Hardware Aware Ensemble Selection for Balancing Predictive Accuracy and Cost

Maier, Jannis[1,2] **Möller, Felix**[1] **Purucker, Lennart**[3]

[1]Helmholtz-Zentrum Berlin [2]Humboldt University Berlin [3]University of Freiburg

Abstract   Automated Machine Learning (AutoML) significantly simplifies the deployment of machine learning models by automating tasks from data preprocessing to model selection to ensembling. AutoML systems for tabular data often employ *post hoc* ensembling, where multiple models are combined to improve predictive accuracy. This typically results in longer inference times, a major limitation in practical deployments. Addressing this, we introduce a hardware-aware ensemble selection approach that integrates inference time into *post hoc* ensembling. By leveraging an existing framework for ensemble selection with quality diversity optimization, our method evaluates ensemble candidates for their predictive accuracy and hardware efficiency. This dual focus allows for a balanced consideration of accuracy and operational efficiency. Thus, our approach enables practitioners to choose from a Pareto front of accurate and efficient ensembles. Our evaluation using 83 classification datasets shows that our approach sustains competitive accuracy and can significantly improve ensembles' operational efficiency. The results of this study provide a foundation for extending these principles to additional hardware constraints, setting the stage for the development of more resource-efficient AutoML systems.

1 Introduction

Automated Machine Learning (AutoML) aims to automate the entire machine learning pipeline—from data preprocessing and feature selection to model selection and hyperparameter tuning—thereby reducing the need for manual intervention and expertise. State-of-the-art AutoML systems often use *post hoc* ensembling to further enhance model predictive accuracy (Purucker et al., 2023; Purucker and Beel, 2023). *Post hoc* ensembling involves creating ensembles from models generated during the model selection phase. By combining the predictions of multiple models, post hoc ensembling leverages the strengths of individual models and mitigates their weaknesses, resulting in improved predictive accuracy and generalization ability. Here, greedy ensemble selection (GES) (Caruana et al., 2004) is used in prominent AutoML frameworks like Auto-Sklearn (Feurer et al., 2020) and AutoGluon (Erickson et al., 2020) to further improve predictive accuracy over the single best model (Feurer et al., 2020; Purucker et al., 2023; Purucker and Beel, 2023).

Yet, *post hoc* ensembling also further increases the predictive cost of AutoML systems w.r.t. disk footprint, memory requirement, and inference time. GES for Auto-Sklearn requires, on average, ~8.5 models for inference (Purucker et al., 2023), whereby each model has to be stored on disk for deployment, in memory for inference, and has its own inference overhead. The problem is that practitioners cannot trade off improved predictive accuracy and increased predictive cost with existing *post hoc* ensembling algorithms. Furthermore, a greedy search for improved predictive accuracy, like in GES, might not provide a good set of options to choose from.

We aim to enable practitioners to balance improved predictive accuracy and increased predictive cost in AutoML. Therefore, in this work, we study *hardware-aware* ensemble selection, an approach to obtain a Pareto front (defined in Appendix A) for predictive accuracy and cost in AutoML. We focus on ensemble selection because it natively produces smaller, less expensive ensembles (Purucker and Beel, 2023) and is most often used in state-of-the-art AutoML systems (Purucker et al., 2023).

Specifically, we amend existing *post hoc* ensemble selection algorithms with concepts from quality diversity optimization (Cully et al., 2015; Mouret and Clune, 2015; Chatzilygeroudis et al., 2021) to obtain a set of solutions with different predictive costs while only optimizing for a single objective, the predictive performance. In detail, we extend GES (Caruana et al., 2004), quality optimization ensemble selection (QO-ES), and quality diversity optimization ensemble selection (QDO-ES) (Purucker et al., 2023) to obtain Pareto fronts for predictive accuracy and cost. Furthermore, we propose two variants of QDO-ES to obtain better Pareto fronts for hardware-aware ensemble selection.

Our experiments with data from TabRepo (Salinas and Erickson, 2023) on 83 classification datasets (58 binary, 25 multi-class, see Table 1) and 1416 machine-learning models show that hardware-aware ensemble selection can effectively trade off predictive accuracy and cost. Moreover, we show that our proposed variants of QDO-ES produce statistically significantly better Pareto fronts than just extending *post hoc* ensemble selection algorithms. At the same time, our results reproduce the conclusions by Purucker et al. (2023) w.r.t. predictive accuracy for data from Auto-Sklearn with TabRepo's data from AutoGluon.

Contributions. In this work, we present (1) hardware-aware ensemble selection, a *post hoc* method for trading off predictive accuracy against predictive cost in AutoML, and (2) a strong hardware-aware ensemble selection algorithm that can serve as a baseline for future work while already being able to support practitioners' choices effectively.

2 Related Work

Integrating hardware constraints into AutoML and Neural Architecture Search (NAS) has gained significant attention over the last few years (Zhang et al., 2019; Benmeziane et al., 2021; Schneider et al., 2022; Sukthanker et al., 2024), reflecting the diverse demands of deployment environments.

Schneider et al. (2022) introduced the application of QDO to NAS, aimed at generating a diverse set of architectures, each optimized for specific hardware constraints. Unlike traditional multi-objective NAS that seeks to approximate a Pareto front, this method focuses on optimizing for predefined, niche-specific requirements. The study illustrates that QDO's targeted approach potentially offers improvements over traditional methods in terms of efficiency and solution quality for hardware-aware NAS.

In the domain of multi-objective ensemble selection, several studies have explored optimizing model robustness and diversity without directly addressing hardware constraints. Works by Cavalcanti et al. (2016); Li et al. (2012); Partalas et al. (2010); Martınez-Munoz and Suárez (2004); Partridge and Yates (1996) have contributed to the understanding of ensemble pruning and selection by focusing on performance and uncertainty measures, yet typically do not consider hardware efficiency as a critical objective. To the best of our knowledge, no research has been done on hardware awareness in ensemble selection. We extend this focus by integrating hardware constraints, thereby enhancing the scope of multi-objective optimization in ensemble methods for AutoML solutions.

Our approach to hardware-aware ensemble selection builds on applying QDO-ES as proposed by Purucker et al. (2023). QDO-ES uses principles from QDO and concepts from ensemble diversity to enhance the performance and robustness of ensemble selection. We extend this by including hardware-aware metrics, like inference time or ensemble size, when applying QDO to ensemble selection to create hardware-aware QDO-ES.

3 Method

In this study, we are comparing the performance of five ensembling methods and their ability to find an optimal Pareto front for predictive accuracy and inference time: (i) GES; (ii) QO-ES; (iii) QDO-ES; (iv) QDO-ES with ensemble size (Size-QDO-ES); and (v) QDO-ES with inference time (Infer-QDO-ES).

Greedy Ensemble Selection. GES was first introduced by Caruana et al. (2004). Starting from the single best model, GES iteratively adds models to the ensemble that lead to the largest improvement. Compared to the other methods, GES usually produces only one solution. Thus, we extend GES by recording the ensemble produced at every iteration. Then, we obtain a set of ensembles that we use to compute a Pareto front.

Quality (Diversity) Optimization Ensemble Selection. QO-ES and QDO-ES implemented by Purucker et al. (2023) maintain a population of ensembles, which both methods' evolution strategies iterate over to improve performance. QDO-ES extends QO-ES in that it makes sure the population includes diverse ensembles. In their application of QO-ES and QDO-ES, the algorithms only returned one solution. We extended both methods to extract the last population of ensembles of QO-ES and QDO-ES. Then, we use this population as the set of ensembles to compute the Pareto front. Apart from this, we left the remaining parameters of the evolutionary algorithms default, following the prior work. We also include the single best model in this set, i.e., an ensemble with only one member. This follows the set for GES, which starts with the single best model.

Hardware-Aware Quality Diversity Optimization Ensemble Selection. GES, QO-ES, and QDO-ES solely optimize for predictive accuracy. Thus, no aforementioned method is hardware-aware[1]. Therefore, we propose a variant of QDO-ES by adjusting its population management. Instead of creating a population that includes diverse ensembles, we guarantee that QDO-ES ensures that the population consists of ensembles with varying degrees of predictive cost.

To this end, we adjust the behavior space (Chatzilygeroudis et al., 2021) that QDO-ES operates on. The behavior space determines which individuals (i.e., ensembles) are kept in the population after each iteration. To guarantee that the behavior space becomes hardware-aware, we replace one of its two dimensions with a measure of predictive cost. In detail, we replace the config space similarity metrics, an ensemble diversity metric used by QDO-ES, with either inference time or ensemble size to obtain Infer-QDO-ES and Size-QDO-ES, respectively. Consequently, after each iteration, the population will contain ensembles with varying degrees of inference time or ensemble size. As for QO-ES and QDO-ES, we used the last population together with the single best model as the set of models for computing the Pareto front. While we could have used all ensembles from all iterations of the genetic algorithm, we want to provide a manageable set of solutions to a user.

We include Size-QDO-ES as an ablation for a different predictive cost measure. Ensemble size is generally a proxy for predictive cost and inference time but does not necessarily represent the true predictive cost. We follow the defaults of the implementation provided by Purucker et al. (2023) for population size and individual selection method.

4 Experiments

We use TabRepo (Salinas and Erickson, 2023) as a foundation for our experiments. TabRepo includes prediction probabilities on validation and test data for up to 1416 model configurations and 200 datasets, making it an invaluable tool for evaluating and simulating ensemble selection methods. We use its data *D244_F3_C1530_100*, which includes the results for 100 datasets with three-fold cross-validation and 1416 models. We omitted the 17 regression datasets from our testing because QDO-ES does not support regression, though it could be extended in the future. Each method was executed across 10 different seeds per fold to ensure robustness in our findings.

To evaluate Pareto fronts for predictive accuracy and cost for each method, we measure predictive accuracy using the ROC AUC on test data and predictive cost by inference time. Inference times are calculated based on the data in TabRepo, where inference times are recorded per dataset and model. We then define an ensemble's inference time as the sum of the inference times of all its models. To measure the quality of Parteo fronts, we employ *hypervolume* (defined in Appendix A).

---

[1]Ensemble selection by itself is not hardware-aware as sparse solutions can have an extreme predictive cost.

(a) Ranking per method (lower is better)

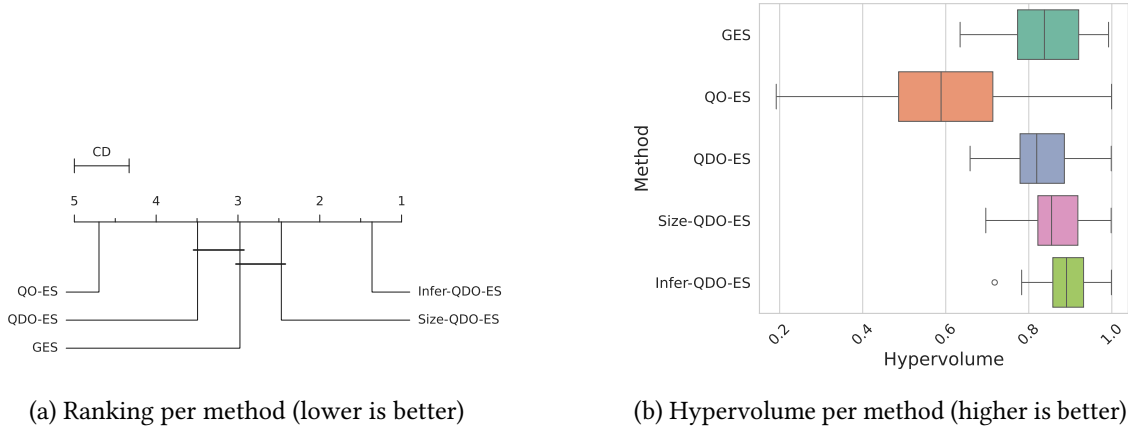(b) Hypervolume per method (higher is better)

Figure 1: Hypervolume plots: (a) a critical difference plot of hypervolume, which shows the average rank (top axis) and a horizontal black bar that connects all methods that are not statistically significantly different after a Friedman test with a Nemenyi posthoc test ($\alpha = 0.05$). (b) a boxplot showing the distribution of a method's obtained hypervolume across datasets.

A higher hypervolume indicates a method's superior capability in constructing an effective set of ensembles that balances predictive accuracy and cost. We calculate hypervolume using the *pygmo* Python package (Biscani and Izzo, 2020). Before computing the hypervolume, we min-max normalized each metric (ROC AUC and inference time) and ensured that lower values represent better values. To obtain an aggregate across folds and seeds, we first calculated hypervolumes for each fold and seed and then averaged them to provide a value for each method per dataset.

In addition to hypervolume, we evaluated each method's predictive accuracy based on the best-scoring ensemble, according to ROC AUC on validation data, in the Pareto front. Moreover, we examined the efficiency of the best-scoring ensembles by analyzing their inference times.

5 Results

Balancing Predictive Accuracy and Cost. To determine which method best balances predictive accuracy and cost, we examined the hypervolumes of the Pareto fronts produced by each method, as illustrated in Figure 1. Infer-QDO-ES demonstrated superior performance, outpacing other methods with a statistically significant margin. It excelled in identifying solutions that balance predictive performance and cost. GES, QDO-ES, and GES showed similar results regarding hypervolume, indicating a lower but comparable capability in optimizing the trade-off between efficiency and performance. QO-ES performed worst, likely attributed to the lack of diversity compared to QDO-ES. We present exemplary Pareto fronts for all methods in Appendix, Figure 4.

Predictive Accuracy after Balancing. Using Infer-QDO-ES and Size-QDO-ES does not detrimentally affect predictive performance when returning only the best ensemble of the Pareto front. These methods even surpassed the performance of QDO-ES, as shown in the Appendix, Figure 2. We additionally provided ROC AUC results for each method per dataset in the Appendix, Table 2. We observe that our methods, alongside GES, QDO-ES, and QO-ES, generally yield comparable, not significantly different high-quality results across the tested datasets. Moreover, including hardware-aware metrics in QDO-ES might even improve the robustness of the generated ensembles.

Our results, as shown in Figure 2, reproduce the conclusions by Purucker et al. (2023) that QO-ES and QDO-ES outrank GES w.r.t. ROC AUC on test data. This is particularly interesting since Purucker et al. (2023) used data from Auto-Sklearn while we used TabRepo's data from AutoGluon.

Predictive Cost after Balancing. When assessing the efficiency of the best ensembles, particularly in inference times, our methods outperformed those created by QDO-ES and QO-ES. This is depicted in the Appendix, Figure 3. However, they did not achieve the efficiency levels of GES, which typically produces a smaller final ensemble requiring fewer models during inference. Given the larger ensemble sizes typically generated by QO-ES and QDO-ES (Purucker et al., 2023), this outcome aligns with our expectations. Moreover, as seen in Figure 1, practitioners could also choose a comparable or better ensemble from the Pareto fronts when using Infer-QDO-ES or Size-QDO-ES.

Summary. Our results demonstrate that Size-QDO-ES and Infer-QDO-ES enhance the efficiency of AutoML solutions and maintain competitive accuracy, effectively managing the trade-offs between accuracy and operational speed. Our methods' ability to generate superior Pareto fronts underscores their effectiveness and provides greater flexibility for practitioners, allowing for choosing models to meet specific hardware or business constraints.

6 Conclusion

Our study demonstrated the viability of hardware-aware post hoc ensemble selection in AutoML, presenting a significant advancement in ensemble selection methodologies that integrate hardware efficiency considerations. By incorporating a hardware constraint like inference time into the ensemble selection process, we have shown that it is possible to maintain competitive accuracy and significantly enhance the models' operational efficiency.

Integrating quality diversity optimization principles into ensemble selection has proven particularly effective, yielding Pareto fronts that reflect an optimal balance between predictive performance and hardware efficiency. Our proposed variant of QDO-ES, Infer-QDO-ES, has outperformed traditional ensemble methods by producing ensembles that are not only diverse and high-performing but also tailored for specific hardware limitations. These findings underscore the importance of considering hardware constraints in the ensemble selection process, particularly in resource-constrained deployment environments.

The implications of this research extend beyond the immediate improvements in ensemble selection; they suggest a paradigm shift in how AutoML systems are designed and deployed. By demonstrating that hardware-aware ensembles can achieve comparable or even superior performance to traditional methods, this study paves the way for more sustainable and efficient AutoML solutions, reducing both computational costs and environmental impacts.

Our work is limited to the evaluation of data from TabRepo and classification tasks. Additionally, we only considered two hardware constraints, omitting others like FLOPs, area, and energy consumption. Finally, the effectiveness of hardware-aware *model selection* compared to our *post hoc* hardware-aware ensemble selection approach remains unclear. For future work, we suggest several directions: (1) To broaden their applicability, extend hardware-aware ensemble methods to other types of machine learning tasks, such as regression. (2) Further exploring hardware constraints or multiple constraints simultaneously, such as energy consumption or memory usage, to better tailor models to deployment scenarios. (3) Investigating these methods' integration and potential user interface into mainstream AutoML frameworks, making them accessible to a wider range of users.

In conclusion, hardware-aware ensemble selection represents a promising advancement in AutoML. It aligns model performance with the practical realities of deployment environments. This approach not only enhances the practicality of AutoML solutions but also contributes to the broader goal of making machine learning more accessible and sustainable.

Broader Impact Statement. After careful reflection, we determined that this work does not have new negative broader impacts that are not already present for existing state-of-the-art AutoML systems. However, we hope that our work will have a positive, broader impact by enabling practitioners to balance costs and profits when applying AutoML systems. Thus, practitioners can select, e.g., less energy-consuming ensembles with only minimal loss in performance.

References

Bader, J. and Zitzler, E. (2011). Hype: An algorithm for fast hypervolume-based many-objective optimization. *Evolutionary computation*, 19(1):45–76.

Benmeziane, H., Maghraoui, K. E., Ouarnoughi, H., Niar, S., Wistuba, M., and Wang, N. (2021). A comprehensive survey on hardware-aware neural architecture search.

Biscani, F. and Izzo, D. (2020). A parallel global multiobjective framework for optimization: pagmo. *Journal of Open Source Software*, 5(53):2338.

Caruana, R., Niculescu-Mizil, A., Crew, G., and Ksikes, A. (2004). Ensemble selection from libraries of models. In *Twenty-first international conference on Machine learning - ICML '04*, ICML '04. ACM Press.

Cavalcanti, G. D., Oliveira, L. S., Moura, T. J., and Carvalho, G. V. (2016). Combining diversity measures for ensemble pruning. *Pattern Recognition Letters*, 74:38–45.

Chatzilygeroudis, K., Cully, A., Vassiliades, V., and Mouret, J.-B. (2021). *Quality-Diversity Optimization: A Novel Branch of Stochastic Optimization*, pages 109–135.

Cully, A., Clune, J., Tarapore, D., and Mouret, J.-B. (2015). Robots that can adapt like animals. *Nature*, 521(7553):503–507.

Erickson, N., Mueller, J., Shirkov, A., Zhang, H., Larroy, P., Li, M., and Smola, A. (2020). Autogluon-tabular: Robust and accurate automl for structured data.

Feurer, M., Eggensperger, K., Falkner, S., Lindauer, M., and Hutter, F. (2020). Auto-sklearn 2.0: Hands-free automl via meta-learning. *arXiv:2007.04074 [cs.LG]*.

Li, N., Yu, Y., and Zhou, Z.-H. (2012). Diversity regularized ensemble pruning. In *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2012, Bristol, UK, September 24-28, 2012. Proceedings, Part I 23*, pages 330–345. Springer.

Martınez-Munoz, G. and Suárez, A. (2004). Aggregation ordering in bagging. In *Proc. of the IASTED International Conference on Artificial Intelligence and Applications*, pages 258–263. Citeseer.

Mouret, J.-B. and Clune, J. (2015). Illuminating search spaces by mapping elites. *ArXiv*, abs/1504.04909.

Partalas, I., Tsoumakas, G., and Vlahavas, I. (2010). An ensemble uncertainty aware measure for directed hill climbing ensemble pruning. *Machine Learning*, 81:257–282.

Partridge, D. and Yates, W. B. (1996). Engineering multiversion neural-net systems. *Neural computation*, 8(4):869–893.

Purucker, L. and Beel, J. (2023). Cma-es for post hoc ensembling in automl: A great success and salvageable failure.

Purucker, L., Schneider, L., Anastacio, M., Beel, J., Bischl, B., and Hoos, H. (2023). Q(d)o-es: Population-based quality (diversity) optimisation for post hoc ensemble selection in automl.

Salinas, D. and Erickson, N. (2023). Tabrepo: A large scale repository of tabular model evaluations and its automl applications.

Schneider, L., Pfisterer, F., Kent, P., Branke, J., Bischl, B., and Thomas, J. (2022). Tackling neural architecture search with quality diversity optimization. In *International Conference on Automated Machine Learning*, pages 9–1. PMLR.

Sukthanker, R. S., Zela, A., Staffler, B., Dooley, S., Grabocka, J., and Hutter, F. (2024). Multi-objective differentiable neural architecture search. *arXiv preprint arXiv:2402.18213*.

Van Veldhuizen, D. A., Lamont, G. B., et al. (1998). Evolutionary computation and convergence to a pareto front. In *Late breaking papers at the genetic programming 1998 conference*, pages 221–228. Citeseer.

Zhang, L. L., Yang, Y., Jiang, Y., Zhu, W., and Liu, Y. (2019). Fast hardware-aware neural architecture search.

Submission Checklist

1. For all authors...

    (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] The abstract and introduction accurately describe our contributions, which include the introduction of hardware-aware ensemble selection (HA-ES) and its evaluation of improving operational efficiency while maintaining competitive accuracy.

    (b) Did you describe the limitations of your work? [Yes] We described the limitations in the conclusion section, specifically the use of only classification datasets from TabRepo, the consideration of only two hardware constraints, and the lack of comparison with hardware-aware model selection methods.

    (c) Did you discuss any potential negative societal impacts of your work? [Yes] In the broader impact statement, we noted that our work does not introduce new negative impacts beyond those already present in existing AutoML systems. We highlighted potential positive impacts such as enabling more energy-efficient model selection.

    (d) Did you read the ethics review guidelines and ensure that your paper conforms to them? https://2022.automl.cc/ethics-accessibility/ [Yes] We have read the ethics review guidelines and ensured that our paper conforms to them.

2. If you ran experiments...

    (a) Did you use the same evaluation protocol for all methods being compared (e.g., same benchmarks, data (sub)sets, available resources)? [Yes] We used the same datasets, evaluation metrics, and computational resources for all ensemble selection methods compared in the experiments.

    (b) Did you specify all the necessary details of your evaluation (e.g., data splits, pre-processing, search spaces, hyperparameter tuning)? [Yes] We specified the use of data from TabRepo, the criteria for selecting datasets, and the evaluation metrics used, including the calculation of hypervolumes for Pareto fronts.

    (c) Did you repeat your experiments (e.g., across multiple random seeds or splits) to account for the impact of randomness in your methods or data? [Yes] We repeated our experiments across 10 different seeds per fold to ensure robustness and account for randomness.

    (d) Did you report the uncertainty of your results (e.g., the variance across random seeds or splits)? [Yes] We reported the average hypervolumes and included statistical comparisons to show the variance across datasets.

    (e) Did you report the statistical significance of your results? [Yes] We used critical difference plots and boxplots to show the statistical significance of our results, specifically comparing the hypervolumes obtained by different methods.

    (f) Did you use tabular or surrogate benchmarks for in-depth evaluations? [Yes] We used the TabRepo dataset, which provides a comprehensive tabular benchmark for evaluating our methods on multiple machine-learning tasks.

    (g) Did you compare performance over time and describe how you selected the maximum duration? [No] Performance over time was not the primary focus of our study. Instead, we compared the quality of Pareto fronts and the best-performing ensembles across different methods.

(h) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] The experiments were conducted on a variety of server configurations. The experiments ran for roughly four days.

(i) Did you run ablation studies to assess the impact of different components of your approach? [Yes] We included an ablation study by comparing the original QDO-ES with our proposed hardware-aware variants, Size-QDO-ES and Infer-QDO-ES, to demonstrate the impact of incorporating hardware constraints.

3. With respect to the code used to obtain your results…

(a) Did you include the code, data, and instructions needed to reproduce the main experimental results, including all requirements (e.g., `requirements.txt` with explicit versions), random seeds, an instructive `README` with installation, and execution commands (either in the supplemental material or as a URL)? [Yes] The GitHub link to the code repository is included in Appendix B. The repository contains `requirements.txt` and instructions for running the experiments.

(b) Did you include a minimal example to replicate results on a small subset of the experiments or on toy data? [Yes] A small example can be replicated by changing the context used for the experiments to a toy context provided by TabRepo (e.g., D244_F3_C1530_10). This allows results to be reproduced on a smaller scale.

(c) Did you ensure sufficient code quality and documentation so that someone else can execute and understand your code? [Yes] The code is documented with necessary comments and written expressively to ensure comprehensibility and ease of use.

(d) Did you include the raw results of running your experiments with the given code, data, and instructions? [Yes] A compressed file named full.zip, which includes the processed raw data ready for plotting, is provided.

(e) Did you include the code, additional data, and instructions needed to generate the figures and tables in your paper based on the raw results? [Yes] The necessary code and instructions for generating figures and tables are included in the provided repository.

4. If you used existing assets (e.g., code, data, models)…

(a) Did you citep the creators of used assets? [Yes] We cited the creators of the TabRepo dataset and other relevant works used in our study.

(b) Did you discuss whether and how consent was obtained from people whose data you're using/curating if the license requires it? [N/A] The datasets used are publicly available and do not require additional consent for use.

(c) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [Yes] We used publicly available datasets from TabRepo, which do not contain personally identifiable information or offensive content.

5. If you created/released new assets (e.g., code, data, models)…

(a) Did you mention the license of the new assets (e.g., as part of your code submission)? [Yes] The code is published under the MIT license, as detailed in the repository.

(b) Did you include the new assets either in the supplemental material or as a URL (to, e.g., GitHub or Hugging Face)? [Yes] The new assets, including code and data, are available in the repository linked in Appendix B and included in the supplemental material.

6. If you used crowdsourcing or conducted research with human subjects…

    (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

    (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

    (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

7. If you included theoretical results…

    (a) Did you state the full set of assumptions of all theoretical results? [N/A]

    (b) Did you include complete proofs of all theoretical results? [N/A]

A Additional Definitions

Pareto Optimality. : A solution is Pareto optimal if no other solution exists that can improve some objectives without worsening others. Such solutions are efficient and provide a set of equally valid alternatives based on different priority scenarios (Van Veldhuizen et al., 1998).

Pareto Front. : The Pareto Front is defined as the collection of all solutions in a multiobjective optimization problem that are Pareto optimal. A solution is Pareto optimal if no other feasible solution exists that improves at least one objective without worsening another. Thus, the Pareto Front represents the boundary in the objective space beyond which no further improvements can be made without trade-offs (Van Veldhuizen et al., 1998).

Hypervolume. : Hypervolume measures the volume enclosed by the Pareto front and a reference point, capturing the extent of the objective space covered. It quantifies both the convergence to the Pareto front and the diversity of solutions, increasing as the set of solutions better approximates the true Pareto front (Bader and Zitzler, 2011).

B Resource Details

The experiments were conducted on a variety of server configurations at [removed], including 1. Dell R740xd and Asus ESC4000 with Xeon 6254/6354, 756GB/1TB RAM; 2. Dell R920 with E7-4880, 1TB RAM; 3. Dell R740 with Xeon 6134, 756GB RAM; and ran for roughly 4 days. We can use diverse hardware since TabRepo precalculates the inference times. Code for the experiments can be found at `https://github.com/Atraxus/HA-ES`.

C Tabrepo Details

In this section, we provide detailed statistics related to the distribution of dataset classes in the context of *D244_F3_C1530_100*. Table 1 enumerates the datasets according to the number of classes they contain. Entries are only included for class counts where datasets are available. The absence of a particular class count indicates that there are no corresponding datasets in this context.

Number of Classes	Count of Datasets
0	17
2	58
3	6
4	3
5	6
6	3
8	2
9	1
10	1
19	1
20	2

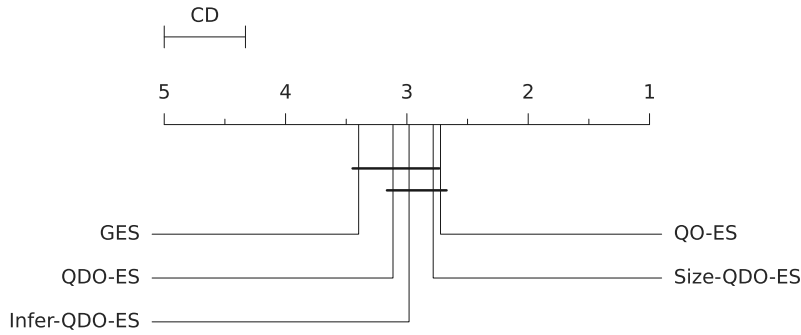Table 1: Distribution of datasets by number of classes

Figure 2: Critirical difference plot for *ROC AUC test score*. The top axis shows the average rank across all datasets for each method. QO-ES significantly differs from GES here (the horizontal bar does not touch QO-ES's line).
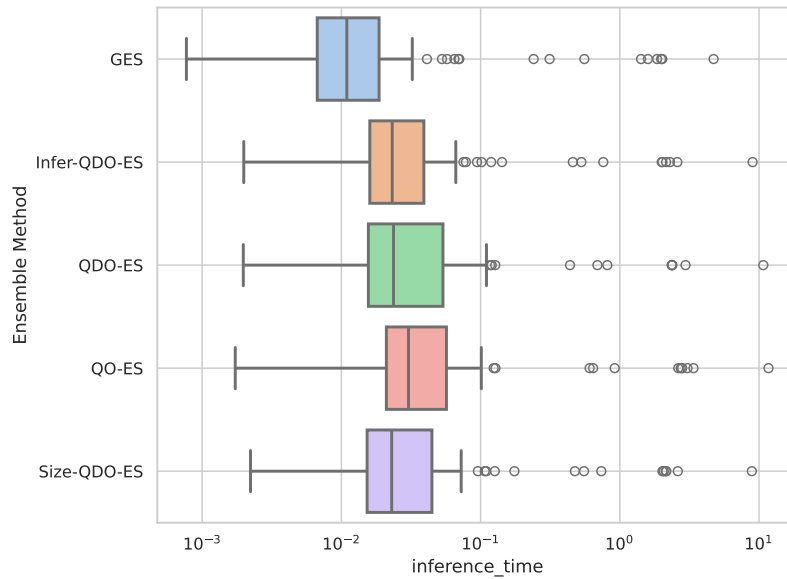

Figure 3: Distribution of the inference times in seconds of each method's best solution across datasets.

Table 2: Test ROC AUC - Binary: The mean and standard deviation of the test score over all folds for each method. The best methods per dataset are shown in bold. All methods close to the best method are considered best (using NumPy's default `isclose` function).

| Dataset	QDO-ES	Size-QDO-ES	Infer-QDO-ES	GES	QO-ES
madeline	0.9380(±0.0073)	0.9379(±0.0070)	0.9374(±0.0076)	0.9370(±0.0097)	**0.9395(±0.0064)**
kc2	0.8892(±0.0205)	0.8882(±0.0202)	0.8868(±0.0190)	0.8914(±0.0240)	**0.8914(±0.0206)**
fri_c2_500_50	0.9560(±0.0160)	0.9559(±0.0139)	0.9562(±0.0154)	0.9552(±0.0156)	**0.9573(±0.0139)**
ozone-level-8hr	0.9438(±0.0215)	0.9444(±0.0215)	0.9434(±0.0210)	**0.9457(±0.0208)**	0.9442(±0.0217)
analcatdata_dmf...	0.6015(±0.0320)	0.6022(±0.0317)	0.6013(±0.0331)	**0.6126(±0.0310)**	0.6045(±0.0278)
splice	0.9965(±0.0005)	**0.9965(±0.0005)**	0.9965(±0.0006)	0.9964(±0.0007)	0.9965(±0.0005)
colleges_usnews	**0.8152(±0.0243)**	0.8134(±0.0246)	0.8139(±0.0246)	0.8117(±0.0278)	0.8133(±0.0244)

| Dataset	QDO-ES	Size-QDO-ES	Infer-QDO-ES	GES	QO-ES
qsar-biodeg	0.9361(±0.0416)	0.9365(±0.0408)	0.9361(±0.0412)	0.9328(±0.0436)	**0.9372(±0.0411)**
volcanoes-a2	**0.9475(±0.0076)**	0.9468(±0.0082)	0.9468(±0.0083)	0.9474(±0.0077)	0.9472(±0.0074)
volcanoes-e1	0.8497(±0.0685)	0.8445(±0.0696)	0.8469(±0.0707)	0.8538(±0.0685)	**0.8580(±0.0628)**
ilpd	0.7255(±0.0614)	0.7239(±0.0611)	0.7247(±0.0625)	**0.7296(±0.0564)**	0.7273(±0.0588)
climate-model-s...	0.8911(±0.0814)	0.8923(±0.0801)	0.8916(±0.0796)	**0.8966(±0.0812)**	0.8907(±0.0811)
volcanoes-a4	**0.9063(±0.0191)**	0.9051(±0.0192)	0.9059(±0.0195)	0.9053(±0.0134)	0.9061(±0.0177)
hill-valley	0.9641(±0.0679)	0.9756(±0.0260)	0.9435(±0.0918)	0.9757(±0.0089)	**0.9788(±0.0079)**
fri_c4_500_100	0.9570(±0.0106)	0.9569(±0.0109)	0.9564(±0.0140)	**0.9591(±0.0096)**	0.9579(±0.0092)
OVA_Kidney	0.9978(±0.0019)	0.9978(±0.0019)	0.9977(±0.0020)	0.9978(±0.0019)	**0.9979(±0.0017)**
eucalyptus	0.9420(±0.0185)	0.9419(±0.0184)	0.9410(±0.0180)	0.9365(±0.0172)	**0.9433(±0.0186)**
autoUniv-au6-75...	0.6038(±0.0176)	**0.6070(±0.0152)**	0.6049(±0.0206)	0.5747(±0.0238)	0.6016(±0.0076)
LED-display-dom...	0.9632(±0.0092)	0.9632(±0.0092)	**0.9633(±0.0092)**	0.9588(±0.0089)	0.9631(±0.0096)
diabetes	0.8505(±0.0705)	0.8511(±0.0702)	**0.8511(±0.0678)**	0.8127(±0.0737)	0.8506(±0.0711)
pc3	0.8902(±0.0468)	0.8888(±0.0470)	0.8869(±0.0472)	0.8862(±0.0535)	**0.8914(±0.0473)**
GAMETES_Epistas...	0.7332(±0.0407)	0.7369(±0.0438)	0.7367(±0.0462)	**0.7598(±0.0226)**	0.7411(±0.0190)
fri_c0_1000_5	0.9753(±0.0088)	0.9752(±0.0089)	0.9750(±0.0089)	**0.9767(±0.0099)**	0.9755(±0.0086)
cmc	0.7429(±0.0307)	0.7426(±0.0306)	0.7426(±0.0307)	**0.7506(±0.0248)**	0.7424(±0.0310)
blood-transfusi...	0.7785(±0.0615)	0.7774(±0.0618)	0.7769(±0.0629)	**0.7874(±0.0551)**	0.7788(±0.0603)
kc1	0.8509(±0.0080)	0.8513(±0.0082)	0.8503(±0.0098)	**0.8556(±0.0089)**	0.8540(±0.0042)
wine-quality-re...	0.8811(±0.0105)	0.8798(±0.0129)	0.8796(±0.0126)	**0.8874(±0.0068)**	0.8849(±0.0090)
car	**1.0000(±0.0000)**	**1.0000(±0.0000)**	**1.0000(±0.0000)**	**1.0000(±0.0000)**	1.0000(±0.0001)
fri_c3_500_50	0.9561(±0.0225)	0.9562(±0.0240)	0.9558(±0.0247)	0.9548(±0.0290)	**0.9565(±0.0223)**
volcanoes-a3	0.9129(±0.0164)	0.9125(±0.0172)	0.9123(±0.0174)	0.9057(±0.0165)	**0.9133(±0.0157)**
parity5_plus_5	**1.0000(±0.0000)**	**1.0000(±0.0000)**	0.9993(±0.0166)	**1.0000(±0.0000)**	**1.0000(±0.0000)**
OVA_Prostate	0.9988(±0.0018)	0.9989(±0.0018)	0.9987(±0.0019)	**0.9993(±0.0019)**	0.9988(±0.0018)
analcatdata_aut...	**1.0000(±0.0001)**	**1.0000(±0.0001)**	**1.0000(±0.0001)**	**1.0000(±0.0001)**	1.0000(±0.0002)
synthetic_contr...	**1.0000(±0.0000)**	**1.0000(±0.0000)**	**1.0000(±0.0000)**	1.0000(±0.0001)	1.0000(±0.0001)
autoUniv-au7-11...	0.7209(±0.0068)	0.7200(±0.0089)	0.7186(±0.0108)	0.7134(±0.0135)	**0.7213(±0.0042)**
arsenic-female-...	0.8231(±0.0125)	0.8225(±0.0139)	0.8228(±0.0171)	0.8224(±0.0127)	**0.8245(±0.0087)**
baseball	0.9711(±0.0092)	0.9708(±0.0095)	0.9705(±0.0093)	**0.9721(±0.0109)**	0.9713(±0.0093)
fri_c3_1000_25	0.9771(±0.0212)	0.9770(±0.0217)	0.9755(±0.0223)	**0.9796(±0.0178)**	0.9779(±0.0207)
arcene	0.9205(±0.1119)	0.9231(±0.1076)	0.9239(±0.1064)	**0.9357(±0.1086)**	0.9181(±0.1162)
UMIST_Faces_Cro...	**1.0000(±0.0001)**	**1.0000(±0.0001)**	**1.0000(±0.0001)**	1.0000(±0.0002)	1.0000(±0.0002)
dna	0.9963(±0.0015)	0.9963(±0.0015)	**0.9963(±0.0015)**	0.9960(±0.0016)	**0.9963(±0.0014)**
pc4	0.9449(±0.0190)	**0.9451(±0.0197)**	0.9446(±0.0192)	0.9435(±0.0208)	0.9448(±0.0209)
MiceProtein	0.9999(±0.0004)	**0.9999(±0.0004)**	**0.9999(±0.0003)**	0.9991(±0.0017)	0.9999(±0.0004)
boston_correcte...	0.9689(±0.0210)	0.9693(±0.0212)	0.9681(±0.0211)	0.9670(±0.0212)	**0.9696(±0.0210)**
tokyo1	**0.9815(±0.0062)**	0.9814(±0.0063)	0.9814(±0.0063)	0.9802(±0.0064)	0.9815(±0.0061)
GAMETES_Epistas...	0.7498(±0.0198)	**0.7533(±0.0177)**	0.7464(±0.0359)	0.7492(±0.0130)	0.7446(±0.0150)
fri_c3_1000_10	0.9808(±0.0071)	0.9803(±0.0076)	0.9806(±0.0075)	0.9799(±0.0063)	**0.9809(±0.0076)**
credit-g	**0.7898(±0.0452)**	0.7883(±0.0447)	0.7885(±0.0461)	0.7846(±0.0424)	0.7875(±0.0439)
no2	0.7589(±0.0539)	0.7593(±0.0514)	0.7582(±0.0511)	0.7540(±0.0593)	**0.7604(±0.0534)**
balance-scale	0.9982(±0.0091)	0.9999(±0.0006)	0.9984(±0.0075)	1.0000(±0.0001)	**1.0000(±0.0001)**
madelon	0.9319(±0.0056)	0.9326(±0.0046)	0.9321(±0.0056)	0.9314(±0.0062)	**0.9337(±0.0031)**
fri_c3_500_10	**0.9751(±0.0151)**	0.9748(±0.0159)	0.9746(±0.0168)	0.9694(±0.0215)	0.9747(±0.0153)
cylinder-bands	0.9467(±0.0139)	0.9445(±0.0133)	0.9462(±0.0159)	0.9410(±0.0119)	**0.9474(±0.0111)**
GAMETES_Epistas...	**0.7013(±0.0178)**	0.7007(±0.0189)	0.6994(±0.0202)	0.6940(±0.0196)	0.7001(±0.0170)
soybean	**0.9986(±0.0006)**	0.9985(±0.0006)	0.9985(±0.0007)	0.9983(±0.0007)	**0.9986(±0.0006)**
autoUniv-au1-10...	0.6610(±0.0402)	0.6583(±0.0401)	0.6585(±0.0403)	**0.6618(±0.0317)**	0.6603(±0.0370)
GAMETES_Heterog...	0.7713(±0.0496)	0.7704(±0.0480)	0.7712(±0.0487)	**0.7737(±0.0461)**	0.7700(±0.0501)

Continued on next page

| Dataset	QDO-ES	Size-QDO-ES	Infer-QDO-ES	GES	QO-ES
fri_c1_1000_50	0.9781(±0.0063)	0.9778(±0.0066)	0.9767(±0.0142)	0.9763(±0.0049)	**0.9790(±0.0062)**
fri_c0_500_5	0.9510(±0.0200)	0.9514(±0.0189)	0.9514(±0.0192)	0.9508(±0.0189)	**0.9522(±0.0188)**
micro-mass	0.9987(±0.0009)	0.9987(±0.0009)	0.9987(±0.0010)	**0.9988(±0.0007)**	0.9988(±0.0008)
fri_c2_1000_25	0.9888(±0.0105)	0.9885(±0.0108)	0.9880(±0.0108)	0.9871(±0.0110)	**0.9889(±0.0110)**
dresses-sales	0.6274(±0.0616)	0.6263(±0.0601)	0.6257(±0.0621)	**0.6427(±0.0644)**	0.6255(±0.0605)
cnae-9	**0.9981(±0.0018)**	0.9977(±0.0023)	0.9978(±0.0022)	0.9980(±0.0017)	0.9980(±0.0019)
pc1	0.9132(±0.0404)	0.9111(±0.0413)	0.9121(±0.0417)	0.9038(±0.0371)	**0.9141(±0.0387)**
kdd_el_nino-sma...	0.9878(±0.0060)	0.9875(±0.0066)	0.9878(±0.0063)	0.9825(±0.0086)	**0.9882(±0.0057)**
Australian	**0.9363(±0.0159)**	0.9355(±0.0169)	0.9350(±0.0164)	0.9353(±0.0183)	0.9360(±0.0171)
Bioresponse	0.8872(±0.0051)	0.8864(±0.0059)	0.8852(±0.0065)	0.8863(±0.0051)	**0.8873(±0.0053)**
OVA_Colon	0.9711(±0.0243)	0.9713(±0.0233)	0.9732(±0.0230)	**0.9790(±0.0198)**	0.9709(±0.0236)
Titanic	0.8037(±0.0234)	**0.8051(±0.0250)**	0.8043(±0.0239)	0.7981(±0.0186)	0.8037(±0.0231)
vehicle	0.9640(±0.0090)	0.9638(±0.0093)	0.9626(±0.0101)	0.9644(±0.0071)	**0.9649(±0.0085)**
rmftsa_ladata	0.9763(±0.0057)	0.9762(±0.0059)	0.9751(±0.0059)	0.9764(±0.0065)	**0.9766(±0.0054)**
GAMETES_Heterog...	0.7822(±0.0231)	0.7824(±0.0222)	0.7812(±0.0227)	0.7789(±0.0261)	**0.7829(±0.0234)**
OVA_Ovary	0.9753(±0.0052)	0.9752(±0.0054)	0.9746(±0.0057)	0.9748(±0.0045)	**0.9753(±0.0050)**
pbcseq	0.9452(±0.0081)	0.9464(±0.0083)	**0.9469(±0.0092)**	0.9224(±0.0191)	0.9467(±0.0055)
jasmine	0.8953(±0.0200)	0.8949(±0.0200)	0.8941(±0.0206)	**0.8992(±0.0189)**	0.8962(±0.0208)
OVA_Lung	0.9752(±0.0254)	0.9745(±0.0245)	0.9749(±0.0244)	**0.9779(±0.0258)**	0.9760(±0.0245)
GAMETES_Epistas...	0.8465(±0.0113)	0.8466(±0.0115)	0.8461(±0.0141)	**0.8510(±0.0089)**	0.8461(±0.0102)
Internet-Advert...	0.9866(±0.0073)	0.9862(±0.0078)	0.9858(±0.0078)	**0.9876(±0.0073)**	0.9868(±0.0070)
meta	**0.9602(±0.0278)**	0.9588(±0.0283)	0.9582(±0.0303)	0.9549(±0.0326)	0.9576(±0.0279)
pm10	**0.5931(±0.0884)**	0.5886(±0.0825)	0.5905(±0.0882)	0.5668(±0.0814)	0.5890(±0.0756)
autoUniv-au7-70...	0.7140(±0.0331)	**0.7146(±0.0327)**	0.7130(±0.0346)	0.6984(±0.0276)	0.7120(±0.0289)
gina	0.9892(±0.0048)	0.9892(±0.0049)	0.9887(±0.0051)	**0.9902(±0.0045)**	0.9898(±0.0045)
OVA_Endometrium	0.9776(±0.0175)	0.9780(±0.0171)	0.9781(±0.0167)	0.9728(±0.0178)	**0.9789(±0.0147)**
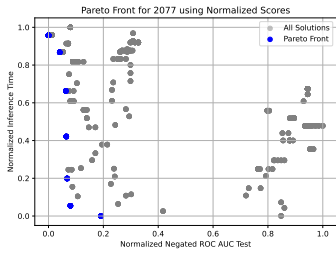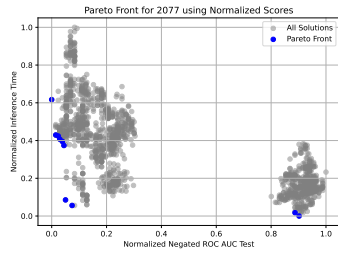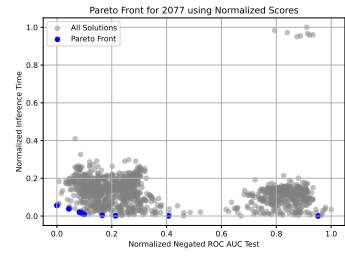
(a) GES   (b) QO-ES   (c) QDO-ES
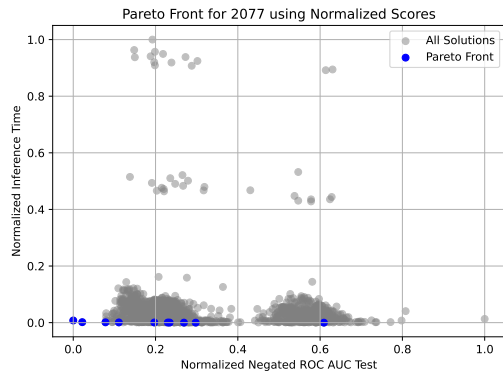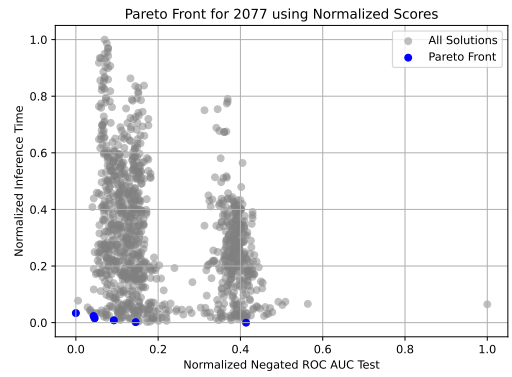
(d) Infer-QDO-ES   (e) Size-QDO-ES

Figure 4: Pareto Fronts (lower is better for both metrics) for the "baseball" dataset.