IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 14 February 2025, accepted 7 March 2025, date of publication 24 March 2025, date of current version 17 April 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3554093

RESEARCH ARTICLE

Understanding the Role of Diversity in Ensemble-Based AutoML Methods for Classification Tasks

SALOMEY OSEI[1], ANDRÉS R. MASEGOSA[2], AND ANTONIO D. MASEGOSA[1,3]
[1]DeustoTech, Faculty of Engineering, University of Deusto, 48007 Bilbao, Spain
[2]Department of Computer Science, Aalborg University, 2450 Copenhagen, Denmark
[3]Ikerbasque, Basque Foundation for Science, 48009 Bilbao, Spain

Corresponding author: Salomey Osei (osei.salomey@deusto.es)

This work was supported in part by European Union's Horizon 2020 Research and Innovation Programme under the Marie Skłodowska-Curie Grant 847624; in part by Spanish Ministry of Science and Innovation under Project PID2022-140612OB-I00; and in part by the Basque Government under Grant IT1564-22, Grant KK-2023/00012, and Grant KK-2023/00038.

ABSTRACT Ensemble-based Automated Machine Learning (AutoML) methods have gained prominence for their ability to combine diverse machine learning models, achieving superior generalization performance. Despite their empirical success, the underlying mechanisms driving this performance, particularly the role of model diversity, are not yet adequately understood. This study uses novel theoretical frameworks related to the role of diversity in ensembles, which were recently proposed, to shed light on this issue. In this work, we focus on AutoML methods for classification tasks. We use AUTO-SKLEARN (a widely used AutoML ensemble-based method) as a basis. More specifically, we examine how individual model diversity and performance evolves across the four key phases of AUTO-SKLEARN (base-learners, meta-learning, Bayesian Optimization (BO), and Caruana Ensemble). We also examine how they contribute to the diversity and performance of the final ensemble produced by the AutoML method. Using datasets from the AutoML benchmark, we empirically validate these insights by analyzing error rates and diversity measures across the mentioned phases. Our findings highlight the trade-off between individual model accuracy and ensemble diversity, showing that phases like BO improve the mean error rate of classifiers by nearly 50% percent but reduce their mean diversity by 20%. However, the Caruana phase increases the diversity by a 50% compared to the BO phase, allowing better generalization despite the higher mean error rate of the selected individual models (48% higher than BO). This work provides theoretical and empirical evidence that diversity is critical to the success of ensemble-based AutoML methods and a deeper understanding of diversity's impact on generalization performance and the role of the different AutoML phases. These findings can contribute to advance the development of more robust and theoretically grounded AutoML frameworks.

INDEX TERMS Automated machine learning (AutoML), AUTO-SKLEARN, Bayesian optimization (BO), diversity, ensemble learning.

I. INTRODUCTION

Understanding Machine Learning (ML) has long been challenging for non-experts, especially for business analysts and non-technical users who may lack the technical expertise needed to build and deploy ML models. This difficulty has not only created a barrier for those interested in engaging with ML but has also increased demand for skilled professionals in the field. To address the need for broader accessibility, the concept of Automated Machine Learning (AutoML) has emerged as a transformative solution, combining automation with machine learning to make ML more approachable for a wider audience [1].

AutoML plays a pivotal role in simplifying and automating the processes involved in traditional ML. Unlike traditional ML methods that demand significant technical expertise

The associate editor coordinating the review of this manuscript and approving it for publication was Muammar Muhammad Kabir.

and manual effort, AutoML streamlines these complexities, making AI more accessible to a broader audience. Traditionally, constructing and deploying ML models required a deep understanding of complex algorithms and extensive programming skills. AutoML alleviates these barriers by automating various stages of the ML pipeline, such as feature engineering, algorithm selection, and hyperparameter tuning [2]. Essentially, AutoML algorithms delve into the vast possibilities of ML configurations, identifying optimal settings and facilitating the development of cost-effective solutions. It is widely acknowledged that no single ML method consistently achieves high performance across all possible learning tasks [3]. The performance of a single learner can vary significantly across diverse data-driven problems. Consequently, one advanced strategy to address this challenge is to integrate multiple ML methods and aggregate or combine their outcomes. This approach of aggregating methods (Ensemble technique) is well known and used in many different fields to improve on general performances of models.

Among the various AutoML approaches, Ensemble AutoML stands out as one of the most widely used and effective techniques [4], [5], [6]. In the context of AutoML, Ensemble AutoML leverages the strengths of different algorithms and models, mitigating the weaknesses of individual components. This approach enhances the reliability and efficiency of ML models, making them more resilient to noisy data and better equipped to handle the complexities of diverse datasets [7], [8]. As a result, Ensemble AutoML has become a cornerstone in the AutoML landscape, contributing significantly to the success and widespread adoption of AutoML methodologies.

Despite the success and widespread use of ensemble-based AutoML methods, our understanding of how and why these methods work remains limited [9]. For example, most successful approaches employ multiple phases to create the ensemble, some involving techniques like Bayesian optimization. It is known that if these phases are omitted, performance deteriorates, but the reasons for this are unclear. Furthermore, these approaches often utilize a wide range of different classifiers to adapt to various kinds of data. Paradoxically, the ensemble composition typically includes classifiers that are not optimal for the specific data at hand, which seems contradictory. However, excluding these suboptimal classifiers from the ensemble also degrades performance, and we do not understand why this occurs. The main issue is that the design of these learning algorithms is conducted in a purely empirical manner. There is no theoretical analysis in this domain to explain why these approaches work and to guide the development of new methods. Our paper aims to fill this gap by leveraging recently introduced theoretical tools that explain the generalization performance of ensemble methods [10], [11], [12], [13].

To address this gap, our paper leverages recently introduced theoretical tools that link ensemble performance to the diversity of its constituent models [10], [11], [12], [13]. Specifically, we explore how the correlation structure among model predictions influences generalization performance. Building on these theoretical insights, we analyze the impact of diverse classifiers within AutoML ensembles, aiming to bridge the gap between empirical success and theoretical understanding. Here, it is important to mentioned that in this paper, we only focus on ensemble-based AutoML methods for classification tasks.

Our contributions are threefold:

- We apply recent theoretical frameworks to analyze the role of diversity in ensemble-based AutoML methods, offering new insights into how diverse classifiers enhance the performance of these methods.
- We examine the role of different AutoML phases in ensemble quality, shedding light on the mechanisms driving their effectiveness.
- We provide empirical validation using datasets from the AutoML benchmark [14], evaluating diversity measures and their relationship with ensemble performance across different stages of AutoML methods.

This paper is organized as follows. Section II reviews related work on ensemble methods and provides an overview of the AUTO-SKLEARN framework. Section III presents the theoretical foundations for understanding ensemble diversity. Section IV details our methodology for assessing diversity and evaluating pipeline phases. Section V reports empirical findings from analyzing ensemble-based AutoML methods. Finally, Section VI concludes with recommendations and directions for future research.

II. BACKGROUND AND RELATED WORK

AutoML seeks to automate the various stages in ML particularly data preprocessing, model selection, hyperparameter tuning, and ensemble learning. The systems uses a combination of optimization techniques to produce high quality models with little efforts from humans. The data preprocessing stage includes tasks like missing value imputation, scaling, encoding of variables and feature extraction to ensure the right format for the algorithm. The model selection involves the exploration of suitable algorithms from a variety of them for a given task based on performance. In the hyperparameter tuning, the aim is to automate the tuning of learning rate, e.t.c. with approaches such as Bayesian optimization. The ensemble systems involve evaluating several models to improve on the performance. The AutoML method types are categorized based on the stages of the ML process. The main types include:

1) **Hyperparameter Optimization (HPO):** This involves the tuning of hyperparameters in search of the best configuration for a model [15]. Some of the methods for this task include grid search, random search, Bayesian optimization and hyperband.
2) **Neural Architecture Search (NAS):** This is the automatic design of the architecture of a neural network with methods such as reinforcement learning,

evolutionary algorithms and differentiable NAS. e.g. includes Auto-Keras [16].

3) **Combined Algorithm Selection and Hyperparameter Optimization (CASH):** This process helps in the selection of the best hyperparameters and algorithm simultaneously [7]. Some of the methods used are random search and Bayesian optimization with examples including AUTO-SKLEARN [4] and TPOT [8].
4) **Pipeline Optimization:** Methods like TPOT, AUTO-SKLEARN, H2O AutoML [5] automates the full ML pipeline using genetic programming, meta-learning and Bayesian optimization respectively [8].
5) **Feature Engineering Automation:** The idea is to automatically create and select the most important features for ML methods [17]. Some of the methods include FeatureTools [18] which automatically generates deep features from relational data to Deep Feature Synthesis [17] which automatically generate features by combining the existing ones and Embedded methods [19] which selects features based on their importance to the models during the training. An example is DataRobot [20].
6) **Ensemble Learning Automation:** This is where multiple models are automatically combined to enhance prediction performance [4]. The methods includes bagging [21](random forest utilizes this method), boosting [22] (e.g. XGBoost [23]), blending and stacking [24]. Examples include H20 AutoML and AUTO-SKLEARN.
7) **Transfer Learning in AutoML:** This leverages pre-trained models and prior knowledge to adapt to new tasks faster. For instance, meta-learning learns from previous tasks to generalize on new ones whiles transfer learning uses pre-trained models and fine tunes them on the new data. Examples include Google AutoML and Auto-Keras [16].
8) **Evolutionary and Genetic Algorithms:** Uses methods such as genetic programming and evolutionary Neural Architecture Search to optimize ML models and pipelines [25]. Examples include TPOT and Deep Evolutionary Learning (DEvol) [26].
9) **Meta-Learning (Learning to Learn):** Methods like Few-shot Learning [27] uses small amount of data to adapt quickly to new tasks whereas Meta-Feature Extraction [28] learns a task-specific features which can be reused for similar tasks. These methods are operating a learning to learn scheme where previous knowledge is sued for other tasks [29]. Example include AUTO-SKLEARN.
10) **End-to-End AutoML Systems:** Methods like Google Cloud AutoML [30] (NAS to automate the process of finding effective neural architectures [31]), Azure AutoML [32] and Amazon SageMaker AutoPilot [33] provides fully automated systems that handles everything from data preprocessing to model deployment for cloud based systems. The aim is to fully automate the entire ML process. Other examples include TPOT, AUTO-SKLEARN and Auto-keras.

A. ENSEMBLE AUTOML AND AUTO-SKLEARN

Ensemble AutoML is an approach within AutoML that focuses on creating and combining multiple machine learning models, or ensembles, to improve predictive performance. This approach leverages the strengths of various algorithms and mitigates their individual weaknesses, often leading to more robust and accurate models compared to single-model approaches. Ensemble methods are a well-established technique in machine learning, and their integration into AutoML frameworks enhances the automation process in several key ways:

1) **Diverse Model Generation:** Ensemble AutoML systems automatically train a variety of different models on a given dataset. These models can include different types of machine learning algorithms, like decision trees, support vector machines, neural networks, etc., each with their own sets of hyperparameters.
2) **Model Selection and Hyperparameter Tuning:** AutoML systems use techniques like Bayesian optimization, genetic algorithms, or grid/random search to find the optimal hyperparameters for each model. This process is automated, significantly reducing the manual effort required in traditional model tuning.
3) **Ensemble Construction:** After training multiple models, the system then automatically combines them into an ensemble. Techniques like bagging, boosting, and stacking are commonly used. For instance, it might use a weighted average of predictions from all models, where weights are assigned based on individual model performance.

Ensemble AutoML is particularly beneficial in scenarios where the optimal machine learning approach is not clear, or where the data is complex and a single model might not capture all the patterns in the data. By automating the process of creating and combining multiple models, Ensemble AutoML democratizes access to advanced machine learning techniques, enabling more users to develop high-performing models for their specific tasks.

B. THE AUTO-SKLEARN WORKFLOW

Among the various tools available for Ensemble AutoML, AUTO-SKLEARN has emerged as one of the most widely used. This popularity can be attributed to several key factors. Firstly, AUTO-SKLEARN is built on top of scikit-learn, one of the most popular machine learning libraries, which makes it accessible and familiar to a broad base of users. Secondly, it incorporates a robust ensemble method that combines models through Bayesian optimization and meta-learning, leading to high-performance outcomes. Furthermore, AUTO-SKLEARN's automated approach to model selection and

hyperparameter tuning significantly reduces the complexity and time required for model development.

AUTO-SKLEARN versions one (1) and two (2) won the first and second calls of the 2015 ChaLearn AutoML challenge respectively. AUTO-SKLEARN has three parts mainly meta-learning, optimization (Bayesian) and a greedy ensemble (which employs the [9] method). The default structure of the AUTO-SKLEARN which makes it more competitive as compared to other methods is the inclusion of a meta-learning step of 38 meta-features from the offline phase whose main aim is to warm-start the Bayesian optimization with instantiations of ML frameworks that are likely to perform well on new datasets. Below we explain the four (4) phases in AUTO-SKLEARN.

1) BASE-LEARNERS PHASE

Base-learners, also referred to as weak learners, represent individual learning algorithms whose combination results in an enhanced ensemble performance. While these learners may not exhibit high predictive accuracy independently, their purpose lies in capturing diverse perspectives of the data for predictive purposes. This approach surpasses random guessing, as the collective insights of these base-learners contribute to an overall improvement in performance. Their notable contribution to diversity arises from being trained with distinct initializations and subsets of the training data. Furthermore, the independence in their training processes introduces unique knowledge to the ensemble, collectively mitigating biases and variances while reducing the risk of overfitting. This phase strategically aims to broaden the pool of candidate models, ensuring the final ensemble benefits from a diverse and well-informed set of contributors. AUTO-SKLEARN leverages the concept of base-learners or weak learners to enhance its model-building capabilities. The use of base-learners in AUTO-SKLEARN is rooted in the fundamental principles of ensemble learning, where combining multiple models often leads to improved predictive performance compared to individual models. While each base-learners may not excel on its own, the collective wisdom of these models enables the ensemble to generalize well to various datasets and make robust predictions. This ensemble approach helps mitigate biases, reduce overfitting, and enhance the overall stability and reliability of the AUTO-SKLEARN system.

2) META-LEARNER

Meta-learners uses the potential of multiple base-learners by combining their outputs through various averaging techniques or learning-based methods to formulate predictions. This exploitation of diversity and knowledge among the base-learners aims to enhance the overall performance of the ensemble. Unlike base-learners, which individually generate predictions, meta-learners take these predictions and determine optimal ways to combine them. AUTO-SKLEARN adopts this strategy as a complementary step to Bayesian optimization, particularly useful for expediting hyperparameter search initiation to be able to perform well on new dataset. Given the initial slowness of Bayesian optimization, an offline phase within AUTO-SKLEARN trains on diverse datasets, extracting meta-features for specific datasets. These meta-features, totaling 38 in this implementation, are later utilized by Bayesian optimization to suggest well-performing instantiations on new datasets. The primary motivation behind incorporating a meta-learner lies in optimizing the combination of outputs from multiple base-learners within the ensemble. While base-learners individually contribute diverse perspectives and predictions, a meta-learner serves the crucial role of learning how to effectively blend these predictions to produce a more accurate and robust final model. The offline phase in AUTO-SKLEARN involves training on different datasets and extracting meta-features, which are indicative of the most appropriate algorithm for a given dataset.

3) BAYESIAN OPTIMIZATION

Bayesian Optimization, derived from the Bayes theorem, provides a robust solution for tackling challenging ''black box'' optimization problems, enhancing search speed by leveraging past performances [34] or when the evaluation of the objective function is computationally expensive. This method emulates manual hyperparameter search, systematically refining changes until optimal parameters are identified, albeit with the efficiency of the Bayesian optimization algorithm [34]. Particularly beneficial for optimizing expensive evaluations of objective functions, Bayesian optimization excels at proposing competitive hyperparameter candidates within a limited number of evaluations, enhancing the overall performance of base-learners and ensembles [35]. The optimization loop involves conditioning the evaluations of past objective functions on a likelihood and prior for each iteration, leading to the computation of a posterior distribution [36]. This utilization of prior knowledge streamlines the sampling process and reduces computations in the search space [35]. Bayesian optimization is efficient for evaluations and adept at balancing the exploration-exploitation trade-off [35]. The beliefs about the objective function are mapped using an ''acquisition function'' that guides exploration of the search space, aiming to maximize this function. Once achieved, the surrogate model is updated, and a new search iteration begins [35]. This framework has been adapted for pipeline selection and recommendation in AutoML methods [4]. It was observed that Bayesian optimization is underutilized when the primary goal is to select only the best-performing model post-training, leading to the discarding of highly efficient models with performance comparable to the best [37]. An efficient strategy to utilize these models is the construction of an ensemble for the top-performing models [4]. AUTO-SKLEARN's use of Bayesian optimization extends beyond hyperparameter tuning; it plays a crucial role in pipeline selection and recommendation within the AutoML context. The approach ensures that the system can efficiently navigate the complex landscape of

TABLE 1. Overview of the 20 datasets used in the study, including details such as the number of features (numerical and categorical), instances, missing values, and class distribution (majority and minority classes). These datasets were selected based on criteria like diversity, and real-world representativeness.

Dataset	Features	Instances	Num. Feat	Cat. Feat	Missing Values	Classes	Majority Class	Minority Class
Air Pressure System Failure (1)	171	76,000	170	1	1,078,695	2	59,000	1,000
King-Rook vs King-Pawn (3)	37	3,196	0	37	0	2	1,669	1,527
Credit-G (5)	21	1,000	7	14	0	2	700	300
Sylvine (14)	21	5,124	20	1	0	2	2,562	2,562
Albert (16)	79	425,240	26	53	2,734,000	2	212,620	212,620
Christine (19)	1,637	5,418	1,599	38	0	2	2,709	2,709
jasmine (18)	145	2984	8	137	0	2	1492	1492
KC1 (21)	22	2,109	21	1	0	2	1,783	326
Miniboone (23)	51	130,064	50	1	0	2	93,565	36,499
Blood-Transf (24)	5	748	4	1	0	2	570	178
Phoneme (26)	6	5,404	5	1	0	2	3,818	1,586
Nomao (27)	119	34,465	89	30	0	2	24,621	9,844
Numerai28.6 (29)	22	92,320	21	1	0	2	48,658	47,662
Bank-Marketing (30)	17	45,211	7	10	0	2	39,922	5,289
Adult (31)	15	48,842	6	9	6,465	2	37,155	11,687
Amazon Employee (33)	10	32,769	0	10	0	2	30,872	1,897
Car (25)	7	1,728	0	7	0	4	1,210	65
CNAE-9 (32)	857	1,080	856	0	0	9	120	120
Fabert (7)	801	8,237	800	1	0	6	1,927	502
Vehicle (6)	19	846	18	1	0	5	218	199

hyperparameter configurations and pipeline choices, enhancing the overall performance of base-learners and ensembles. Moreover, Bayesian optimization is adept at balancing the exploration-exploitation trade-off, contributing to effective decision-making in the selection of optimal models and configurations [35].

4) CARUANA ENSEMBLE

Initially introduced by Caruana [9], this method aims to enhance the performance of ensemble models through the amalgamation of multiple base models. Caruana's approach revolves around the selection of a suitable ensemble from a pool of candidate models, achieved by training a substantial number of base models on diverse subsets of the training data, followed by evaluating their performance on a validation set. This method is designed to strike a delicate balance between model accuracy and ensemble diversity. Notably, AUTO-SKLEARN adopts this ensemble strategy, leveraging the inherent performance boost that ensembles provide over individual models. Instead of discarding equally performing models obtained from Bayesian optimization, AUTO-SKLEARN embraces an automated ensemble construction approach. This does not only mitigates the risk of overfitting but also introduces diversity through the varied nature of the models identified by Bayesian optimization. To implement this strategy, AUTO-SKLEARN employs the greedy ensemble approach proposed by Caruana, constructing an ensemble of size 50 from the top-performing 50 models.

III. MEASURING DIVERSITY OF ENSEMBLES-BASED AUTOML

Ensemble methods combine multiple models to enhance predictive performance. Ensemble learning, a machine learning approach where multiple individual predictors are integrated, has demonstrated superior performance across various prediction tasks. A key factor in this improved performance is the diversity among the individual members of the ensemble. When models make different errors, they can complement each other, reducing the likelihood of similar mistakes [38].

In this context, diversity refers to the variability in predictions made by the individual models within the ensemble. Diverse models are more likely to make independent errors, and aggregating their predictions tends to mitigate these errors, thereby improving the ensemble's overall accuracy. This concept, while intuitive, is also supported by both empirical evidence and theoretical analysis.

Given the pivotal role of ensemble methods in AutoML systems, it is reasonable to hypothesize that diversity among the models is essential, as similar models are likely to make correlated errors. While ensemble methods have achieved significant success in practice, the lack of a theoretically principled approach to understanding the role of diversity in these systems represents a critical gap in the literature. Addressing this gap could provide deeper insights into the mechanisms driving the success of ensemble AutoML methods and guide the development of more effective strategies. Such analysis, currently missing from the existing research, is necessary to bridge the divide between empirical success and theoretical understanding, ensuring that diversity becomes a foundational element of future AutoML ensemble methodologies.

A recent theoretical framework [11], [13] has provided insights into the relationship between diversity and generalization performance in ensemble models. Building on prior research, this framework offers a thorough understanding

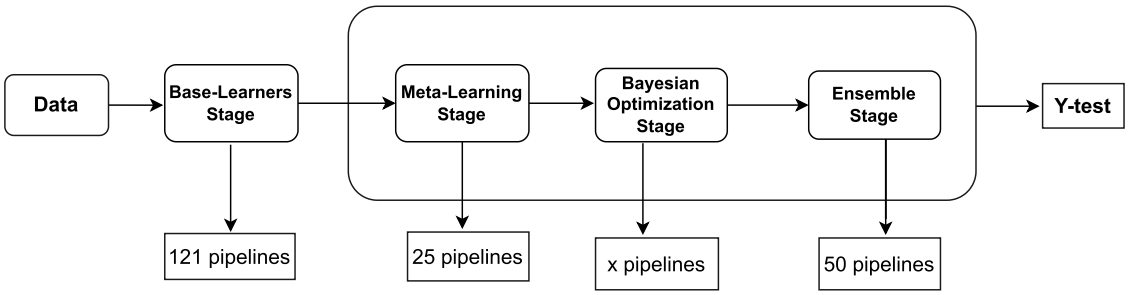

FIGURE 1. Stages that have been created in order to perform experiments. The first stage shows the 121 pipelines that exist in the offline stage of the meta-learning. The second stage also shows the 25 pipelines that are recommended by the Bayesian optimization. The third and final stage shows the pipelines that are generated in Bayesian optimization and the 50 pipelines in the ensemble respectively.

of how diversity influences ensemble performance across various methods, loss functions, and model combination strategies. A notable contribution of this work is the derivation of an upper bound on the ensemble's expected loss, articulated in terms of the average loss of the individual models and a newly introduced diversity measure.

This diversity measure is pivotal for grasping the impact of diversity on ensemble effectiveness. It adheres to intuitive properties, such as being zero when all ensemble members produce identical predictions or when there is no diversity in their outputs. Furthermore, the measure diminishes as the predictive error of individual models decreases, aligning with empirical observations.

Importantly, the framework establishes that greater diversity within an ensemble amplifies the gap between the test loss of individual models and the ensemble's test loss. This underscores that higher diversity enhances the benefits of combining models. Consequently, the effectiveness of an ensemble relative to a single model hinges on achieving a sufficiently large level of diversity.

Building on the theoretical framework for understanding the role of diversity in ensemble performance, our goal is to delve deeper into how this framework can be applied to analyze the phases of ensemble construction in AutoML methods for classification tasks, particularly those following Caruana's Ensemble building approach, like AUTO-SKLEARN. Specifically, we aim to explore the unique contributions of each phase—meta-learning, Bayesian optimization, and the ensemble construction stage—to the overall effectiveness of the ensemble. This includes addressing the following four key research questions such as: (RQ1) Why is it essential to have distinct phases in this process? (ii) What roles do meta-learning, Bayesian optimization, and the final ensemble phase play in enhancing model performance? (iii) How do these stages collectively contribute to building better ensembles? (iv) Are their efforts directed primarily at improving individual models, or do they explicitly or implicitly enhance the diversity of the ensemble? By investigating these aspects, we hope to uncover the mechanisms by which these phases interact to optimize ensemble diversity and generalization, thus providing a deeper understanding of their purpose and effectiveness.

Before delving into these questions, the next subsection outlines the methodology we will use to precisely measure diversity and error loss in the ensembles constructed at each phase of AUTO-SKLEARN AutoML approach.

A. ERROR AND DIVERSITY MEASURES

As discussed above, we will leverage the tools introduced in [11], [13] to precisely measure the diversity of an ensemble. Before describing the diversity measure, we want to point out that, as explained above, the error and diversity measures described here are not a contribution of this paper but a tool that we will use to analyze the role of diversity in AutoML Ensemble-based methods for classification tasks.

Let h_i denote a single predictor, with $h_i(x)$ representing its prediction for an input $x \in \mathcal{X}$. We denote Θ the set of K predictors defining the ensemble, $\Theta = \{h_1, \ldots, h_K\}$.

A *majority voting ensemble* 1 is a composite predictor formed from K predictors in Θ. The ensemble's final prediction is determined by the class receiving the highest number of *votes*. Formally, it can be expressed as:

$$h_\Theta(x) = \arg\max_{y' \in \mathcal{Y}} \sum_i \mathbf{1}[h_i(x) = y'], \tag{1}$$

where $\mathbf{1}[a]$ is the indicator function, equal to 1 if the condition a is true and 0 otherwise.

The expected error rate of an individual predictor h_i is defined as $L(h_i) = \mathbb{E}_\nu[\mathbf{1}[h_i(x) \neq y]]$, where $E_\nu[\cdot]$ denotes an expectation over the data generating distribution ν from which both the training and the test data are sampled. Similarly, the expected error rate of the ensemble predictor Θ is given by $L(\Theta) = \mathbb{E}_\nu[\mathbf{1}[h_\Theta(x) \neq y]]$.

Following the definitions in [11] and [13], the diversity between two predictors h_i and h_j within a majority voting ensemble Θ can be quantified as:

$$d(h_i, h_j) = \mathbb{E}_\nu\big[\mathbf{1}[h_i(x) \neq y]\, \mathbf{1}[h_j(x) = y]\big]. \tag{2}$$

This quantity measures the expected proportion of instances where h_i errs while h_j provides the correct prediction, capturing the anti-correlation in their errors. For

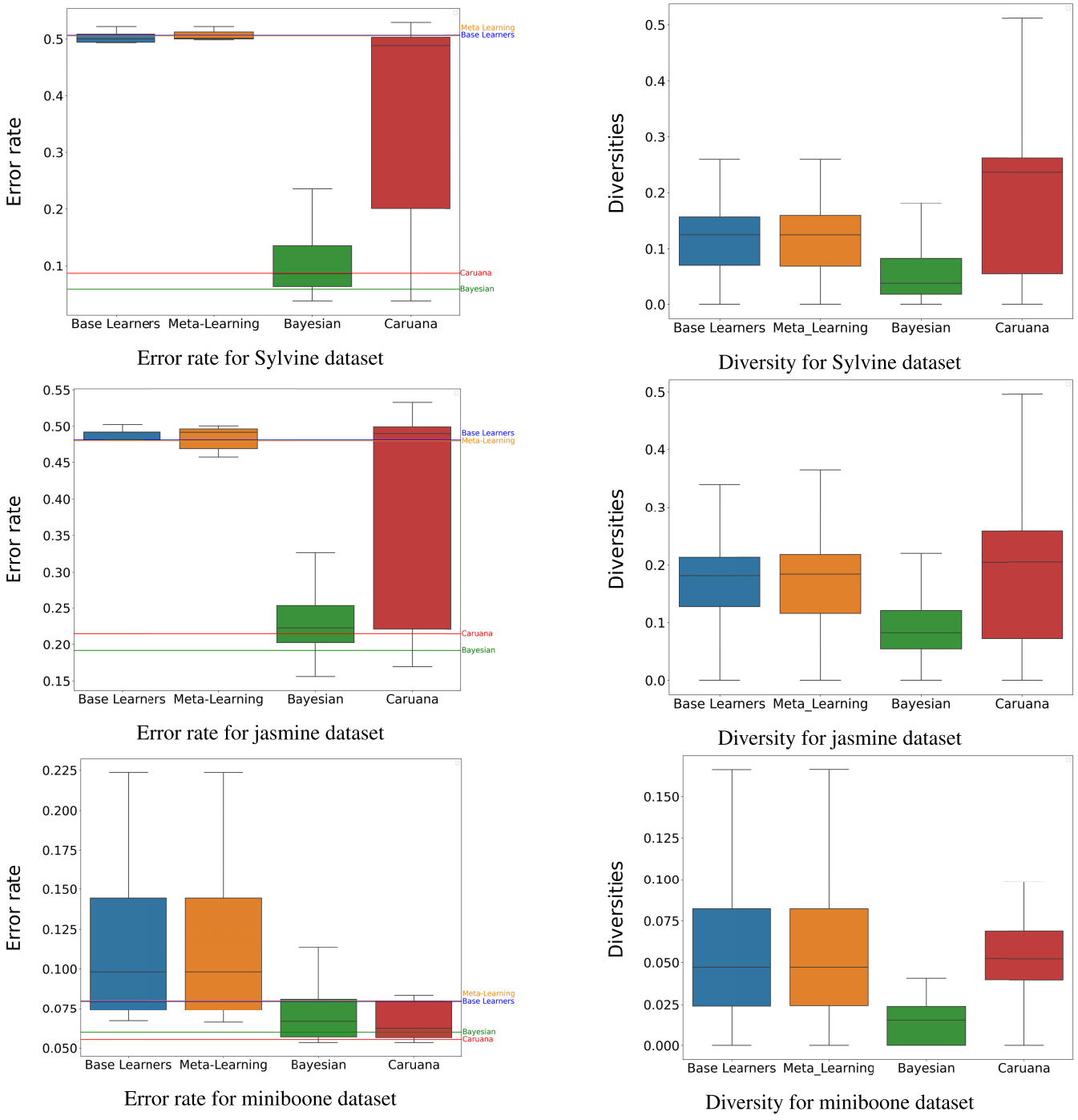

FIGURE 2. **Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: Sylvine, Jasmine, and Miniboone. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_i)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.**

identical models, the diversity is zero, i.e., $d(h_i, h_i) = 0$. If the errors of h_i and h_j are independent, the diversity becomes $d(h_i, h_j) = L(h_i)L(h_j)$.

The total diversity of an ensemble, denoted as $D(\Theta)$, is calculated as the average diversity across all possible pairs of predictors:

$$D(\Theta) = \frac{1}{K^2} \sum_{h_i, h_j \in \Theta} d(h_i, h_j), \tag{3}$$

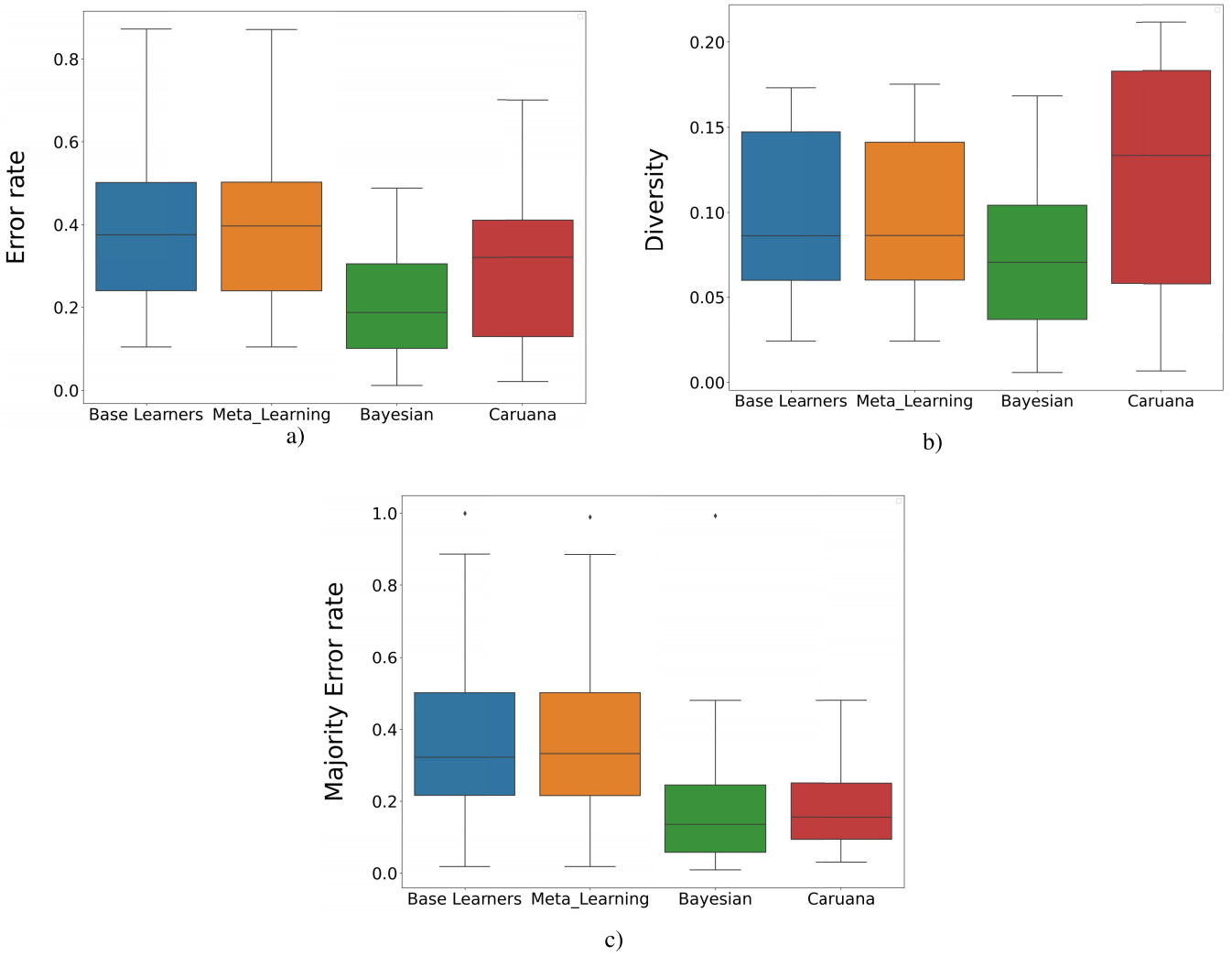

FIGURE 3. Analysis of pipeline error rates and diversity across the four stages of the AutoML framework for 20 datasets: (a) Average error rates of individual pipelines at each stage, highlighting performance differences across phases. (b) Diversity of pipelines across the stages, similarly highlighting performance differences across phases. (c) Majority vote ensemble error rates for each stage, showcasing the combined impact of pipeline performance and diversity on ensemble effectiveness.

where $h_i, h_j \in \Theta$ iterate over all pairs of predictors in the ensemble.

In [13], the relationship between the ensemble's error rate $L(\Theta)$, the error rates of its individual predictors $L(h_i)$, and the ensemble's diversity $D(\Theta)$ is captured by the following inequality:

$$L(\Theta) \leq 4\left(\frac{1}{K}\sum_i L(h_i) - D(\Theta)\right). \tag{4}$$

The term $\frac{1}{K}\sum_i L(h_i)$ in equation 4 represents the average error rate of the individual predictors that make up the ensemble. A set of strong predictors with low error rates contributes to the construction of a robust ensemble. However, the inequality also emphasizes the importance of diversity among the predictors: When $D(\Theta)$ is high, the ensemble benefits from reduced error rates due to the complementary nature of the predictors' errors, resulting in a more effective majority voting ensemble.

Here, it is important to mention that the error and diversity measures described above are specific for the majority vote and weighted voting ensembles. For this reason, the analysis and conclusions that will be shown in the following sections are only generalizable to categories of ensembles based on these two aggregation strategies, for example, Bagging [21] and Boosting [22], [23], but not for other ensemble methodologies that do not use these two aggregation strategies, for example, Stacking [24].

IV. EMPIRICAL ANALYSIS OF DIVERSITY IN ENSEMBLES-BASED AUTOML

A. EXPERIMENTAL DETAILS

1) DATASETS

For our experiments, we utilized 20 datasets from the AutoML benchmark [14], comprising 16 binary classification tasks and 4 multi-class classification tasks. This dataset by OpenML is a curated collection of datasets designed to

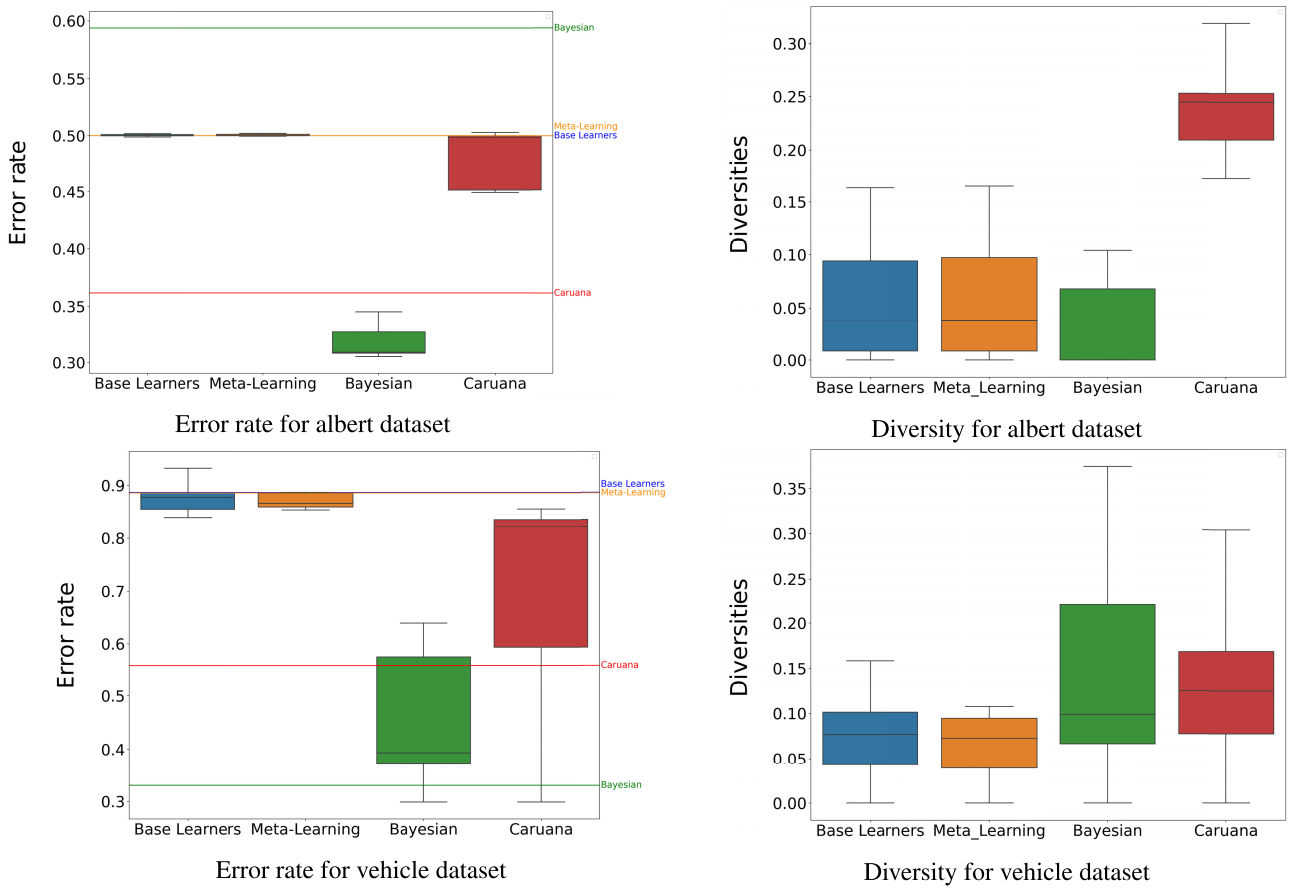

FIGURE 4. Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: albert, and vehicle. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_j)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.

evaluate and compare AutoML frameworks. It comprises datasets from diverse domains such as finance, healthcare, image recognition, and text classification, ensuring a broad representation of real-world challenges. The datasets vary in complexity, size, number of features, and class distributions, providing a robust testing ground for AutoML methods. Each dataset is preprocessed and standardized, ensuring consistency across evaluations. Additionally, the benchmark includes metadata such as the number of instances, features, missing values, and class imbalance levels, which are crucial for assessing AutoML methods. The detailed statistics of the datasets, including the number of samples, features, missing values, and the presence of categorical and numerical features, are summarized in Table 1. A significant advantage of this collection is its diversity, as it encompasses datasets with a wide range of sample sizes and feature counts, offering varied and challenging benchmarks for evaluating AutoML methods.

2) IMPLEMENTATION

We analyze how diversity evolves across the four main stages of the AUTO-SKLEARN framework as shown in Figure 1: Base-Learners, Meta-Learning, Bayesian Optimization, and the Caruana Ensemble. At each stage, we measure the diversity and error rates of the pipelines generated, aiming to understand their contributions to the overall ensemble performance. Our goal is to make these measures comparable across stages and evaluate the framework's effectiveness in enhancing diversity and performance.

Specifically, our empirical analysis focuses on: 1) Evaluating the performance of each step in the AUTO-SKLEARN framework; and 2) Measuring the error rate and diversity of classifiers at each stage. The experimental stages of our study, including Base-Learners, Meta-Learning, Bayesian optimization, and the Caruana, are illustrated in Figure 1 which provides a visual representation of how pipelines evolve through the different stages.

Stages of Analysis:

1. Base-Learners: In this stage, 121 pipelines are generated in the offline phase of meta-learning. We extract their predictions and compute diversity and error rates among these pipelines.

2. Meta-Learning: Meta-learning selects 25 pipelines from the base-learners, which are then recommended for

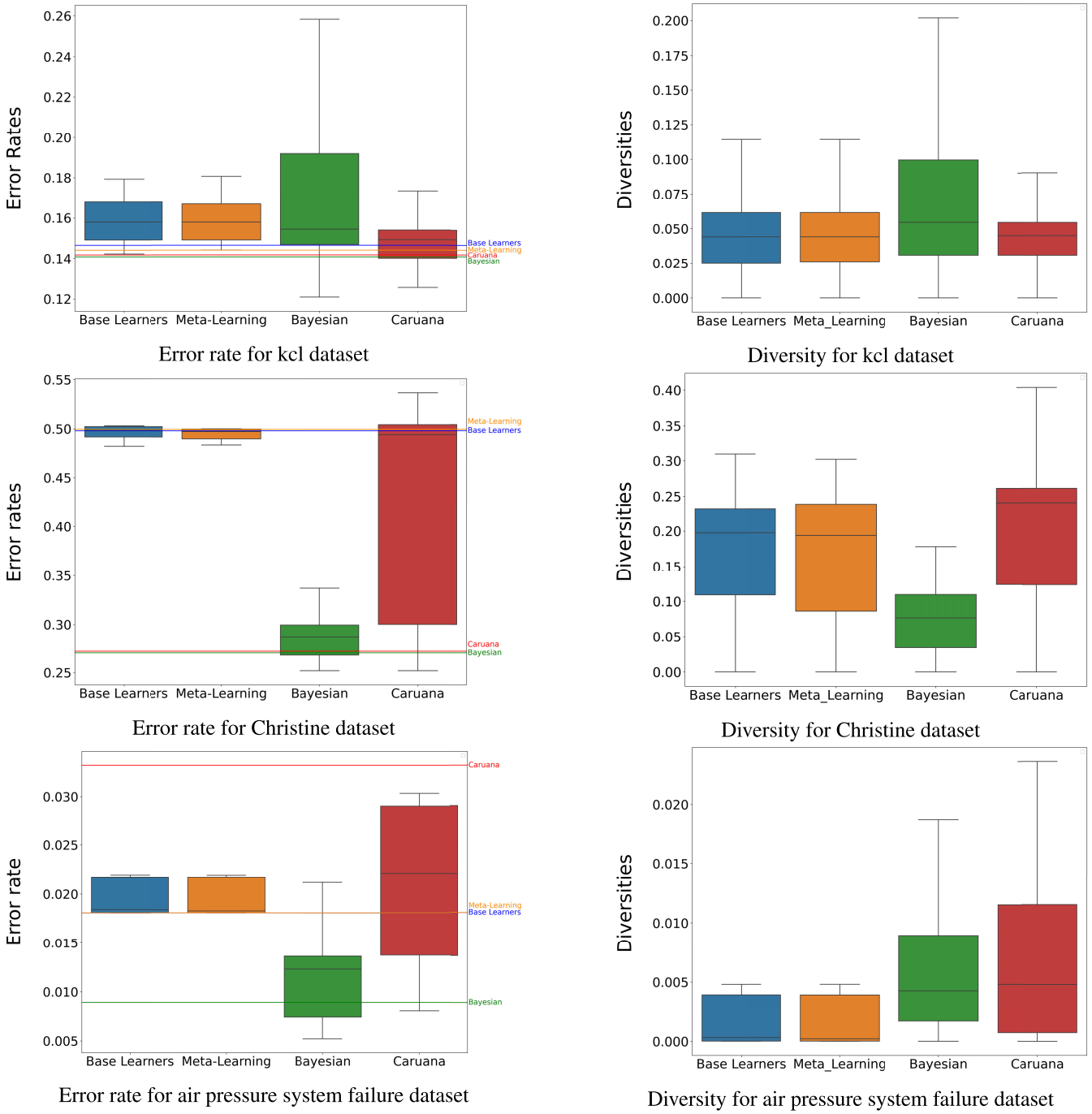

FIGURE 5. **Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: kcl, christine, and air pressure system failure. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_i)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.**

further optimization. We measure the diversity of the predictions from these pipelines.

3. Bayesian Optimization: Bayesian optimization generates pipelines with varying configurations based on the dataset. We analyze the diversity of the outputs produced during this optimization phase.

4. Caruana: The pipelines generated by Bayesian optimization are used to construct an ensemble based on Caruana's approach. Up to 50 pipelines are included in this ensemble by default. We calculate the diversity of the predictions among all pairs of pipelines in the ensemble.

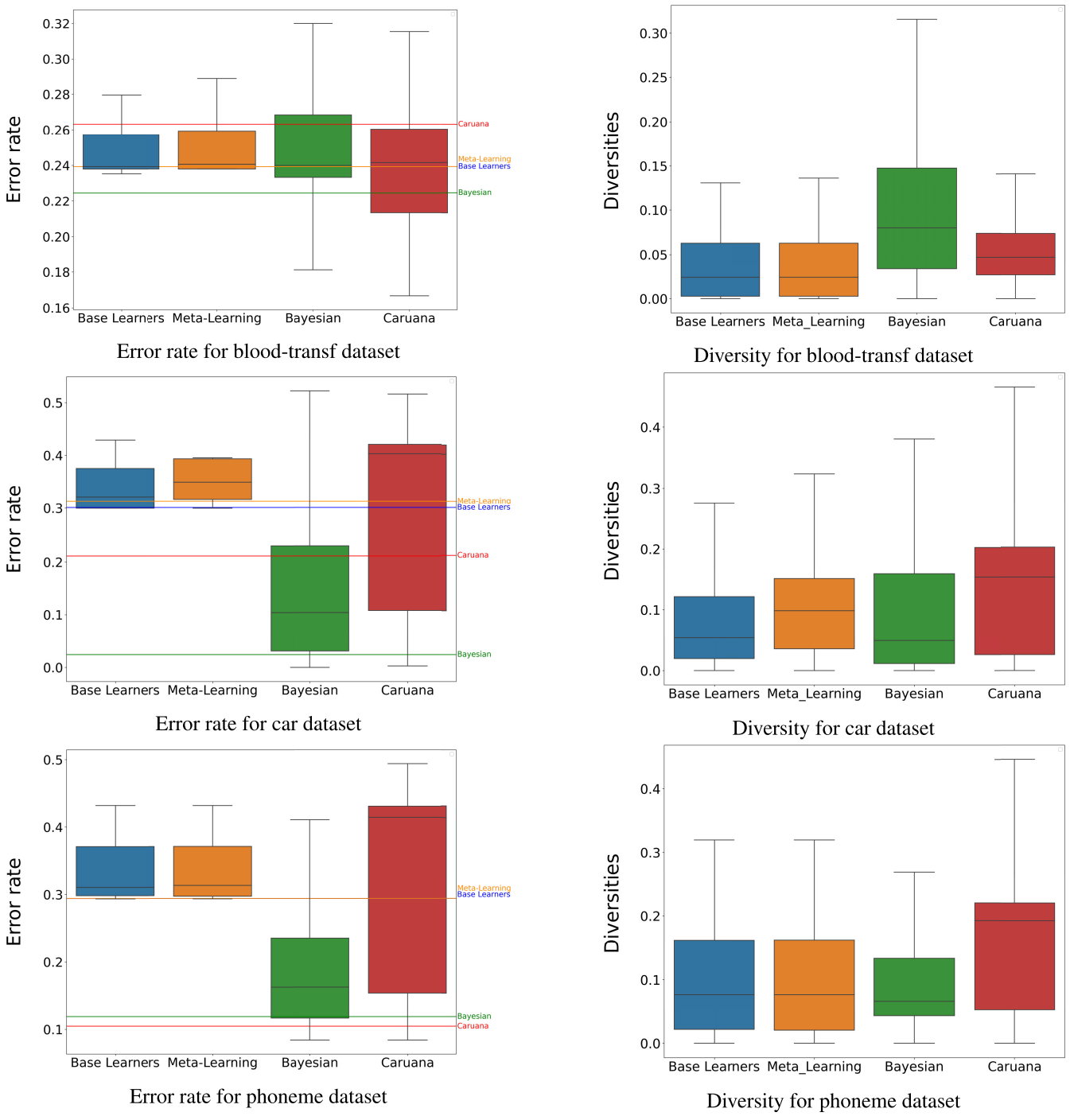

FIGURE 6. Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: blood transfusion, car, and phoneme. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_i)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.

All the pipeline sizes mentioned above as well as the parameters of the different methods involved in each stage were set at the default parameters of AUTO-SKLEARN that were suggested by the authors in [4]. We compute the error rates, $L(h_i)$ and diversity for all the pairs of pipelines at each stage, $d(h_i, h_j)$. We also created a majority-vote ensemble with the present pipelines at each state and computed its error rate $L(\Theta)$.

3) DATA PARTITION

The data is initially split into training and testing sets, with the training set further divided into training and validation

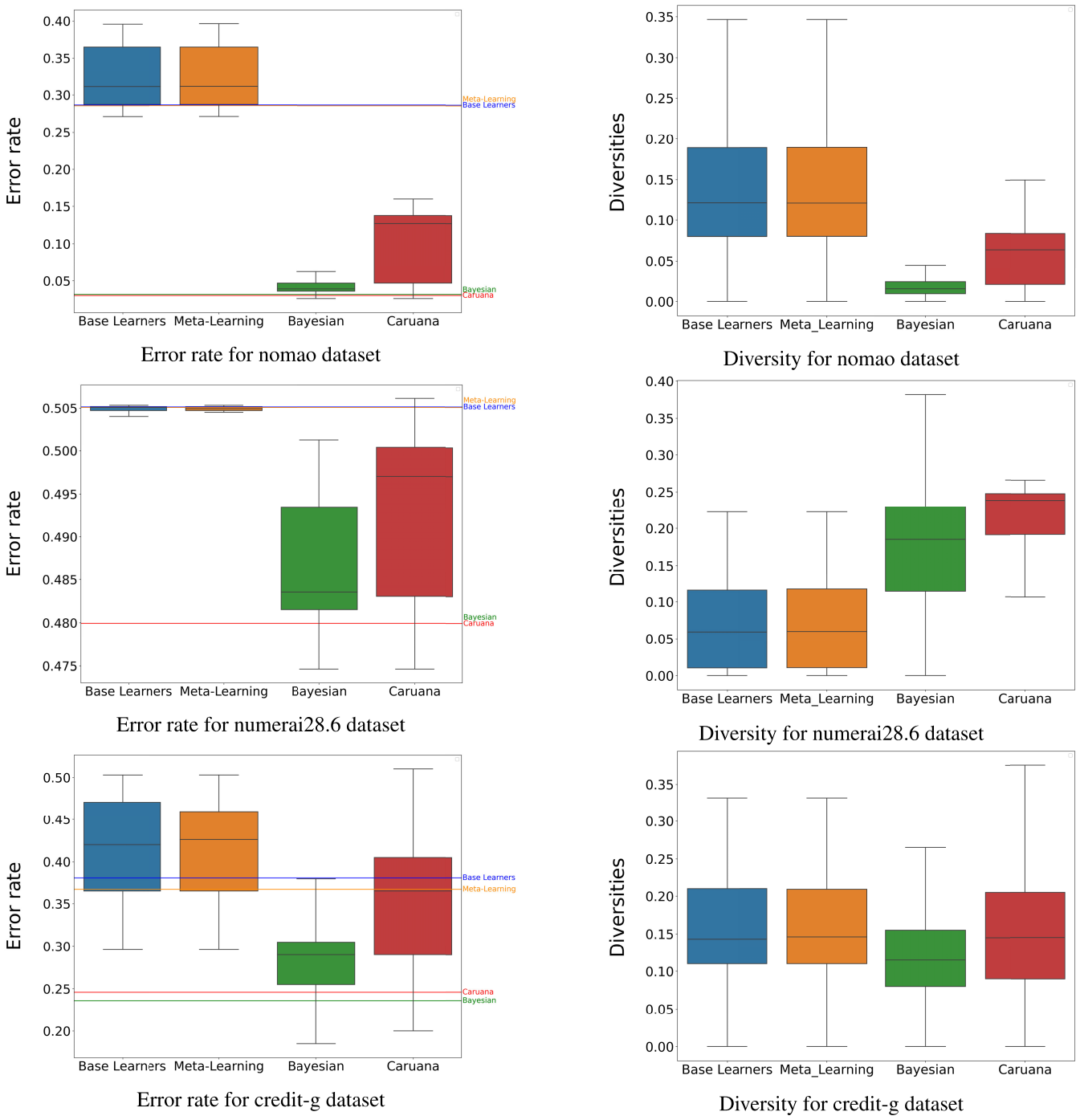

FIGURE 7. Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: nomao, numerai28.6, and credit-g. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_j)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.

subsets. The training data is used in three stages: base-learners, meta-learning, and Bayesian optimization, while the validation subset is reserved for the Caruana Ensemble stage.

During the base-learners stage, some pipelines may fail, resulting in the validation of 112 pipelines on average, though this number can vary by dataset. These validated pipelines are then used to generate predictions against the true labels. For the base-learners and meta-learning stages, we extract and analyze the pipelines recommended by AUTO-SKLEARN, with meta-learning focusing on 25 recommended pipelines.

For the ensemble stage, we follow AUTO-SKLEARN's resampling strategy to select pipelines, and after the Bayesian

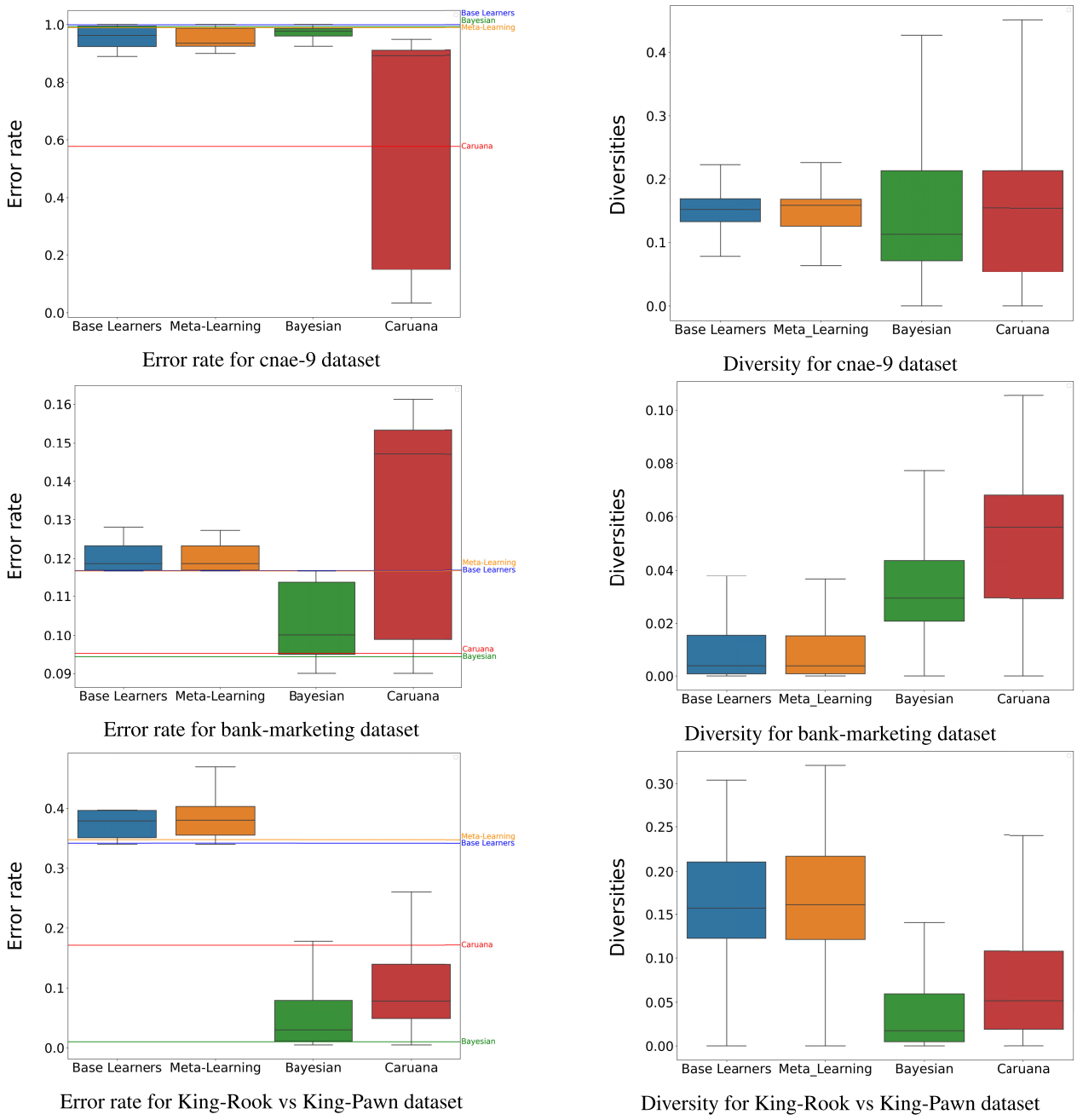

FIGURE 8. Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: cnae-9, bank-marketing, and King-Rook vs King-Pawn. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_i)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.

optimization stage, we use AUTO-SKLEARN's callback function to extract configurations, converting them into pipelines for further predictions. This process ensures consistent data partitioning and analysis across all stages.

V. EMPIRICAL RESULTS

A. ERROR RATE EVOLUTION

Figure 2 presents the results of this analysis for three datasets: Sylvine, Jasmine, and Miniboone. Each figure on the left displays four box plots, corresponding to the four phases of Caruana's Ensemble method. These box plots represent the distribution of error rates for the individual models (or pipelines) generated during each phase. For example, in the Caruana phase, which consists of 25 models/pipelines, the red box plot illustrates the error rates $L(h_i)$ for all 25 models.

Additionally, for each of the four phases, we constructed an ensemble using the existing pipelines and calculated its overall error rate $L(\Theta)$. These ensemble error rates

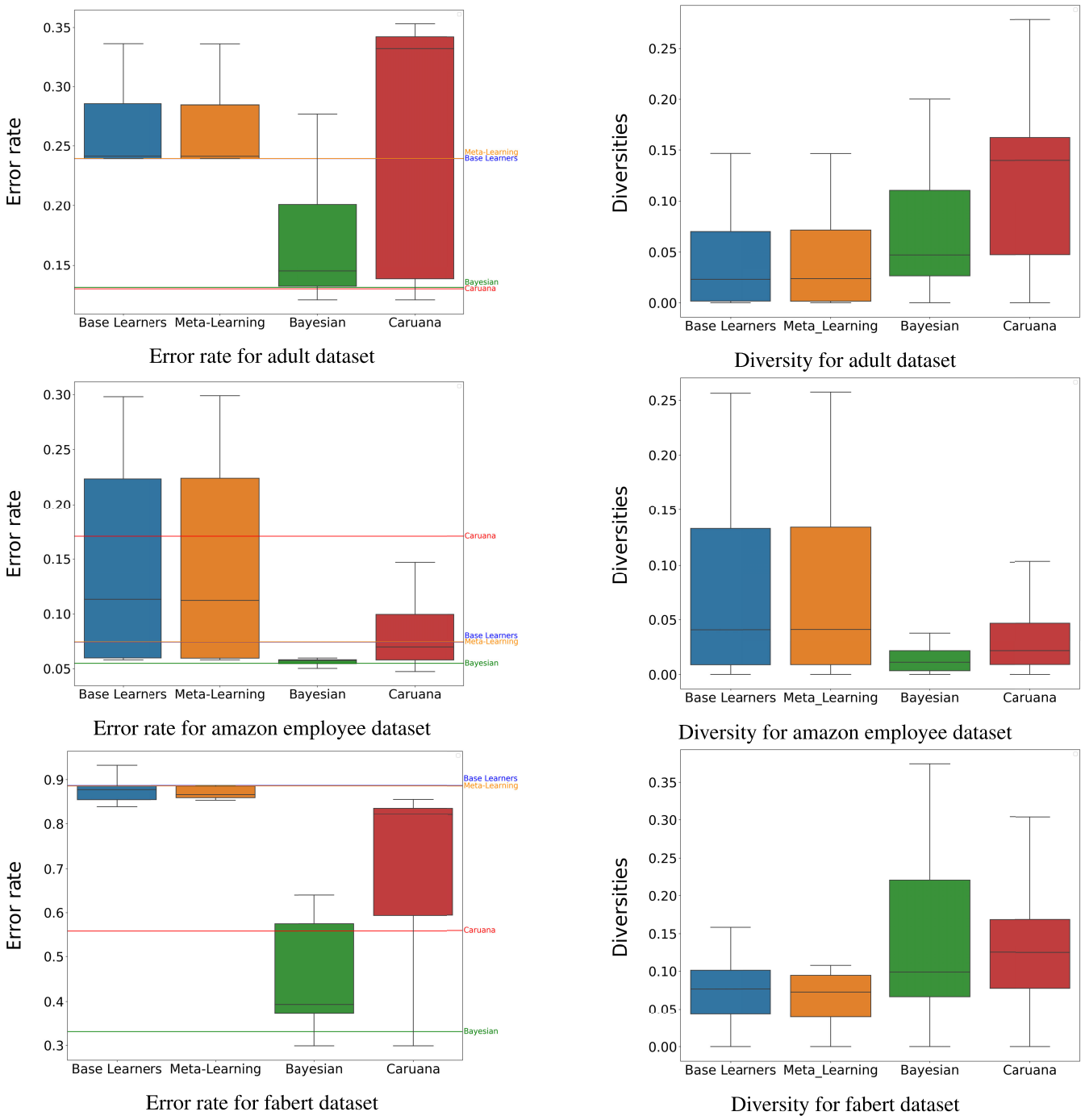

FIGURE 9. Analysis of error rates and diversity across the four phases of Caruana's Ensemble method for three datasets: adult, amazon employee, and fabert. The box plots on the left depict the distribution of individual pipeline error rates ($L(h_i)$) at each phase. Horizontal lines indicate the overall error rate ($L(\Theta)$) of the ensembles constructed from the pipelines in each phase, allowing a comparison between individual pipeline performance and ensemble performance. The box plots on the right depicts the diversity $d(h_i, h_i)$ of individual pipelines at each phase where Equation 2 computes the diversity of the possible pairs of pipelines and Equation 3 also computed the total diversity $D(\Theta)$ across all the possible pairs of predictors.

are represented by four distinct horizontal lines in the figure, providing a comparison between individual pipeline performance and the combined ensemble performance.

Figure 3(a) presents an aggregation of the individual pipeline error rates across all 20 datasets. Specifically, the box plots in this figure illustrate the average error rate of the pipelines generated at each phase, calculated as $\frac{1}{K}\sum_i L(h_i)$. This visualization highlights the overall behavior of these error rates across the datasets. Similarly, Figure 3(c) displays the distribution of ensemble error rates for each phase across the 20 datasets, offering insight into the performance of the ensembles at different stages.

TABLE 2. Descriptive Statistics for pipeline error rates across the different stages for 20 datasets.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.40	0.40	0.21	0.31
Std	0.22	0.22	0.13	0.18
Min	0.10	0.10	0.01	0.02
25th Percentile	0.24	0.24	0.10	0.13
Median	0.38	0.40	0.19	0.32
75th Percentile	0.50	0.50	0.30	0.41
Max	0.87	0.87	0.49	0.70

TABLE 3. Descriptive Statistics for pipeline diversity across the different stages for 20 datasets.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.10	0.10	0.08	0.12
Std	0.05	0.05	0.04	0.07
Min	0.02	0.02	0.01	0.01
25th Percentile	0.06	0.06	0.04	0.06
Median	0.09	0.09	0.07	0.13
75th Percentile	0.15	0.14	0.10	0.18
Max	0.17	0.18	0.17	0.21

TABLE 4. Descriptive statistics for pipeline error rate for air pressure systems failure.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.10	0.10	0.01	0.02
Std	0.26	0.26	0.00	0.01
Min	0.0181	0.0181	0.0052	0.0080
25th Percentile	0.02	0.02	0.01	0.01
Median (50th Percentile)	0.02	0.02	0.01	0.02
75th Percentile	0.02	0.02	0.01	0.03
Max	0.9373	0.9373	0.0247	0.0303

TABLE 5. Descriptive statistics for pipeline diversity for air pressure systems failure.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.08	0.08	0.01	0.01
Std	0.25	0.25	0.01	0.01
Min	0.00	0.00	0.00	0.00
25th Percentile	0.00	0.00	0.00	0.00
Median (50th Percentile)	0.00	0.00	0.00	0.00
75th Percentile	0.00	0.00	0.01	0.01
Max	0.9235	0.9235	0.0219	0.0236

TABLE 6. Descriptive statistics for pipeline error rate for sylvine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.50	0.51	0.13	0.38
Std	0.01	0.01	0.12	0.18
Min	0.49	0.50	0.038	0.038
25th Percentile	0.49	0.50	0.06	0.20
Median (50th Percentile)	0.50	0.50	0.09	0.49
75th Percentile	0.51	0.51	0.14	0.50
Max	0.52	0.52	0.51	0.53

As shown in these figures, the initial two stages—Base and Meta—tend to produce pipelines with relatively high error rates. This observation aligns with the statistics presented in Table 2, where the Meta-Learning and Base-Learners phases exhibit the highest median error rates, 0.40 and 0.38, respectively. These results indicate that while these initial stages generate diverse models, their accuracy remains relatively limited compared to later phases.

The Bayesian Optimization (BO) stage generates new pipelines with significantly lower error rates, achieving

TABLE 7. Descriptive statistics for pipeline diversity for sylvine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.13	0.12	0.08	0.20
Std	0.08	0.08	0.11	0.14
Min	0.00	0.00	0.00	0.00
25th Percentile	0.07	0.07	0.02	0.05
Median (50th Percentile)	0.12	0.12	0.04	0.24
75th Percentile	0.16	0.16	0.08	0.26
Max	0.26	0.26	0.50	0.51

TABLE 8. Descriptive statistics for pipeline error rate for albert.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.50	0.50	0.32	0.47
Std	0.00	0.00	0.02	0.05
Min	0.4988	0.4995	0.3051	0.3576
25th Percentile	0.50	0.50	0.31	0.45
Median (50th Percentile)	0.50	0.50	0.31	0.50
75th Percentile	0.50	0.50	0.33	0.50
Max	0.5031	0.5031	0.3453	0.5028

TABLE 9. Descriptive statistics for pipeline diversity for albert.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.06	0.06	0.04	0.21
Std	0.05	0.05	0.04	0.09
Min	0.00	0.00	0.00	0.00
25th Percentile	0.01	0.01	0.00	0.21
Median (50th Percentile)	0.04	0.04	0.00	0.24
75th Percentile	0.09	0.10	0.07	0.25
Max	0.16	0.16	0.10	0.32

TABLE 10. Descriptive statistics for pipeline error rate for jasmine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.48	0.48	0.24	0.40
Std	0.02	0.02	0.07	0.14
Min	0.4571	0.4571	0.1558	0.1692
25th Percentile	0.48	0.47	0.20	0.22
Median (50th Percentile)	0.49	0.49	0.22	0.49
75th Percentile	0.49	0.50	0.25	0.50
Max	0.5024	0.5003	0.6493	0.5327

TABLE 11. Descriptive statistics for pipeline diversity for jasmine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.17	0.17	0.10	0.18
Std	0.09	0.09	0.08	0.13
Min	0.00	0.00	0.00	0.00
25th Percentile	0.13	0.12	0.05	0.07
Median (50th Percentile)	0.18	0.18	0.08	0.20
75th Percentile	0.21	0.22	0.12	0.26
Max	0.37	0.37	0.59	0.50

TABLE 12. Descriptive statistics for pipeline error rate for christine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.50	0.50	0.29	0.43
Std	0.02	0.02	0.02	0.10
Min	0.4720	0.4720	0.2518	0.2518
25th Percentile	0.49	0.49	0.27	0.30
Median (50th Percentile)	0.50	0.50	0.29	0.49
75th Percentile	0.50	0.50	0.30	0.50
Max	0.5408	0.5312	0.3370	0.5365

TABLE 13. Descriptive statistics for pipeline diversity for christine.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.17	0.16	0.07	0.20
Std	0.09	0.10	0.05	0.11
Min	0.00	0.00	0.00	0.00
25th Percentile	0.11	0.09	0.03	0.12
Median (50th Percentile)	0.20	0.19	0.08	0.24
75th Percentile	0.23	0.24	0.11	0.26
Max	0.31	0.30	0.18	0.40

TABLE 14. Descriptive Statistics for pipeline error rate for KC1.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.17	0.17	0.19	0.15
Std	0.04	0.04	0.10	0.01
Min	0.14	0.14	0.12	0.13
25th Percentile	0.15	0.15	0.15	0.14
Median (50th Percentile)	0.16	0.16	0.15	0.15
75th Percentile	0.17	0.17	0.19	0.15
Max	0.35	0.35	0.85	0.18

TABLE 15. Descriptive Statistics for pipeline diversity for KC1.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.06	0.06	0.09	0.04
Std	0.06	0.06	0.11	0.02
Min	0.00	0.00	0.00	0.00
25th Percentile	0.03	0.03	0.03	0.03
Median (50th Percentile)	0.04	0.04	0.05	0.05
75th Percentile	0.06	0.06	0.10	0.05
Max	0.31	0.31	0.85	0.10

TABLE 16. Descriptive statistics for pipeline error rate for Miniboone.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.12	0.12	0.08	0.07
Std	0.05	0.05	0.03	0.02
Min	0.07	0.07	0.05	0.05
25th Percentile	0.07	0.07	0.06	0.06
Median (50th Percentile)	0.10	0.10	0.07	0.06
75th Percentile	0.14	0.14	0.08	0.08
Max	0.22	0.22	0.13	0.14

TABLE 17. Descriptive statistics for pipeline diversity for miniboone.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.06	0.06	0.02	0.05
Std	0.05	0.05	0.02	0.03
Min	0.00	0.00	0.00	0.00
25th Percentile	0.02	0.02	0.00	0.04
Median (50th Percentile)	0.05	0.05	0.02	0.05
75th Percentile	0.08	0.08	0.02	0.07
Max	0.19	0.19	0.07	0.13

TABLE 18. Descriptive statistics for pipeline error rate for blood-transf.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.26	0.26	0.27	0.25
Std	0.05	0.05	0.08	0.06
Min	0.24	0.24	0.17	0.17
25th Percentile	0.24	0.24	0.23	0.21
Median (50th Percentile)	0.24	0.24	0.24	0.24
75th Percentile	0.26	0.26	0.27	0.26
Max	0.46	0.46	0.77	0.53

a mean error rate of 0.21, as shown in Table 2. This improvement arises because the BO stage actively learns from the training data to refine the pipelines selected in the earlier phases. Moreover, BO explores various pipeline configurations, identifying those with minimal error rates, resulting in pipelines with substantially better performance compared to the initial stages.

TABLE 19. Descriptive statistics for pipeline diversity for blood-transf.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.05	0.05	0.11	0.06
Std	0.07	0.07	0.11	0.07
Min	0.00	0.00	0.00	0.00
25th Percentile	0.00	0.00	0.03	0.03
Median (50th Percentile)	0.02	0.02	0.08	0.05
75th Percentile	0.06	0.06	0.15	0.07
Max	0.32	0.32	0.77	0.46

TABLE 20. Descriptive statistics for pipeline error rate for car.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.09	0.13	0.11	0.15
Std	0.12	0.13	0.15	0.13
Min	0.00	0.00	0.00	0.00
25th Percentile	0.02	0.04	0.01	0.03
Median (50th Percentile)	0.05	0.10	0.05	0.15
75th Percentile	0.12	0.15	0.16	0.20
Max	0.61	0.57	0.96	0.51

TABLE 21. Descriptive statistics for pipeline diversity for car.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.36	0.40	0.16	0.31
Std	0.10	0.11	0.17	0.17
Min	0.30	0.30	0.00	0.00
25th Percentile	0.30	0.32	0.03	0.11
Median (50th Percentile)	0.32	0.35	0.10	0.40
75th Percentile	0.38	0.39	0.23	0.42
Max	0.72	0.70	0.96	0.52

TABLE 22. Descriptive statistics for pipeline error rate for phoneme.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.33	0.34	0.18	0.34
Std	0.04	0.04	0.08	0.14
Min	0.29	0.29	0.08	0.08
25th Percentile	0.30	0.30	0.12	0.15
Median (50th Percentile)	0.31	0.31	0.16	0.41
75th Percentile	0.37	0.37	0.24	0.43
Max	0.43	0.43	0.64	0.49

TABLE 23. Descriptive statistics for pipeline diversity for phoneme.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.10	0.10	0.09	0.16
Std	0.09	0.09	0.07	0.11
Min	0.00	0.00	0.00	0.00
25th Percentile	0.02	0.02	0.04	0.05
Median (50th Percentile)	0.08	0.08	0.07	0.19
75th Percentile	0.16	0.16	0.13	0.22
Max	0.32	0.32	0.59	0.44

TABLE 24. Descriptive statistics for pipeline error rate for nomao.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.34	0.34	0.04	0.10
Std	0.09	0.09	0.01	0.05
Min	0.27	0.27	0.03	0.03
25th Percentile	0.29	0.29	0.04	0.05
Median (50th Percentile)	0.31	0.31	0.04	0.13
75th Percentile	0.36	0.36	0.05	0.14
Max	0.63	0.63	0.08	0.16

TABLE 25. Descriptive statistics for pipeline diversity for nomao.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.15	0.15	0.02	0.06
Std	0.12	0.12	0.01	0.04
Min	0.00	0.00	0.00	0.00
25th Percentile	0.08	0.08	0.01	0.02
Median (50th Percentile)	0.12	0.12	0.02	0.06
75th Percentile	0.19	0.19	0.02	0.08
Max	0.61	0.61	0.06	0.15

TABLE 26. Descriptive statistics for pipeline error rate for numerai28.6.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.50	0.50	0.49	0.49
Std	0.00	0.00	0.01	0.01
Min	0.50	0.50	0.47	0.47
25th Percentile	0.50	0.50	0.48	0.48
Median (50th Percentile)	0.51	0.50	0.48	0.50
75th Percentile	0.51	0.51	0.49	0.50
Max	0.51	0.51	0.52	0.51

TABLE 27. Descriptive statistics for pipeline diversity for numerai28.6.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.08	0.08	0.17	0.20
Std	0.08	0.08	0.08	0.07
Min	0.00	0.00	0.00	0.00
25th Percentile	0.01	0.01	0.11	0.19
Median (50th Percentile)	0.06	0.06	0.19	0.24
75th Percentile	0.12	0.12	0.23	0.25
Max	0.22	0.22	0.42	0.27

TABLE 28. Descriptive statistics for pipeline error rate for king-rook vs king-pawn.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.39	0.40	0.08	0.12
Std	0.06	0.07	0.12	0.12
Min	0.34	0.34	0.00	0.00
25th Percentile	0.35	0.35	0.01	0.05
Median (50th Percentile)	0.38	0.38	0.03	0.08
75th Percentile	0.40	0.40	0.08	0.14
Max	0.55	0.59	0.48	0.48

TABLE 29. Descriptive statistics for pipeline diversity for king-rook vs king-pawn.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.17	0.18	0.06	0.09
Std	0.10	0.10	0.11	0.11
Min	0.00	0.00	0.00	0.00
25th Percentile	0.12	0.12	0.00	0.02
Median (50th Percentile)	0.16	0.16	0.02	0.05
75th Percentile	0.21	0.22	0.06	0.11
Max	0.48	0.50	0.48	0.48

Interestingly, the final stage, Caruana's phase, often produces pipelines with higher error rates than those from the BO stage, with a mean error rate of 0.31 and a standard deviation of 0.18, as reported in Table 2. However, as shown in Figure 3(c), the performance of ensembles at the Caruana phase is comparable to those at the BO phase, despite Caruana's Ensembles being built with only 25 pipelines. This suggests that Caruana's method leverages diversity effectively, balancing moderate individual error rates with ensemble-level performance.

TABLE 30. Descriptive statistics for pipeline error rate for bank-marketing.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.13	0.13	0.11	0.13
Std	0.04	0.04	0.02	0.03
Min	0.12	0.12	0.09	0.09
25th Percentile	0.12	0.12	0.09	0.10
Median (50th Percentile)	0.12	0.12	0.10	0.15
75th Percentile	0.12	0.12	0.11	0.15
Max	0.31	0.31	0.20	0.16

TABLE 31. Descriptive statistics for pipeline diversity for bank-marketing.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.02	0.02	0.04	0.05
Std	0.05	0.05	0.03	0.03
Min	0.00	0.00	0.00	0.00
25th Percentile	0.00	0.00	0.02	0.03
Median (50th Percentile)	0.00	0.00	0.03	0.06
75th Percentile	0.02	0.02	0.04	0.07
Max	0.23	0.23	0.18	0.11

TABLE 32. Descriptive statistics for pipeline error rate for adult.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.29	0.29	0.17	0.27
Std	0.13	0.13	0.05	0.09
Min	0.24	0.24	0.12	0.12
25th Percentile	0.24	0.24	0.13	0.14
Median (50th Percentile)	0.24	0.24	0.14	0.33
75th Percentile	0.29	0.28	0.20	0.34
Max	0.73	0.73	0.28	0.35

TABLE 33. Descriptive statistics for pipeline diversity for adult.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.09	0.09	0.07	0.12
Std	0.17	0.17	0.05	0.08
Min	0.00	0.00	0.00	0.00
25th Percentile	0.00	0.00	0.03	0.05
Median (50th Percentile)	0.02	0.02	0.05	0.14
75th Percentile	0.07	0.07	0.11	0.16
Max	0.71	0.71	0.20	0.28

TABLE 34. Descriptive statistics for pipeline error rate for CNAE-9.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.96	0.95	0.97	0.65
Std	0.04	0.03	0.03	0.37
Min	0.89	0.90	0.82	0.03
25th Percentile	0.92	0.93	0.96	0.15
Median (50th Percentile)	0.96	0.94	0.98	0.89
75th Percentile	0.99	0.99	0.99	0.91
Max	1.00	1.00	1.00	0.95

TABLE 35. Descriptive statistics for pipeline diversity for CNAE-9.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.15	0.14	0.16	0.19
Std	0.03	0.05	0.14	0.19
Min	0.00	0.00	0.00	0.00
25th Percentile	0.13	0.13	0.07	0.05
Median (50th Percentile)	0.15	0.16	0.11	0.15
75th Percentile	0.17	0.17	0.21	0.21
Max	0.23	0.23	0.71	0.71

Additional visualizations illustrating the error rates and diversity across different stages for 17 datasets can be found in the appendix A. These supplementary figures provide a

TABLE 36. Descriptive statistics for pipeline error rate for amazon employee.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.14	0.14	0.08	0.12
Std	0.08	0.08	0.07	0.10
Min	0.06	0.06	0.05	0.05
25th Percentile	0.06	0.06	0.05	0.06
Median (50th Percentile)	0.11	0.11	0.06	0.07
75th Percentile	0.22	0.22	0.06	0.10
Max	0.30	0.30	0.37	0.39

TABLE 37. Descriptive statistics for pipeline diversity for amazon employee.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.07	0.07	0.03	0.06
Std	0.07	0.07	0.07	0.10
Min	0.00	0.00	0.00	0.00
25th Percentile	0.01	0.01	0.00	0.01
Median (50th Percentile)	0.04	0.04	0.01	0.02
75th Percentile	0.13	0.13	0.02	0.05
Max	0.26	0.26	0.35	0.36

detailed breakdown of the results and further support the analysis discussed in this section.

B. DIVERSITY EVOLUTION

Figure 2 presents the results of the diversity analyses for 3 datasets: Sylvine, Jasmine, and Minibone. Each boxplot on the right corresponds to the diversity of the different phases for the individual pipelines generated during each phase. We calculate the diversity between the individual pairs of pipelines $d(h_i, h_i)$ 2. We also calculate the total diversity of the ensemble $D(\Theta)$ for each of the four phases by averaging the diversity of all the possible pairs of predictors. The results are shown at the right side of Figure 2 with datasets Sylvine, jasmine, and Minibone. Figure 3 (b) shows the aggregation of the individual pipeline diversity for all the 20 datasets. The boxplot illustrates the diversity of the pipelines across the different phases highlighting their performances across the different phases. The two stages; Base-Learners and Meta-Learning produces pipelines with relatively high diversity which aligns with the statistics in Table 3 with mean and median diversity of 0.09. While Bayesian selects pipelines with relatively low diversity, the Caruana does the opposite. How is it possible that the final Caruana stage ends up selecting pipelines with higher error rates? And why do Caruana's Ensembles remain the most competitive? This apparent paradox is clarified in Figures 3(b) and 3(c), which display the diversity of pipelines produced at each stage using box plots and the majority vote ensemble error rates for each stage. These figures, together with Table 3, highlight that while the BO stage generates pipelines with lower individual models error rates, it does so at the cost of reduced diversity. In contrast, the Caruana stage produces pipelines with significantly higher diversity despite their higher error rates but, at the end, it ends up building better ensembles as shown in Figure 3(c).

Table 3 provides a quantitative perspective on diversity across different methods. The Caruana phase achieves the highest mean diversity (0.12), surpassing Bayesian optimization (0.08) and even base/meta-learners (0.10). Furthermore, it attains the highest maximum diversity (0.2114), underscoring its ability to explore and leverage a broad range of

TABLE 38. Descriptive statistics for pipeline error rate for Credit-G.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.41	0.41	0.30	0.35
Std	0.07	0.07	0.07	0.07
Min	0.30	0.30	0.19	0.20
25th Percentile	0.36	0.36	0.26	0.29
Median (50th Percentile)	0.42	0.43	0.29	0.36
75th Percentile	0.47	0.46	0.30	0.41
Max	0.50	0.50	0.70	0.51

TABLE 39. Descriptive statistics for pipeline diversity for Credit-G.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.15	0.15	0.13	0.15
Std	0.08	0.08	0.09	0.08
Min	0.0	0.0	0.0	0.0
25th Percentile	0.11	0.11	0.08	0.09
Median (50th Percentile)	0.14	0.15	0.12	0.14
75th Percentile	0.21	0.21	0.15	0.20
Max	0.33	0.33	0.70	0.40

TABLE 40. Descriptive statistics for pipeline error rate for vehicle.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.76	0.76	0.34	0.57
Std	0.02	0.02	0.16	0.25
Min	0.71	0.71	0.13	0.13
25th Percentile	0.74	0.75	0.22	0.24
Median (50th Percentile)	0.76	0.76	0.28	0.72
75th Percentile	0.78	0.78	0.43	0.75
Max	0.83	0.78	0.78	0.80

TABLE 41. Descriptive statistics for pipeline diversity for vehicle.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.15	0.14	0.16	0.19
Std	0.03	0.05	0.14	0.19
Min	0.00	0.00	0.00	0.00
25th Percentile	0.13	0.13	0.07	0.05
Median (50th Percentile)	0.15	0.16	0.11	0.15
75th Percentile	0.17	0.17	0.21	0.21
Max	0.23	0.23	0.71	0.71

TABLE 42. Descriptive statistics for pipeline error rate for fabert.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.87	0.87	0.45	0.70
Std	0.02	0.01	0.12	0.19
Min	0.84	0.85	0.30	0.30
25th Percentile	0.86	0.86	0.37	0.59
Median (50th Percentile)	0.88	0.87	0.39	0.82
75th Percentile	0.89	0.89	0.58	0.84
Max	0.93	0.89	0.64	0.86

TABLE 43. Descriptive statistics for pipeline diversity for fabert.

Statistic	Base	Meta	Bayesian	Caruana
Mean	0.07	0.06	0.14	0.17
Std	0.04	0.04	0.11	0.15
Min	0.00	0.00	0.00	0.00
25th Percentile	0.04	0.04	0.07	0.08
Median (50th Percentile)	0.08	0.07	0.10	0.13
75th Percentile	0.10	0.09	0.22	0.17
Max	0.16	0.11	0.37	0.59

pipelines. Interestingly, the standard deviation of diversity is relatively low, suggesting that the Caruana stage consistently maintains diversity across datasets. These findings reinforce the patterns observed in Figure 3, confirming that Caruana's method is particularly effective in enhancing ensemble diversity.

When considering both error rates and diversity together, a clear pattern emerges: the BO stage pipelines achieve lower individual error rates but lack diversity, whereas the Caruana stage pipelines, despite higher error rates, contribute much greater diversity to the ensemble. This finding aligns with the theoretical insights on generalization performance for majority vote ensembles discussed in Section III. These insights emphasize that an ensemble's overall performance depends on balancing individual pipeline error rates with ensemble diversity.

The Caruana stage excels in this balance, as its highly diverse pipelines enhance the ensemble's ability to generalize effectively. This diversity compensates for the higher individual error rates, enabling the Caruana Ensemble to achieve superior overall error rates compared to ensembles built in the BO stage. The results from Table 3 further validate this conclusion by demonstrating that Caruana's approach consistently achieves the highest diversity levels, reinforcing the critical role of diversity in ensemble generalization.

C. DISCUSSION

According to our findings, the answers to the specific research questions that we posed in this paper are the following:

- Why is it essential to have distinct phases in this process?
- What roles do meta-learning, Bayesian optimization, and the final ensemble phase play in enhancing model performance?
- Are their efforts directed primarily at improving individual models, or do they explicitly or implicitly enhance the diversity of the ensemble?
- How do these stages collectively contribute to building better ensembles?

Regarding the first question, as stated above, recent theoretical findings showed that the performance and generalization capacity of ensembles depend on the performance of the individual models and their diversity. Our empirical results showed that having different phases is essential because the methodologies applied in some stages contribute to improving the performance of individual models, while others contribute to improving the diversity of the ensemble.

As for the second question and the third question, our findings showed that the meta-learning phase does not have a strong contribution to the performance of the AUTO-SKLEARN method because neither improved the error rate of the base classifiers nor the diversity; the BO phase contributes to improving the performance of the individual classifiers at the expense of reducing diversity; and the Caruana phase, despite including pipelines with higher error rates, fosters significantly the diversity, which compensates for individual inaccuracies and enhances ensemble generalization.

Finally, regarding the last question, the experimentation performed proved that the combination of the different phases of AUTO-SKLEARN contributed to a nuanced trade-off between individual model accuracy and ensemble diversity and emphasized that diversity is paramount to achieving robust ensemble performance.These results reinforce the theoretical understanding that ensemble performance is a function of both the accuracy and diversity of its constituent models.

VI. QUANTITATIVE ANALYSIS OF ERROR RATE AND DIVERSITY

To better understand the impact of diversity on ensemble performance and error rate, we present the following statistics. These statistics provide insight into 20 datasets used for this work. The following tables summarize these key findings.

Tables 2 and 3 provide descriptive statistics for four distinct phases—Base-Learners, Meta-Learning, Bayesian, and Caruana—across 20 datasets, focusing on error rates and diversity. In Table 2, which summarizes error rate statistics, all methods exhibit similar average error rates around 0.40. Meta-Learning and Base-Learners show the highest medians, 0.40 and 0.38, respectively, indicating relatively balanced performance. Bayesian achieves the lowest mean error rate (0.21), as it optimizes individual models for accuracy. However, Caruana, with a mean of 0.31 and a standard deviation of 0.18, balances moderate error rates with ensemble-level performance by leveraging diversity.

Table 3, focused on diversity, demonstrates reduced overall means, with base-learners and meta-learners averaging 0.10, Bayesian at 0.08, and Caruana achieving 0.12. This aligns with the 3, where Caruana's method excels in enhancing ensemble diversity despite higher individual error rates. The variability in diversity, indicated by standard deviations, is lower, suggesting more consistent diversity outcomes across the datasets. The Caruana phase achieves the highest maximum diversity (0.2114), reflecting its ability to leverage diverse pipelines for improved generalization. Additional descriptive statistics for 20 datasets focusing on error rate and diversity can be found in the B from Table 4, 5 to Table 42, 43.

VII. CONCLUSION AND LIMITATIONS

This study delves into the critical role of diversity in ensemble-based Automated Machine Learning (AutoML) systems, providing both theoretical and empirical insights. By extending the understanding of ensemble diversity to AutoML frameworks such as AUTO-SKLEARN, this work bridges the gap between empirical observations and theoretical explanations. Through a detailed analysis of the phases in ensemble construction—including base-learners, meta-learning, Bayesian optimization, and Caruana Ensembles—we illustrate how each phase contributes to the overall generalization performance.

While our study focuses on the role of diversity in ensemble-based AutoML methods, we acknowledge that variance, bias, and model complexity are also critical factors influencing generalization performance. Our analysis aims to extend the understanding of how diversity **alone** contributes to improved generalization in AutoML ensembles. However, a comprehensive exploration of the interplay between diversity, bias, variance, and model complexity remains an open question. Future research could investigate how these factors interact within AutoML frameworks, potentially leading to more effective ensemble construction strategies.

Here, it is also important to add that in this paper we have made an in-depth analysis of the role of the diversity of individual classifiers in the performance of ensemble-based AutoML methods. However, there are other factors that can affect the performance of these methods, depending on the purpose of their application, which we have not taken into account in this paper. Some of these factors are, for example, the computational complexity of the individual classifiers or their explainability.

Future work could explore integrating advanced diversity optimization techniques within AutoML frameworks to further enhance ensemble construction. Additionally, extending these insights to other AutoML platforms and diverse datasets could validate the generality of our findings. By combining theoretical rigor with empirical validation, this study contributes to the development of more effective and theoretically grounded AutoML methodologies, paving the way for their broader application and adoption.

APPENDIX A ADDITIONAL VISUALIZATIONS

This appendix contains additional figures that provide further insight into the error rate and diversity distributions for the various stages of Base-Learners, Meta-learning, Bayesian and Caruana. They include Figures 4, 5, 6, 7, 8 and 9. These visualizations supplement the analysis discussed in section V by showing detailed comparisons and trends covered in the main text.

APPENDIX B DESCRIPTIVE STATISTICS FOR INDIVIDUAL DATASETS

This appendix contains additional information that provide further insights to the error and diversity statistics (mean, median, etc.) of the 20 combined and the individual statistics of the datasets. These Tables (4, 6, 8, 10, 12, 14, 16, 18, 20, 22, 24, 26, 28, 30, 32, 34, 36, 38, 40, 42) representing the statistics for the error rates and (5, 7, 9, 11, 13, 15, 17, 19, 21, 23, 25, 29, 31, 33, 35, 37, 39, 41, 43) representing statistics for diversity provides individual statistics to complement Tables 2 and 3 which are a combination of the 20 datasets used. The descriptive statistics of each of the 20 datasets are shown here in this section.

REFERENCES

[1] Z. Shen, Y. Zhang, L. Wei, H. Zhao, and Q. Yao, "Automated machine learning: From principles to practices," 2018, *arXiv:1810.13306*.

[2] X. He, K. Zhao, and X. Chu, "AutoML: A survey of the state-of-the-art," *Knowl.-Based Syst.*, vol. 212, Jan. 2021, Art. no. 106622.

[3] J. A. R. Marshall and T. G. Hinton, "Beyond no free lunch: Realistic algorithms for arbitrary problem classes," in *Proc. IEEE Congr. Evol. Comput.*, Jul. 2010, pp. 1–6.

[4] M. Feurer, A. Klein, K. Eggensperger, J. T. Springenberg, M. Blum, and F. Hutter, "Efficient and robust automated machine learning," *Adv. Neural Inf. Process. Syst.*, vol. 28, pp. 2755–2763, Dec. 2015.

[5] E. LeDell and S. Poirier, "H_2o automl: Scalable automatic machine learning," in *Proc. AutoML Workshop at ICML*, 2020, pp. 1–12.

[6] N. Erickson, J. Mueller, A. Shirkov, H. Zhang, P. Larroy, M. Li, and A. Smola, "AutoGluon-tabular: Robust and accurate AutoML for structured data," 2020, *arXiv:2003.06505*.

[7] C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown, "Auto-WEKA: Combined selection and hyperparameter optimization of classification algorithms," in *Proc. 19th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2013, pp. 847–855.

[8] R. S. Olson and J. H. Moore, "Tpot: A tree-based pipeline optimization tool for automating machine learning," in *Proc. Workshop Automat. Mach. Learn.*, 2016, pp. 66–74.

[9] R. Caruana, A. Niculescu-Mizil, G. Crew, and A. Ksikes, "Ensemble selection from libraries of models," in *Proc. 21st Int. Conf. Mach. Learn. (ICML)*, 2004, p. 18.

[10] A. R. Masegosa, "Learning under model misspecification: Applications to variational and ensemble methods," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Jan. 2020, pp. 5479–5491.

[11] A. R. Masegosa, S. Lorenzen, C. Igel, and Y. Seldin, "Second order PAC-Bayesian bounds for the weighted majority vote," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 33, Jan. 2020, pp. 5263–5273.

[12] Y.-S. Wu, A. R. Masegosa, S. Lorenzen, C. Igel, and Y. Seldin, "Chebyshev-cantelli PAC-bayes-Bennett inequality for the weighted majority vote," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 34, Dec. 2021, pp. 12625–12636.

[13] L. Á. O. Cuesta, R. Cabañas, and A. R. Masegosa, "Diversity and generalization in neural network ensembles," in *Proc. Int. Conf. Artif. Intell. Statist.*, Jan. 2021, pp. 11720–11743.

[14] P. Gijsbers, E. LeDell, J. Thomas, S. Poirier, B. Bischl, and J. Vanschoren, "An open source AutoML benchmark," 2019, *arXiv:1907.00909*.

[15] J. Snoek, H. Larochelle, and R. P. Adams, "Practical Bayesian optimization of machine learning algorithms," in *Proc. Adv. Neural Inf. Process. Syst.*, Jan. 2012, pp. 1–12.

[16] H. Jin, Q. Song, and X. Hu, "Auto-keras: An efficient neural architecture search system," in *Proc. 25th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2019, pp. 1946–1956.

[17] J. M. Kanter and K. Veeramachaneni, "Deep feature synthesis: Towards automating data science endeavors," in *Proc. IEEE Int. Conf. Data Sci. Adv. Anal. (DSAA)*, Oct. 2015, pp. 1–10.

[18] Featuretools Developers. *Featuretools: An Open Source Automated Feature Engineering Library*. Accessed: Oct. 17, 2024. [Online]. Available: https://www.featuretools.com

[19] R. Tibshirani, "Regression shrinkage and selection via the lasso," *J. Roy. Stat. Soc. Ser. B, Stat. Methodol.*, vol. 58, no. 1, pp. 267–288, Jan. 1996.

[20] DataRobot. *Automated Machine Learning Platform*. Accessed: Oct. 17, 2024. [Online]. Available: https://www.datarobot.com

[21] L. Breiman, "Bagging predictors," *Mach. Learn.*, vol. 24, no. 2, pp. 123–140, Aug. 1996.

[22] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, Aug. 1997.

[23] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," in *Proc. 22nd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2016, pp. 785–794.

[24] D. H. Wolpert, "Stacked generalization," *Neural Netw.*, vol. 5, no. 2, pp. 241–259, Jan. 1992.

[25] J. Koza, "Genetic programming as a means for programming computers by natural selection," *Statist. Comput.*, vol. 4, no. 2, pp. 87–112, Jun. 1994.

[26] J. Hoover. (2018). *Devol: Deep Evolutionary Learning for Neural Networks*. Accessed: Oct. 17, 2024. [Online]. Available: https://github.com/joeddav/devol

[27] C. Finn, P. Abbeel, and S. Levine, "Model-agnostic meta-learning for fast adaptation of deep networks," in *Proc. 34th Int. Conf. Mach. Learn. (ICML)*, Jan. 2017, pp. 1126–1135.

[28] B. J. McGarry and A. D. D. Gonzalez, "Meta-features for data mining," *Comput. Intell.*, vol. 31, no. 1, pp. 127–145, 2015.
[29] M. Andrychowicz, M. Denil, S. L. S. Gómez, M. W. Hoffman, D. Pfau, T. Schaul, B. Shillingford, and N. D. Freitas, "Learning to learn by gradient descent by gradient descent," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 29, Jan. 2016, pp. 1–18.
[30] G. Cloud. (2018). *Cloud Automl: Making Ai Accessible To Every Business*. Accessed: Oct. 22, 2024. [Online]. Available: https://blog.google/products/google-cloud/cloud-automl-making-ai-accessible-every-business/
[31] B. Zoph, V. Vasudevan, J. Shlens, and Q. V. Le, "Learning transferable architectures for scalable image recognition," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Jun. 2018, pp. 8697–8710.
[32] J. Barnes, "Azure machine learning," in *Microsoft Azure Essentials*, 1st ed., Redmond, WA, USA: Microsoft, 2015.
[33] P. Das et al., "Amazon SageMaker autopilot: A white box AutoML solution at scale," in *Proc. 4th Int. Workshop Data Manage. End-End Mach. Learn.*, Jun. 2020, pp. 1–7.
[34] M. A. Gelbart, J. Snoek, and R. P. Adams, "Bayesian optimization with unknown constraints," 2014, *arXiv:1403.5607*.
[35] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," 2010, *arXiv:1012.2599*.
[36] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. de Freitas, "Taking the human out of the loop: A review of Bayesian optimization," *Proc. IEEE*, vol. 104, no. 1, pp. 148–175, Jan. 2016.
[37] J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl, "Algorithms for hyper-parameter optimization," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 24, Dec. 2011, pp. 1–16.
[38] R. E. Schapire, "The strength of weak learnability," *Mach. Learn.*, vol. 5, no. 2, pp. 197–227, Jun. 1990.

SALOMEY OSEI received the M.Phil. degree in applied mathematics from the Kwame Nkrumah University of Science and Technology, the M.Sc. degree in industrial mathematics from the African Institute for Mathematical Science (AIMS), and the M.Sc. degree in machine intelligence from AMMI/AIMS, Ghana, fully funded by Google and Facebook. She is currently pursuing the Ph.D. degree with the Faculty of Engineering, University of Deusto.

She is currently a Research Assistant with DeustoTech. She is also a COFUND Marie Sklodowska-Curie Fellow working on fostering data-driven decision making in transportation through automatic machine learning. She has been a recipient of other prestigious scholarships, including the MasterCard Foundation Scholarship. Before enrolling into the Ph.D. programme, her working experiences has been in the sectors of ML applied to NLP. She is actively involved in the organization of other grassroot movements, such as Ghana NLP and Masakhane with publications in central NLP conferences, TACL which were co-presented at Findings of EMNLP. Additionally, she has been an Organizer of the Black in AI, Women in Machine Learning (WiML) and Women in Machine Learning and Data Science (WiMLDS). She is very passionate about mentoring students, especially females in STEM.

ANDRÉS R. MASEGOSA received the bachelor's and Ph.D. degrees in computer science from the University of Granada, Spain, in 2003 and 2009, respectively.

He is currently an Associate Professor in computer sciences with Aalborg University, Copenhagen Campus, Denmark. The result of this research work is reflected in the publication of more than 25 papers in journals indexed in JCR, more than 30 contributions in international conferences in the field. He also participated in numerous research projects, being the main researcher in two of them. He usually carries out reviews of scientific papers for different journals and international top conferences in machine learning. His research interests include probabilistic machine learning, with a focus on uncertainty quantification and robust and reliable machine learning.

ANTONIO D. MASEGOSA received the University degree in computer engineering and the Ph.D. degree in computer sciences from the University of Granada, Spain, in 2005 and 2010, respectively.

From June 2010 to November 2014, he was a Postdoctoral Researcher with the Research Center for ICT, University of Granada. In 2014, he received an IKERBASQUE Research Fellowship to work in the Mobility Unit of the Deusto Institute of Technology, Bilbao, Spain. In 2019, he was awarded the IKERBASQUE Research Associate mention. Currently, he is the Principal Investigator of the Deusto Smart Mobility Group. He has published four books, 29 JCR papers, and more than 30 papers in both international and national conferences. His main research interests include artificial intelligence, intelligent systems, soft computing, hybrid metaheuristics, machine learning, deep learning, intelligent transportation systems, logistic networks, travel behavior analysis, traffic forecasting, and traffic accident prediction.