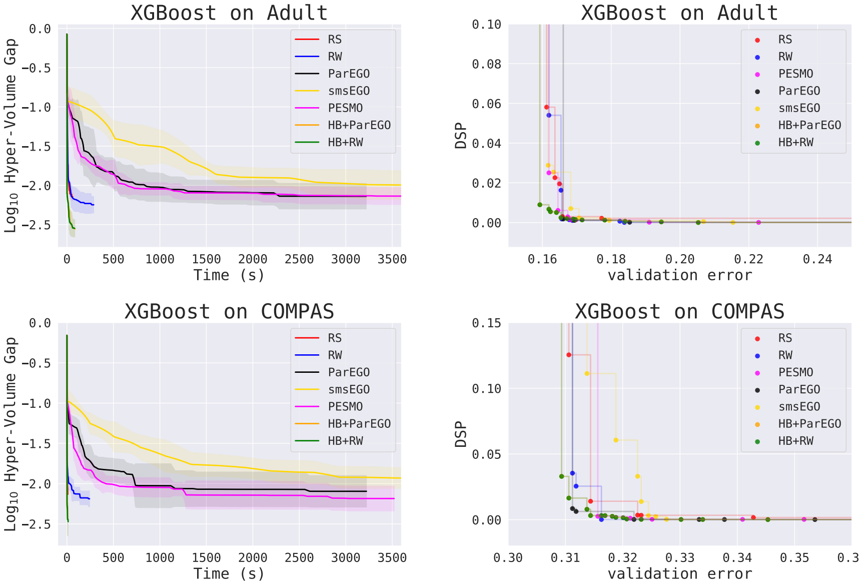

Figure 5: [Left] Dominated hyper-volume of the Pareto front approximations of XGBoost classifiers over time under error and DSP objective on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. [Right] Corresponding Pareto front approximations. For both scalarization schemes, our method obtains Pareto front approximations with larger hyper-volume in a significantly shorter wall-clock time.

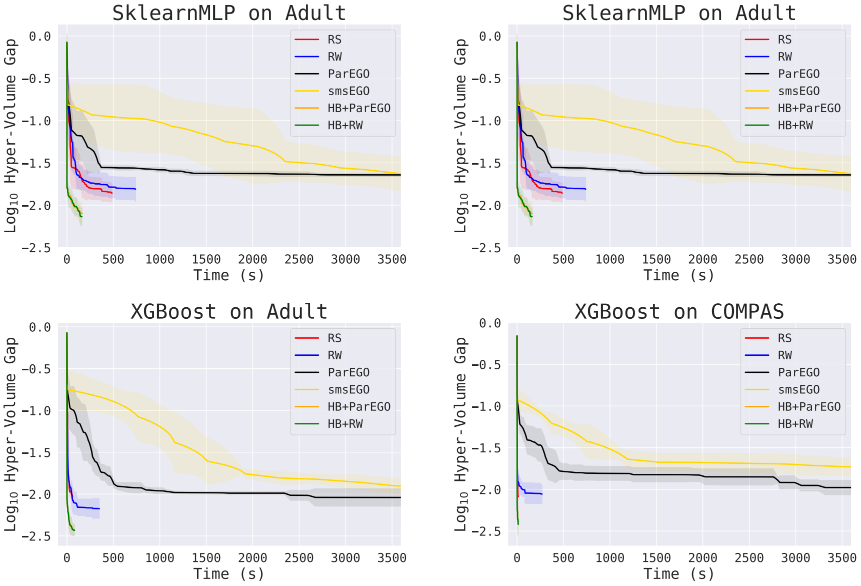

Figure 6: Dominated hyper-volume of the Pareto front approximations of MLP and XGBoost classifiers over time under error, DSP and DEO objective on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. For both scalarization schemes, our method obtains Pareto front approximations with larger hyper-volume in a significantly shorter wall-clock time. With increasing problem dimension smsEGO requires a larger computational budget to determine the next candidate configuration at each step.

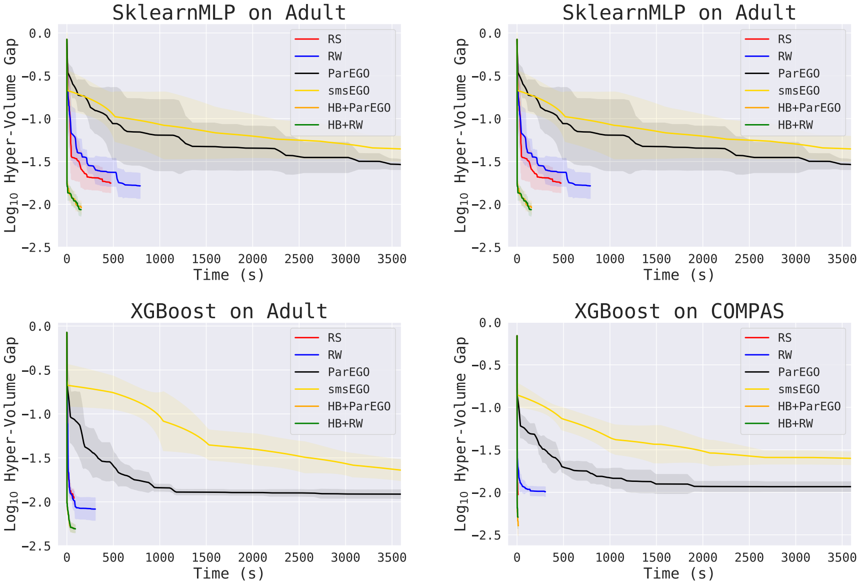

Figure 7: Dominated hyper-volume of the Pareto front approximations of MLP and XGBoost classifiers over time under error, DSP, DEO and DFP objective on Adult and COMPAS dataset. The average and standard deviation for 5 random seeds is shown. For both scalarization schemes, our method obtains Pareto front approximations with larger hyper-volume in a significantly shorter wall-clock time. With increasing problem dimension smsEGO requires a larger computational budget to determine the next candidate configuration at each step.